Bregman proximal minimization algorithms, analysis and applications

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Mr. Mahesh Chandra Mukkamala

aus Jolapuram, Indien

Tübingen

2021

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	16.11.2021
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Peter Ochs
2. Berichterstatterin:	Prof. Dr. Emilie Chouzenoux
3. Berichterstatter:	Prof. Dr. David Russell Luke

Acknowledgements

First and foremost, I would like to thank my advisor, Prof. Dr. Peter Ochs. Without his guidance and consistent support, it would have been difficult, if not impossible for me to turn my passion of being an able researcher into reality. His knack for finding my mistakes in the minutest of mathematical details has helped me in learning at a faster pace and improving with time. His perfection has led us to publish some very high quality research papers which would otherwise have been impossible. I will forever be indebted to my advisor for accepting me as his PhD student.

I am deep grateful to the German Research Foundation, for providing me with the financial support for the entirety of my PhD via DFG Grant OC 150/1-1.

I also convey my thanks to my second advisor, Prof. Dr. Jürgen Hausen. Even though our interaction was brief, his quick responsiveness in certain administrative tasks is quite commendable and a quality which I should also aim at. I also thank Prof. Dr. Emilie Chouzenoux for accepting to be a reviewer for my thesis. I would also like to thank my research collaborators, Prof. Dr. Shoham Sabach, Prof. Dr. Thomas Pock, Felix Westerkamp, Emanuel Laude, Prof. Dr. Daniel Cremers, Prof. Dr. Jalal Fadili. Their collaborations have helped me gain the technical knowledge and the experience to solve hard and complicated problems patiently and efficiently.

I would like to convey my thanks to Prof. Dr. Matthias Hein as he was the one who inspired me to embark on this tough path of optimization and machine learning related research, in the early stages of my career.

My big thanks to Ellen Wintringer, Dr. Lars Schneider, Martina Jung for their support with the administrative issues and Peter W. K. Franke for his support with the information and technical issues. Also, I am thankful to my colleagues, Sheheryar Mehmood, Jan-Hendrik Lange, Shida Wang, Oskar Adolfson for all the fruitful discussions we had during our research seminar, and support in general.

I would like to thank my family for their love and affection, without whom I would not have been able to pursue my research ambitions. I convey special thanks to my wife Anu, for her continuous support, understanding and love. Without her my life would be just incomplete. My son Shiva was born while I was writing my thesis. Apart from his innocent love towards me, his full efforts to not let me finish my thesis is commendable.

For

Anu and Shiva

Abstract

In this thesis, we tackle the optimization of several non-smooth and non-convex objectives that arise in practice. The classical results in context of Proximal Gradient algorithms rely on the so-called Lipschitz continuous gradient property. Such conditions do not hold for many objectives in practice, including the objectives arising in matrix factorization, deep neural networks, phase retrieval, image denoising and many others. Recent development, namely, the L-smad property allows us to deal with such objectives via the so-called Bregman distances, which generalize the Euclidean distance. Based on the L-smad property, Bregman Proximal Gradient (BPG) algorithm is already well-known. In our work, we propose an inertial variant of BPG, namely, CoCaIn BPG which incorporates adaptive inertia based on the function's local behavior. Moreover, we prove the global convergence of the sequence generated by CoCaIn BPG to a critical point of the function. CoCaIn BPG outperforms BPG with a significant margin, which is attributed to the proposed non-standard double backtracking technique. A major challenge in working with BPG based methods is designing the Bregman distance that is suitable for the objective. In this regard, we propose Bregman distances that are suitable to three applications, matrix factorization, deep matrix factorization and deep neural networks. We start with the matrix factorization setting and propose the relevant Bregman distances, then we tackle the deep matrix factorization and deep neural network settings. In all these settings, we also propose the closed form update steps for BPG based methods, which is crucial for practical application. We also propose the closed form inertia that is suitable for efficient application of CoCaIn BPG. However, until here the setting is restricted to additive composite problems and generic composite problems such as the objectives that arise in robust phase retrieval are out of the scope. In order to tackle generic composite problems, the L-smad property needs to be generalized even further. In this regard, we propose MAP property and based on which we propose Model BPG algorithm. The classical techniques of the convergence analysis based on the function value proved to be restrictive. Thus, we propose a novel Lyapunov function that is suitable for the global convergence analysis. We later unify Model BPG and CoCaIn BPG, to propose Model CoCaIn BPG for which we provide the global convergence results. We supplement all our theoretical results with relevant empirical observations to show the competitive performance of our methods compared to existing state of the art optimization methods.

Keywords: Composite non-convex non-smooth minimization, non-Euclidean distances, Bregman distance, Bregman Proximal Gradient method, inertial methods, deep learning, matrix factorization, deep linear neural networks, global convergence, model functions, Lyapunov function.

Zusammenfassung

In dieser Arbeit beschäftigen wir uns mit der Optimierung mehrerer nicht-glatter und nicht-konvexer Probleme, die in der Praxis auftreten. Die klassischen Ergebnisse im Zusammenhang mit proximalen Gradientenverfahren beruhen auf der sogenannten Lipschitz-kontinuierlichen Gradienteneigenschaft. Diese Bedingungen gelten jedoch nicht für alle Zielfunktionen in der Praxis, einschließlich der Probleme, die bei der Matrixfaktorisierung, tiefen neuronalen Netzen, dem Phase retrieval, der Bildentrauschung und vielen anderen auftreten. Eine neuere Entwicklung, nämlich die L-smad-Eigenschaft, erlaubt es uns, solche Probleme über die sogenannten Bregman-Distanzen zu behandeln, die die euklidische Distanz verallgemeinern. Basierend auf der L-smad-Eigenschaft ist der Bregman Proximal Gradient (BPG) Algorithmus bereits bekannt. In unserer Arbeit schlagen wir eine variante von BPG mit Momentum vor, nämlich CoCaIn BPG, die adaptive Trägheit basierend auf dem lokalen Verhalten der Funktion einbezieht. Außerdem beweisen wir die globale Konvergenz der von CoCaIn BPG erzeugten Sequenz zu einem kritischen Punkt der Funktion. CoCaIn BPG übertrifft BPG mit einem signifikanten Vorsprung, was auf die vorgeschlagene nicht-standardisierte Double-Backtracking-Technik zurückgeführt wird. Eine große Herausforderung bei der Arbeit mit BPG-basierten Methoden ist der Entwurf der Bregman-Distanz, die für das Problem geeignet ist. In diesem Zusammenhang schlagen wir Bregman-Distanzen vor, die für drei Anwendungen geeignet sind: Matrixfaktorisierung, tiefe Matrixfaktorisierung und tiefe neuronale Netze. Wir beginnen mit der Matrixfaktorisierung und schlagen die relevanten Bregman-Distanzen vor, dann gehen wir die tiefe Matrixfaktorisierung und tiefe neuronale Netzwerke an. In all diesen Anwendunge schlagen wir auch die Update-Schritte in geschlossener Form für BPG-basierte Methoden vor, was für die Praxis entscheidend ist. Wir schlagen auch die geschlossene Form der Trägheit vor, die für eine effiziente Anwendung von CoCaIn BPG geeignet ist. Bis hierhin ist die Einstellung jedoch auf additive zusammengesetzte Probleme beschränkt und generische zusammengesetzte Probleme, wie die Probleme, die bei der robusten Phasenrückgewinnung auftreten, sind außerhalb des Rahmens. Um generische zusammengesetzte Probleme angehen zu können, muss die L-smad-Eigenschaft noch weiter verallgemeinert werden. In diesem Zusammenhang schlagen wir die MAP-Eigenschaft vor, auf deren Basis wir den Modell-BPG-Algorithmus entwickeln. Die klassischen Techniken der Konvergenzanalyse, die auf dem Funktionswert basieren, erwiesen sich als einschränkend. Daher schlagen wir eine neuartige Lyapunov-Funktion vor, die für die globale Konvergenzanalyse geeignet ist. Später vereinen wir das Modell BPG und CoCaIn BPG, um das Modell CoCaIn BPG vorzuschlagen, für das wir die globalen Konvergenzergebnisse liefern. Wir ergänzen alle unsere theoretischen Ergebnisse durch relevante empirische Beobachtungen, um die konkurrenzfähige Leistung unserer Methoden im Vergleich zu bestehenden State-of-the-Art-Optimierungsmethoden zu zeigen.

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

Non-convex and non-smooth optimization is prevalent in many research fields such as Machine Learning, Data Science, Computer Vision, Statistics and many others (for example, see Chapters 5, 6, 7, 8, 9, 10). By non-convex, we mean not necessarily convex, and similarly by non-smooth, we mean not necessarily smooth. The non-smoothness and non-convexity can arise due to various factors of the problem structure, such as sparsity inducing function components, objectives enabling low rank structures, objectives enabling robust statistics, function components based on robust regularization, optimization over a constraint set and many others. Such factors in turn influence the design of optimization algorithms. Owing to the non-smoothness and the non-convexity, the design and the availability of suitable algorithms is challenging. The standard black box solvers developed for smooth optimization problems, such as Steepest Descent, Conjugate Gradient, L-BFGS, Newton's Method and many other algorithms are not suitable for non-smooth non-convex problems, in general. There exist few extensions of such algorithms which are also valid for certain non-smooth problems, however, such extensions are valid only in a restrictive setting. For example, BFGS is conducive for box constraints, however, its generalization to generic constraint sets is difficult.

Moreover, the objective functions that arise in the above-mentioned research fields are usually large in scale, as the datasets used in constructing the objectives are ever increasing in size. In order to optimize such objectives efficiently, the plausible algorithms must have computationally cheap update steps. In this regard, algorithms relying on line-search procedure are not preferable as each iteration can be computationally expensive (for example, several function evaluations might be required at each iteration). Also, methods relying on second-order information, such as Newton's method, are not suitable as the involved updates are either computationally expensive or numerically unstable. However, algorithms like Gradient Descent and some of its variants are preferable to tackle large-scale problems, as they rely on the first order information thus resulting in computationally cheap updates. Such first order methods are increasingly becoming popular (for example, see [13]). Notably, there exist several first order information based algorithms in the context of convex optimization. Such algorithms and their corresponding theoretical guarantees are not suitable for non-convex non-smooth problems, in general. However, it is possible to draw ideas from convex optimization to develop suitable first order algorithms while tackling several of the above-mentioned factors that influence the problem structure.

In order to achieve such a goal, it is important to detect the problem classes that enable a clear classification of the properties of the objective, which in turn can be leveraged to develop first order algorithms. Some popular problem classes include additive composite problem class (Chapter 5) and generic composite problem class (Chapters 9). Additive composite setting essentially deals with functions, where the objective function is made up of a non-smooth component and a smooth component. For example, objectives with a smooth data term and a non-smooth regularization term fall under additive composite setting. The generic composite problem setting involves objectives made up of a non-smooth function and a function which is a composition of two functions. It is often the case that each problem class is explored individually to develop appropriate algorithms. However, we discuss later in this thesis that it is possible to tackle both the problem classes and beyond in an unified manner.

In the context of additive composite setting, the development of many popular algorithms, such as Proximal Gradient Method (for example, see Chapter 2) and its inertial variant iPiano [137], relied on the so-called Lipschitz continuous gradient property (defined below) of the smooth component of the objective. Notably, many objectives that arise in practical applications have a Lipschitz continuous gradient. However, many contemporary research problems use objectives that do not have a Lipschitz continuous gradient. For example, objectives arising in matrix factorization, deep neural networks and many others do not have a Lipschitz continuous gradient. Moreover, Lipschitz continuous gradient property is not suitable for composite problem structures. This motivates various extensions of the Lipschitz continuous gradient property, which forms the main premise of this thesis. In particular, we explore various extensions of the Lipschitz continuous gradient property, develop related algorithms and provide their convergence analysis, while taking into consideration several of above-mentioned factors that influence the problem structure.

In this regard, we consider the optimization of non-convex and non-smooth objectives of the following form:

$$\inf_{x\in\mathbb{R}^N} f(x), \tag{1.1.1}$$

where $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper lower semicontinuous function and is lower bounded. We assume that the reader has some familiarity with the basics of real and convex analysis. As we will see in later chapters, many practical applications fall under this category, such as Poisson linear inverse problems (Chapter 9), phase retrieval problems (Chapter 5), matrix factorization problems (Chapter 6), deep matrix factorization (Chapter 7) problems and many others.

Our main objective here is to develop algorithms that optimize (1.1.1) with theoretical convergence guarantees. In order to achieve this goal, the function is required to have good structural properties. One such property is the Lipschitz continuous gradient property, which we recall below. For illustration purposes, let f be a continuously differentiable function over $\mathbb{R}^N$. The function f is said to be (classically) L-smooth (has L-Lipschitz continuous gradient), if there exists $L > 0$, such that for all $x, y \in \mathbb{R}^N$, we have

$$\|\nabla f(x) - \nabla f(y)\| \le L \|x - y\| \,. \tag{1.1.2}$$

The norm used in the above equation is the ℓ_2-norm, and we use the same notation for the rest of this thesis, unless specified otherwise. Simple one-dimensional functions like x^2, $\log(1+x^2)$ have a Lipschitz continuous gradient. Typical objectives arising in regularized least squares problems (for example, see [9]) have a Lipschitz continuous gradient. The setting in such problems involve $A \in \mathbb{R}^{M\times N}, b \in \mathbb{R}^M$, and an objective function $f : \mathbb{R}^N \to \mathbb{R}$ given by $f(x) = \frac{1}{2}\left\|Ax-b\right\|^2$. It is straightforward to see that the function f has a L-Lipschitz continuous gradient with $L = \left\|A^TA\right\|$.

A notable implication of (1.1.2) is the following Descent Lemma:

$$f(y) + \langle \nabla f(y), x-y\rangle - \frac{L}{2}\left\|x-y\right\|^2 \le f(x) \le f(y) + \langle \nabla f(y), x-y\rangle + \frac{L}{2}\left\|x-y\right\|^2 , \quad \forall\, x,y \in \mathbb{R}^N . \quad (1.1.3)$$

The upper bound of the Descent Lemma is also referred to as a quadratic convex majorant, whereas the lower bound is also referred to as a quadratic concave minorant. We illustrate the Descent Lemma in Figure 1.1, where the upper and lower quadratic bounds of the Descent Lemma are provided for a one-dimensional function $f(x) = x^2$.

The above-mentioned Descent Lemma (1.1.3) plays a crucial role in the convergence analysis of Gradient Descent, Proximal Gradient (PG) method and many others (see [13, 124]). For example, the update step involved in the Gradient Descent algorithm is essentially the minimizer of the upper bound in the Descent Lemma (1.1.3) at the current iterate, say $x_k \in \mathbb{R}^N$, as illustrated below

$$x_{k+1} = \mathrm{argmin}_{x\in\mathbb{R}^N}\left\{ f(x_k) + \langle \nabla f(x_k), x-x_k\rangle + \frac{L}{2}\left\|x-x_k\right\|^2 \right\} \quad \Leftrightarrow \quad x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k) . \quad (1.1.4)$$

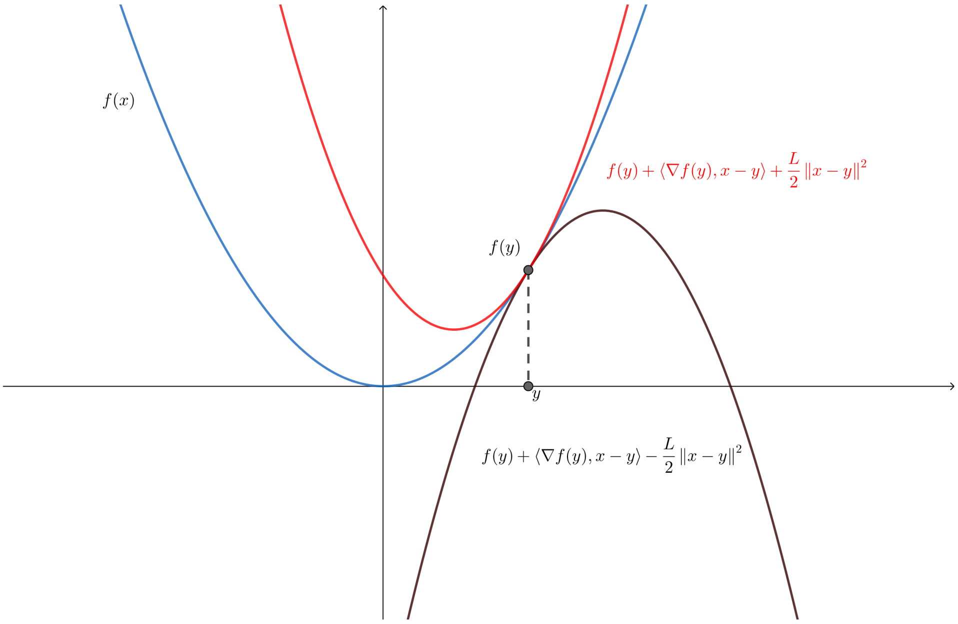


FIGURE 1.1: The inequalities in (1.1.3) guarantee that the objective function $f(x) = x^2$ has a quadratic concave minorant and a quadratic convex majorant at any $y \in \mathbb{R}$ with any $L \ge 2$.

Simple functions like $x^4, x^3, (x^2+y^2)^2, (1-xy)^2$ do not have a Lipschitz continuous gradient, thus the quadratic bounds in the Descent Lemma do not exist. Several objectives that arise in practice also do not have a Lipschitz continuous gradient, for example, the objectives that arise in phase retrieval, matrix factorization, deep neural networks and many others (see Chapter 4). This means that algorithms based on the Lipschitz continuous gradient property are not applicable. This motivates the extensions of the Lipschitz continuous gradient property which can be leveraged to produce algorithms with theoretical convergence guarantees. One such extension is the L-smad property [28].

The key is to observe that the condition (1.1.3) is equivalent to the convexity of $L\frac{\|\cdot\|^2}{2} - f$ and $L\frac{\|\cdot\|^2}{2} + f$. The L-smad property (Definition 4.4.1.1) deals with replacing the squared ℓ_2 norm with a so-called Legendre function (Definition 4.3.0.1), say h. For simplicity, we use a convex and a continuously differentiable h. The L-smad property states that a pair of functions (f, h) is L-smad on $\mathbb{R}^N$ if there exists a constant $L > 0$ such that the functions $Lh - f$ and $Lh + f$ are convex on $\mathbb{R}^N$. This eventually implies an Extended Descent Lemma given by

$$|f(x) - f(y) - \langle \nabla f(y), x - y\rangle| \leq LD_h(x, y)\,, \quad \forall\, x,\, y \in \mathbb{R}^N, \tag{1.1.5}$$

where $D_h(x, y)$ is the Bregman distance between the points x, y generated by h given by

$$D_h(x, y) = h(x) - h(y) - \langle \nabla h(y), x - y\rangle\ ,$$

and f is assumed to be continuously differentiable over $\mathbb{R}^N$. Notably, the bounds in the L-smad property need not be quadratic, and the bounds with higher order behavior can be incorporated via an appropriate Bregman distance. The precise setting will be explained later in Chapter 5. We illustrate the Extended Descent Lemma for a simple function x^4 in Figure 1.1, where the upper and lower bounds of the Extended Descent Lemma are considered at a point y. Due to choice of the Bregman distance, the bounds in Figure 1.1 are quartic and not quadratic.

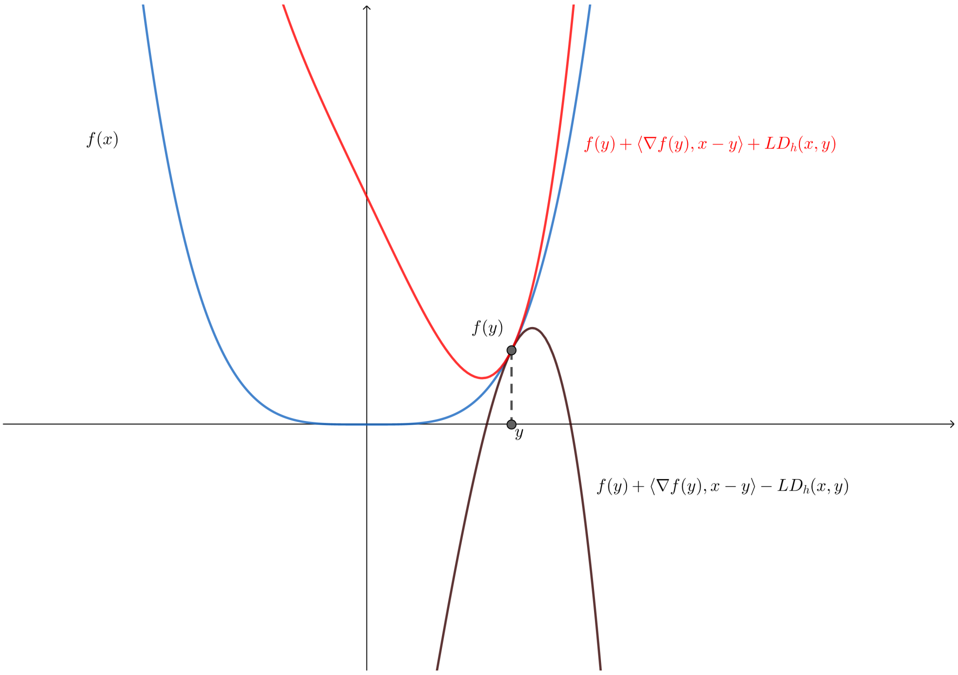


FIGURE 1.2: The inequalities in (1.1.5) guarantee that the objective function $f(x) = x^4$ has a concave minorant and a convex majorant that is not quadratic. Here, we use $h(x) = \frac{1}{4}x^4$ and $L \geq 4$ such that L-smad property holds true. Here, it is not possible to construct a quadratic majorant or a quadratic minorant at every $y \in \mathbb{R}$.

Based on this Extended Descent Lemma, the popular Bregman Proximal Gradient (BPG) algorithm was proposed in [28] with a global convergence guarantee. The update step of BPG is essentially the minimizer of the upper bound in (1.1.5), as illustrated below

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^N} \left\{ f(x_k) + \langle \nabla f(x_k), x - x_k \rangle + L D_h(x, x_k) \right\} .$$

Such an update step was classically considered in the so-called Mirror Descent (for example, see [14]) update step. BPG is more general and will be discussed in Chapter 4.

Equipped with the above-mentioned notions, in this thesis we aim to answer the following questions.

- The update step of BPG relies on the upper bound of the L-smad property. However, the significance of the lower bound is not clear. Thus, we ask the questions: What is the significance of the lower bound? What are its implications? How can it be leveraged in optimization algorithms? Can we achieve theoretical convergence guarantees to such algorithms?

- For illustrative purposes, we relied on the continuous differentiable property of f. In general, this need not hold. In this regard, we ask the questions: Is the L-smad property valid in general? If so, what class of problems is it valid for? If not, are there any extensions of the L-smad property? How can we leverage both the upper and lower bounds in such extensions in optimization algorithms? What guarantees do such algorithms have?

We describe our contributions below.

1.2 Overview

We broadly classify our work into two parts, namely,

- Part I: additive composite setting (Chapter 4 - 8),

- Part II: generic composite setting (Chapter 9 - 10).

1.2.1 Part I: additive composite setting

The problems of type (1.1.1) are difficult to tackle due to their generic nature. We first aim at solving a special case of the above problem, namely the additive composite problems given by

$$\inf_{x \in \mathbb{R}^N} f(x) , \quad f(x) := f_0(x) + f_1(x) , \tag{1.2.1}$$

where f_0 is a proper lower semicontinuous function and f_1 is a continuously differentiable function that satisfies certain favorable properties, which we will detail later. The separable nature of the function f can be exploited to develop the relevant algorithms. Many practical applications such as phase retrieval (Chapter 5), matrix factorization (Chapter 6), deep matrix factorization (Chapter 7) and many others fall under the category of additive composite problems. Thus, it is justified to explore these problems initially and we later consider the optimization problem in (1.1.1). We aim at providing first-order algorithms that are suitable for additive composite problems of type (1.2.1) based on the L-based property.

We recall the Bregman distance notion and several of its properties in Chapter 4. We also recall the L-based property in Chapter 4. In Chapter 5, we recall the popular BPG algorithm [28] which is based on the L-smad property. In the same chapter, we propose the CoCaIn BPG algorithm, which is an inertial variant of BPG. We note that BPG relies only on the upper bound for the function obtained via the L-smad property.

However, CoCaIn BPG makes use of both upper and lower bounds in the L-smad property in order to incorporate inertia, that is conducive for non-convex non-smooth problems. Both BPG and CoCaIn BPG rely on Bregman distances, which can be problem-dependent. Designing such Bregman distances is usually non-trivial and hard. In this regard, we tackle the following applications in the context of additive composite problems:

- standard phase retrieval (see Chapter 5),
- image denoising (see Chapter 5),
- matrix factorization (see Section 4.5 and Chapter 6),
- deep matrix factorization (see Section 4.6 and Chapter 7),
- deep neural networks (see Section 4.7, Section 4.8 and Chapter 8),
- Poisson linear problems (see Chapter 9).

Detailed discussion about the applications is provided below in Section 1.2.1.1. For most of the above mentioned applications, we propose Bregman distances that are suitable for the problem in Chapter 4, which in turn results in the applicability of BPG algorithms and their theoretical guarantees. In other cases, we use the previously proposed Bregman distances from the literature. For each of the application mentioned above, we provide relevant empirical illustrations by comparing BPG based algorithms with other state of the art algorithms and show the competitiveness of BPG based algorithms. For certain problems, we propose a variant of CoCaIn BPG, namely CoCaIn BPG CFI, where CFI stands for closed form inertia. We develop the theory required for CoCaIn BPG CFI, which involves obtaining the closed form solution for the inertia in the CoCaIn BPG algorithm.

1.2.1.1 Practical applications

We briefly detail here the above-mentioned practical applications.

Standard phase retrieval. In Chapter 5, 9, we consider the standard phase retrieval problem, a special case of the so-called quadratic inverse problems. Tackling the phase retrieval problem has been an active area of research in the recent years [40, 47, 64, 110, 164]. The setting involves certain sampling vectors $a_i \in \mathbb{R}^N$ and measurements $b_i > 0$, for $i = 1, 2, \ldots, M$. The goal is to find $x \in \mathbb{R}^N$, for which the following quadratic system of equations is approximately satisfied:

$$|\langle a_i, x\rangle|^2 \approx b_i^2, \quad \forall\, i = 1, 2, \ldots, M. \tag{1.2.2}$$

Such system of quadratic equations is solved through the following optimization problem:

$$\min_{x\in\mathbb{R}^N} \left\{ f(x) := \frac{1}{M}\sum_{i=1}^{M} (x^T A_i x - b_i)^2 + \mathcal{R}(x) \right\}, \tag{1.2.3}$$

where $\mathcal{R}(x)$ is the regularization term and $A_i = a_i a_i^T$, for $i = 1, 2, \ldots, M$. The Bregman distances that are suitable for this problem were initially considered in [28]. We use the same Bregman distances in order to apply BPG and CoCaIn BPG (see Chapter 5) for the above-mentioned phase retrieval problems.

Image denoising. In Chapter 5, we consider the problem of image denoising of a given possible noisy image $b \in \mathbb{R}^{M\times N}$, where $M, N \in \mathbb{N}$. The goal is to obtain the true image, denoted by $x \in \mathbb{R}^{M\times N}$. Such problems are popular in the context of image processing [36]. We need the following technical details to provide the full problem statement. The spatial finite difference operator is given by $(\mathcal{D}x)_{i,j} := \left((\mathcal{D}x)^1_{i,j}, (\mathcal{D}x)^2_{i,j}\right)$ where $i \in [M]$ and $j \in [N]$. The horizontal spatial finite differences are given by $(\mathcal{D}x)^1_{i,j} := x_{i+1,j} - x_{i,j}$ for all $i < M$ and 0 otherwise. The vertical spatial finite differences are given by $(\mathcal{D}x)^2_{i,j} := x_{i,j+1} - x_{i,j}$ for all $j < N$ and 0 otherwise. In the setting of (1.2.1), the problem here involves the following functions

$$f_0(x) := \sum_{i=1}^{M}\sum_{j=1}^{N} \log\left(1 + |x_{i,j} - b_{i,j}|\right) , \tag{1.2.4}$$

$$f_1(x) := \lambda \sum_{i=1}^{M}\sum_{j=1}^{N} \log\left(1 + \rho \left\|(\mathcal{D}x)_{i,j}\right\|_2^2\right) , \tag{1.2.5}$$

where $\lambda, \rho > 0$. The function f_0 is non-smooth non-convex and f_1 is smooth non-convex. The function f_1 is a non-convex variant of the popular Total Variation (TV) regularizer, which is used to prefer smooth signals while preserving sharp changes in the signal (such as edges of images). We show that the proposed variants of BPG outperform other state of the art optimization methods. We also illustrate that the denoised image obtained with our setting is much better compared to various other choices of f_0.

Matrix factorization. The matrix factorization problem is considered in Section 4.5 and Chapter 6. Matrix factorization has numerous applications in machine learning [112, 156], computer vision [48, 82, 157, 170], bio-informatics [35, 155] and many others. Here given a matrix $A \in \mathbb{R}^{M\times N}$, one is interested in the factors $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$ such that $A \approx UZ$ holds. This is usually cast into the following non-convex optimization problem

$$\min_{U\in\mathcal{U}, Z\in\mathcal{Z}} \left\{ f(U,Z) \equiv \frac{1}{2}\|A - UZ\|_F^2 + \mathcal{R}_1(U) + \mathcal{R}_2(Z) \right\} , \tag{1.2.6}$$

where $\mathcal{U}, \mathcal{Z}$ are constraint sets and $\mathcal{R}_1, \mathcal{R}_2$ are regularization terms. In Section 4.5, we propose the Bregman distances that are suitable for the matrix factorization problem such that the L-smad property holds. Using such Bregman distances, we propose further the BPG-MF and the CoCaIn BPG-MF algorithms in Chapter 6, which are actually the special cases of BPG and CoCaIn BPG for the matrix factorization setting. Moreover, we provide various pointers for efficient implementation of these algorithms and various empirical observations are provided to illustrate the superior performance of BPG methods over the classical alternating technique based methods (for example, PALM [26] or iPALM [144]).

Deep matrix factorization. In Section 4.6 and Chapter 7, we consider the following optimization problem

$$\min_{W_i\in\mathcal{W}_i, \forall i\in\{1,\ldots,N\}} \left\{ f(W) := \frac{1}{2}\|W_1W_2\cdots W_NX - Y\|_F^2 + \mathcal{R}(W) \right\} , \tag{1.2.7}$$

where N denotes the number of layers and $\mathcal{R}(W)$ is certain separable regularization term. Such problems arise in the context of deep matrix factorization or deep linear neural networks [77, 175]. We denote by $\mathcal{W}_i = \mathbb{R}^{d_i\times d_{i+1}}$ where $d_i \in \mathbb{N}$, for all $i \in \{1,\ldots,N\}$. Let $d_{N+1} = d$ and $X \in \mathbb{R}^{d\times n_T}$ be fixed, where $n_T \in \mathbb{N}$, which typically corresponds to the number of training samples. Similarly we have fixed $Y \in \mathbb{R}^{d_1\times n_T}$, which typically corresponds to the labels of the inputs in X. We denote by $W := (W_1, \ldots, W_N)$, meaning W lies in the product space $\mathcal{W} := \mathcal{W}_1 \times \cdots \times \mathcal{W}_N$, equipped with the norm $\|W\|_F^2 := \sum_{i=1}^N \|W_i\|_F^2$. We focus on $N \geq 2$ in this thesis. Deep matrix factorization model also has applications in matrix completion (c.f. Section 7.6). We develop Bregman distances suitable for this setting in Section 4.6. On deep matrix factorization problems, we provide empirical observations of BPG and CoCaIn BPG algorithms vs the alternating strategy based algorithms such as PALM [26] and iPALM [144], and also non-alternating strategy based algorithms such as forward–backward splitting with backtracking (FBS-WB) and iPiano with backtracking (iPiano-WB) [137]. Here, CoCaIn BPG is the best performing algorithm compared to other algorithms.

Deep neural networks. Deep neural networks has been an active area of research in the recent years [77], due to the state of the art performance on various artifical intelligence tasks [77, 96, 105, 154]. In this regard, we consider two problem settings, the regression setting and the classification setting. We start with the description of the regression setting. In Chapter 8, under the same notation as in deep matrix factorization, we consider the following optimization problem that arises in deep neural network training:

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in [N]} \left\{ f(W) := \frac{1}{2} \|\sigma_N(W_N \ldots \sigma_1(W_1 X)) - Y\|_F^2 + \mathcal{R}(W) \right\} , \tag{1.2.8}$$

where $\sigma_i : \mathbb{R}^N \to \mathbb{R}$, for $i = \{1, \ldots, N\}$ are the so-called activation functions (Definition B), that are smooth and $\mathcal{R}(W)$ is certain separable regularization term. The above given problem falls under the category of regression setting. We now describe the classification setting. Let K be the number of classes. Given a training dataset with M inputs, denoted $x_j \in \mathbb{R}^{d_1}$ for $j \in \{1, \ldots, M\}$, and the corresponding class j_k in $\{1, 2 \ldots, K\}$ for each input. Continuing the notation in the regression setting, x_j is the j^{th} column of X and set $K = d$, $M = n_T$. Here, the label for the j^{th} sample would be $y_j \in \mathbb{R}^N$, such that all the elements are zero except the j_k^{th} element which is set to one. The goal is find a model which uses this training dataset to predict the class labels for new unseen datapoints. In the classification setting, we consider the following objective:

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in [N]} \left\{ f(W) := \sum_{j=1}^M \left(-\log \left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^K e^{z_{j,k}}} \right) \right) + \mathcal{R}(W) \right\} . \tag{1.2.9}$$

where the vector $z_j \in \mathbb{R}^N$ is generated via certain deep neural network, which can be possibly be a linear network or a non-linear network for the j^{th} sample and z_{j,j_k} is the j_k^{th} coordinate of z_j, j_k denotes the class of j^{th} sample and it lies in $\{1, 2 \ldots, K\}$. For $j \in \{1, \ldots, N\}$, with deep linear neural networks we have $z_j = W_1 \ldots W_N x_j$, and with generic deep non-linear neural network we have $z_j := \sigma_N(W_N \ldots \sigma_1(W_1 x_j))$. For both the regression and the classification settings, we provide Bregman distances in Chapter 4 and closed form solutions for the update steps of BPG algorithms are provided in Chapter 8. We also provide few efficient ways to implement CoCaIn BPG in Chapter 8. Using real world datasets, we provide few empirical comparisons on BPG vs BPG-WB (BPG with backtracking) vs FBS-WB (forward–backward splitting with backtracking), and illustrate that BPG-WB is the best performing method.

Poisson linear inverse problems. Inverse problems under Poisson noise have various applications in nuclear medicine, electron microscopy and many others [18, 176]. For all $i = 1, \ldots, M$, let $b_i > 0$, $a_i \neq 0$ and $a_i \in \mathbb{R}_+^N$ be known. For any $x \in \mathbb{R}_+^N$, $\langle a_i, x \rangle > 0$ and $\sum_{i=1}^M (a_i)_j > 0$, for all $j = 1, \ldots, N$, $i = 1, \ldots, M$. The optimization problem involved in Poisson linear inverse problems is:

$$\min_{x\in\mathbb{R}_+}\left\{f(x) := \sum_{i=1}^{M}\left(\langle a_i, x\rangle - b_i\log(\langle a_i, x\rangle)\right) + \phi(x)\right\}, \tag{1.2.10}$$

where ϕ is the regularizing function, which is potentially non-convex. For this problem, the suitable Bregman distances were already developed in previous works such as [10, Lemma 7]. We use their ideas to illustrate the applicability of Model BPG algorithm. We provide empirical comparisons of Model BPG variants vs IBPM-LS [139].

In all of the above mentioned problems, we illustrate the numerical competitiveness of our methods based on the BPG.

1.2.2 Part II: generic composite setting

Additive composite setting can be restrictive and certain practical applications such as robust phase retrieval (Chapter 9) and many other problems are out of the scope. Thus, we consider the optimization of generic non-smooth non-convex problems of type (1.1.1) in Chapter 9, 10. Firstly, note that the continuous differentiability of f required in the L-smad property is restrictive. Simple functions like $\left|x^4 - 1\right|$ and many objectives that arise in practice are not continuously differentiable. Thus, more general notion compared to the L-smad property is sought. Based on the initial work in [55], we propose an extension of L-smad property, which we call as MAP property in Chapter 9. MAP property relies on the notion of so-called model function (see Definition 9.3.0.2) which serves as a local function approximation satisfying certain growth property. Based on the model function notion and the MAP property, we propose an extension of the BPG algorithm, which we call Model BPG. This unifies several algorithms that are suitable for several generic composite problems and additive composite problems, for example, Proximal Gradient Method [13, Chapter 10], Bregman Proximal Gradient [28], Prox-Linear Method [43, 62] and many others. The convergence analysis of Model BPG is non-trivial and requires the development of new theoretical tools. In this regard, we propose a novel Lyapunov function as a measure of progress for the Model BPG algorithm and provide the global convergence analysis for the sequence generated by Model BPG. We later provide an inertial variant of Model BPG that relies on the same strategy as CoCaIn BPG, which we call Model CoCaIn BPG. Based on similar theoretical tools of Model BPG, we provide the convergence analysis for Model CoCaIn BPG.

In the context of generic composite problems, we consider the following applications:

- standard phase retrieval (see Chapter 9),
- robust phase retrieval (see Chapter 9, 10).

1.2.2.1 Practical applications

We briefly detail here the above-mentioned practical applications.

Standard phase retrieval. We have provided a brief explanation of the standard phase retrieval problem in Section 1.2.1.1. In Chapter 9, we use the same objective and rewrite it such that it falls under generic composite problem setting. Here, we illustrate the applicability of Model BPG and its variants.

Robust phase retrieval. Considering the same notation as in the standard phase retrieval problem provided above, the robust phase retrieval problem involves the following optimization problem :

$$\min_{x\in\mathbb{R}^N}\left\{ f(x) := \frac{1}{M}\sum_{i=1}^{M}\left|x^T A_i x - b_i\right| + \mathcal{R}(x)\right\} ,$$

where $\mathcal{R}(x)$ is the regularization term. Such a setting was recently explored in [56, 64]. We note that the above-mentioned problem falls under generic composite problem category considered in Chapter 9, 10. For this case, we illustrate the applicability of Model BPG in Chapter 9 and Model CoCaIn BPG in Chapter 10.

1.3 Publications

Some chapters in this thesis comprises the content of some journal publications, conference publications and pre-prints which we list below. The Chapter 9 comprises the content of a paper that is in review for a journal. The content in Sections 4.7 and 4.8, Chapters 8 and 10 is neither published anywhere nor in review.

The Chapter 5 comprises the content of the following publication ([121]):

- M. C. Mukkamala, P. Ochs, T. Pock, and S. Sabach. Convex-Concave backtracking for inertial Bregman Proximal Gradient algorithms in non-convex optimization. SIAM Journal on Mathematics of Data Science, 2(3):658–682, 2020.

The Chapter 9 is comprises the content of the following pre-print ([118]) that is in review for a journal:

- M. C. Mukkamala, J. Fadili, and P. Ochs. Global convergence of model function based Bregman proximal minimization algorithms. arXiv preprint arXiv:2012.13161, 2020.

The Section 4.5 and Chapter 6 comprises the content of the following publication ([120]):

- M. C. Mukkamala and P. Ochs. Beyond alternating updates for matrix factorization with inertial Bregman proximal gradient algorithms. In Advances in Neural Information Processing Systems, pages 4266–4276, 2019.

The Section 4.6 and Chapter 7 comprises the content of the following pre-print ([122]):

- M. C. Mukkamala, F. Westerkamp, E. Laude, D. Cremers, and P. Ochs. Bregman proximal framework for deep linear neural networks. arXiv preprint arXiv:1910.03638, 2019.

A condensed version of the above pre-print is published and the publication details ([123]) are given below.

- M. C. Mukkamala, F. Westerkamp, E. Laude, D. Cremers, and P. Ochs. Bregman proximal gradient algorithms for deep matrix factorization. In Scale Space and Variational Methods in Computer Vision, pages 204-215, 2021.

Chapter 2

Convex analysis

In this chapter we briefly explain the popular notions from the convex analysis and the convex optimization fields. Note that this chapter is not my contribution and we only list the results that are detailed in [19, 150].

2.1 Affine sets

The line between two points $x_1, x_2 \in \mathbb{R}^N$ such that $x_1 \neq x_2$ is a collection of points of the following form:

$$\theta x_1 + (1-\theta)x_2 \,,$$

where $\theta \in \mathbb{R}$. The set of such points for $\theta \in [0,1]$ form a line segment. A set $C \subset \mathbb{R}^N$ is an affine set, if for any two points $x, y \in C$ and $\theta \in \mathbb{R}$, the point $\theta x + (1-\theta)y$ lies in C. Intuitively, this means that the line generated using any two distinct points in C lies in C, if the set C is an affine set. It is possible to define affine sets using more than two points. Firstly, we need to define the notion of affine combination. An affine combination of the points $x_1, \ldots, x_k$ in $\mathbb{R}^N$ is given by $\theta_1 x_1 + \ldots + \theta_k x_k$, where $\theta_1 + \ldots + \theta_k = 1$ and $\theta_i \in \mathbb{R}$, for $i \in \{1, \ldots, k\}$. A standard induction argument reveals that an affine set contains all the affine combinations of its points. Let $A \in \mathbb{R}^{M \times N}, b \in \mathbb{R}^M$, then the set of points given by $C := \{x \in \mathbb{R}^N \mid Ax = b\}$ is an affine set. Note that C is the solution set to the linear system of equations $Ax = b$.

Consider any set $C \subset \mathbb{R}^N$. The affine hull of the set C, denoted $\operatorname{aff} C$, is given by

$$\operatorname{aff} C := \{\theta_1 x_1 + \ldots + \theta_k x_k \,|\, x_1, \ldots, x_k \in C,\, \theta_1, \ldots, \theta_k \in \mathbb{R},\, \theta_1 + \ldots + \theta_k = 1\}\,.$$

The affine dimension of a set $C \subset \mathbb{R}^N$ is the dimension of its affine hull. This can possibly be different from the standard notion of the dimension of a set. For example, consider the set $C := \{(x_1, x_2) \in \mathbb{R}^2 \,|\, x_1^2 + x_2^2 = 1\}$, which is a unit circle. Usually, C is said to have dimension one. However, its affine dimension is two, as $\operatorname{aff} C = \mathbb{R}^2$. Consider a set $C \subset \mathbb{R}^N$, then a point $y \in \mathbb{R}^N$ is said to be an interior point of C, if there exists $r > 0$ such that $B(x, r) \subset C$ where $B(x, r) := \{y \in \mathbb{R}^N \mid \|x - y\| \leq r\}$. The set of all the interior points of C is said to be the interior of C, denoted as $\operatorname{int} C$. Many concepts in optimization rely on the notion of interior, however, interior is a restrictive notion as there can exist sets for which interior is empty and a more general notion is sought. Relative interior serves this purpose and it is given by

$$\operatorname{ri} C = \{x \in C : B(x, r) \cap \operatorname{aff} C \subset C, \text{ for certain } r \geq 0\}\,.$$

The relative boundary of a set $C \subset \mathbb{R}^N$ is given by $\operatorname{cl} C \setminus \operatorname{ri} C$, where $\operatorname{cl} C$ denotes the closure of C. As an example for relative interior and relative boundary, we record below example given in [33, Example 2.2].

Consider the following set

$$C := \{x \in \mathbb{R}^3 \,|\, -1 \leq x_1 \leq 1,\, -1 \leq x_2 \leq 1,\, x_3 = 0\}\,.$$

Then, we have $\operatorname{aff} C = \{x \in \mathbb{R}^3 \,|\, x_3 = 0\}$. Note that the interior of the set C is empty, however, its relative interior is nonempty and is given by

$$\operatorname{relint} C := \{x \in \mathbb{R}^3 \,|\, -1 < x_1 < 1,\, -1 < x_2 < 1,\, x_3 = 0\}\,.$$

The relative boundary is given by

$$\{x \in \mathbb{R}^3 \mid \max\{|x_1|, |x_2|\} = 1,\, x_3 = 0\}\,.$$

2.2 Convex sets

A set $C \subset \mathbb{R}^N$ is said to be convex if $\lambda x + (1 - \lambda) y \in C$, whenever $x, y \in C$ and $0 \leq \lambda \leq 1$. In other words, C is convex if the line segment between any two points in C also lies in C. An example of convex set is $B(x, r)$, for some $x \in C$ and $r > 0$. It is straightforward to see that all affine sets are convex. In the same spirit as the notion of affine combination, the convex combination of the points $x_1, \ldots, x_k \in C$ for $C \subset \mathbb{R}^N$ is given by $\theta_1 x_1 + \ldots + \theta_k x_k$ where $\theta_1 + \ldots + \theta_k = 1$ and $\theta_1, \ldots, \theta_k \geq 0$. The convex hull of C is the collection of all the convex combinations of the points in C, given by

$$\operatorname{conv} C = \{\theta_1 x_1 + \ldots + \theta_k x_k \,|\, x_1, \ldots, x_k \in C,\, \theta_1 + \ldots + \theta_k = 1 \text{ and } \theta_1, \ldots, \theta_k \geq 0\}\,.$$

Note that $\operatorname{conv} C$ is convex and $\operatorname{conv} C$ is the smallest convex set that contains C. Extensions of convex combination of infinite terms are possible (see [33, Section 2.1.4]). Some examples of convex sets are given below.

- The Euclidean ball given by $B(x, r) := \{x \in \mathbb{R}^N \mid \|x - x_0\|_2 \leq r\}$ is convex for any $x_0 \in \mathbb{R}^N$.

- The empty set $\emptyset$ is convex.
- The half-space given by $\{x \in \mathbb{R}^N \,|\, Ax \leq b\}$ for some $A \in \mathbb{R}^{M \times N}$, $b \in \mathbb{R}^M$.

Various examples of convex sets can be found in [33, Section 2.1.4].

2.2.1 Operations preserving convexity of sets

In order to determine the convexity of sets, it is useful to explore the properties of convex sets, in particular the operations that preserve the convexity. Such operations can be useful to generate more convex sets from given convex sets. The intersection of convex sets is convex, i.e., $\cap_{i \in I} C_i$ of any collection $\{C_i \,|\, i \in I\}$ of convex sets is convex, where I is an index set. The cardinality of I can even be infinite. The vector sum $C_1 + C_2$ of two convex sets C_1 and C_2 is convex. Convexity is preserved under scaling and translation, i.e., λC and $C + a$ are convex for any convex set $C \subset \mathbb{R}^N$, scalar $\lambda \in \mathbb{R}$ and $a \in \mathbb{R}^N$. The image and inverse image of a convex set under a affine function are convex. Several other operations that preserve convexity can be found in [33, Section 2.3].

2.2.2 Properties of convex sets

If C is a convex set in $\mathbb{R}^N$ and λ_1, λ_2 are positive scalars, then the following holds:

$$(\lambda_1 + \lambda_2)C = \lambda_1 C + \lambda_2 C\,.$$

We now recall few statements from [148, Chapter 6]. Let $x \in \operatorname{ri} C$ and $y \in \operatorname{cl} C$. Then $(1-\lambda)x + \lambda y$ belongs to $\operatorname{ri} C$ for $0 \leq \lambda < 1$. Also, $\operatorname{cl}(\operatorname{ri} C) = \operatorname{cl} C$ and $\operatorname{ri}(\operatorname{cl} C) = \operatorname{ri} C$. The closure and the relative interior of C are convex while having the same affine hull as C. For various other properties of convex sets, we refer the reader to [148, Chapter 6].

2.3 Convex functions

For a function $f : \mathbb{R}^N \to \mathbb{R}$, we denote $\operatorname{dom} f$ to be a subset in $\mathbb{R}^N$ on which f is defined. We record the following definition of a convex function from [33, Section 3.1.1]. A function $f : \mathbb{R}^N \to \mathbb{R}$ is said to be convex, if $\operatorname{dom} f$ is convex, and if for any $x, y \in \operatorname{dom} f$, $\lambda \in [0, 1]$ the following holds:

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)\,.$$

A convex function f is strictly convex if the above condition holds strictly when $x \neq y$ and $\lambda \in (0, 1)$. Some of the examples of convex functions are provided below. Here, we refer to [33, Section 3.1.4, Section 3.1.5].

- The function $f : \mathbb{R}^N \to \mathbb{R}$ given by $f(x) = \frac{1}{2}x^T Q x + q^T x + r$ is convex, where Q is a positive semi-definite matrix, $q \in \mathbb{R}^N$ and $r \in \mathbb{R}$.
- The function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = e^{ax}$ is convex for $a \in \mathbb{R}$.
- The function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = x^a$ is convex on $\mathbb{R}_{++}$ when $a \geq 1$ or $a \leq 0$.
- The function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = |x|^p$ is convex on $\mathbb{R}$ when $p \geq 1$.
- The function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = -\log(x)$ is convex on $\mathbb{R}_{++}$.

Note that the convexity of $\operatorname{dom} f$ is embedded into the definition of a convex function f provided above. It is often convenient to work with a definition without explicitly stating the requirement on the domain. In this regard, extended real-value functions are used, wherein a function f is extended to all of $\mathbb{R}^N$ by assigning the function value to $+\infty$ outside $\operatorname{dom} f$. Thus, an extended real-valued function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is convex if it satisfies the following property:

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y)\,, \qquad \text{for all } x, y \in \mathbb{R}^N,\ \lambda \in [0,1]\,,$$

where $\overline{\mathbb{R}} := [-\infty, +\infty]$ is the extended real line, which we define shortly. The domain of f is then redefined as

$$\operatorname{dom} f := \{x \in \mathbb{R}^N \,|\, f(x) < \infty\}\,.$$

The extended real valued system is defined as $\overline{\mathbb{R}} := [-\infty, +\infty]$ and entails the usual arithmetic operations and the following additional operations:

- $0(\pm\infty) = (\pm\infty)0 = 0$;
- for $a \in \mathbb{R}$, the condition $a \pm \infty = \pm\infty$ holds;
- $(-\infty) + (-\infty) = (-\infty)$, $(+\infty) + (+\infty) = (+\infty)$, $(+\infty) + (-\infty) = (-\infty) + (+\infty) = (+\infty)$;
- for $a \in (0, +\infty]$, the condition $a(\pm\infty) = (\pm\infty)$ holds;
- for $a \in [-\infty, 0)$, the condition $a(\pm\infty) = (\mp\infty)$ holds;
- the expression $(\pm\infty)/(\pm\infty)$ is undefined.

The extended real line $\overline{\mathbb{R}}$ has all the standard properties of a compact set, including the existence of a supremum (least upper bound) and an infimum (greatest lower bound) with possibly infinite values for every subset in $\overline{\mathbb{R}}$. For an empty set, the following entities are standard $\inf \emptyset = +\infty$, $\sup \emptyset = -\infty$, thus $\inf \emptyset > \sup \emptyset$. For simplicity, we denote $+\infty$ as ∞. Note that the notion of extended real-valued function is suitable for convex functions as well as for generic functions which we will focus in Chapter 3.

For a function $f : \mathbb{R}^N \to \mathbb{R}$, the graph of the function is given by

$$\text{Graph} f := \{(x, \alpha) \in \mathbb{R}^{N+1} \,|\, \alpha = f(x)\}\,.$$

The epigraph of a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is defined by

$$\text{epi} f := \{(x, y) \,|\, f(x) \leq y,\ x \in \mathbb{R}^N,\ y \in \mathbb{R}\}\,.$$

A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$

- is convex if $\text{epi} f$ is a convex set;
- is closed if $\text{epi} f$ is a closed set;
- is said to be a proper function if it attains $f(x) < \infty$ for atleast one $x \in \mathbb{R}^N$, and f does not attain $-\infty$ function value;
- is said to be a lower semi-continuous function at $\bar{x} \in \mathbb{R}^N$ if $f(\bar{x}) \leq \liminf_{x \to \bar{x}} f(x)\,;$

- is said to be a lower semi-continuous function if it is a lower semi-continuous function at each point in $\mathbb{R}^N$;

For a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$, with $\alpha \in \mathbb{R}$ the α-level set is defined by

$$\mathrm{lev}_{\leq \alpha} f := \{x \in \mathbb{R}^N \mid f(x) \leq \alpha\} .$$

For a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ the following notions are equivalent, f is closed, f is a lower semi-continuous function and $\mathrm{lev}_{\leq \alpha} f$ is closed for any $\alpha \in \mathbb{R}$.

We now provide an example of an extended real-valued function that is convex. Consider a closed convex set C, then the indicator function $\delta_C : \mathbb{R}^N \to \overline{\mathbb{R}}$ is given by

$$\delta_C(x) = \begin{cases} 0 , & x \in C , \\ \infty , & x \notin C , \end{cases} \tag{2.3.1}$$

is a convex function. A comprehensive list of examples of convex functions is provided in [33, Section 3.1.5].

2.3.1 Operations preserving convexity of functions

Detection of convexity of functions is an important practical problem. In general, verifying the definition of convexity can be cumbersome. Thus, it is preferable to have certain standard operations that preserve convexity, which can also be used to generate new convex functions. These operations can also help in detecting convexity of a function from its components. Some of operations under which convexity is stable are provided below.

- Weighted sum of convex functions with non-negative weights is convex. This is also true for infinite sums and integrals.
- Convexity is preserved under composition with an affine mapping, i.e., if f is convex on $\mathbb{R}^N$, then $f(Ax + b)$ is convex, where $A \in \mathbb{R}^{N\times M}$, $b \in \mathbb{R}^N$ and $x \in \mathbb{R}^M$.
- Convexity is preserved under pointwise supremum or maximum of any family of convex functions.

For detailed description of operations and additional operations that preserve convexity, we refer the reader to [33, Section 3.2].

2.3.2 Convexity tests

We now provide some criterion for testing the convexity of a function. One could directly verify the definition of a convex function. However, for certain functions with some nice properties, additional criterion can be used for ease of checking convexity.

Consider a differentiable function $f : C \to \mathbb{R}$, where $C \subset \mathbb{R}^N$ is an open set. Then, f is convex if and only $\mathrm{dom}\, f \; (:= C)$ is convex and

$$f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle \ , \quad \text{holds for all } x, y \in C. \tag{2.3.2}$$

In order to test for strict convexity, the above inequality is replaced by strict inequality, i.e.,

$$f(y) > f(x) + \langle \nabla f(x), y - x\rangle \ , \quad \text{holds for all } x, y \in C,$$

and the rest of the conditions remain the same. The property in (2.3.2) is remarkable, because for any fixed $x \in C$, the affine function $f(x) + \langle \nabla f(x), \cdot - x\rangle$ is a global underestimator the function. Thus, only by using the local information, certain global information can be obtained.

Consider a twice differentiable function $f : C \to \mathbb{R}$, such that for $\nabla^2 f(x)$ exists for all x in C, which is open. Then, f is convex if and only $\operatorname{dom} f$ $(:= C)$ is convex and $\nabla^2 f(x) \succeq 0$, for all $x \in C$. If $\nabla^2 f(x) \succ 0$ for all $x \in C$, then f is strictly convex, however, the converse is not true (for example, x^4 is strictly convex with $\nabla^2 f(0) = 0$).

2.3.3 Conjugate function

We now consider the conjugate function notion. Consider an extended real-valued function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$. Then, the conjugate of the function $f^* : \mathbb{R}^N \to \overline{\mathbb{R}}$ is defined by

$$f^*(y) := \sup_{x \in \mathbb{R}^N} \langle x, y\rangle - f(x)\,.$$

Conjugate function is also called as Legendre transformation, Legendre-Fenchel transform, or Fenchel conjugate. Intuitively, for a given (slope) y, the conjugate function $f^*(y)$ provides the minimum shift that is required to place the affine function $\langle x, y\rangle$ such that it is exactly below the graph of the function f. A remarkable property of the conjugate function is that f^* is a closed convex function, even if f is not a convex function.

2.3.4 Subgradient and subdifferential

Let $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ be a proper convex function. Let $x \in \operatorname{dom} f$ and the $v \in \mathbb{R}^N$ is said to be a subgradient of f at x if we have

$$f(y) \geq f(x) + \langle v, y - x\rangle \;, \quad \text{for all } y \in \mathbb{R}^N.$$

Collection of all subgradients of f at $x \in \mathbb{R}^N$ is called subdifferential of f at x, and it is given by

$$\partial f(x) = \{v \in \mathbb{R}^N \mid f(y) \geq f(x) + \langle v, y - x\rangle \text{ for all } y \in \mathbb{R}^N\}\,.$$

For a point x such that $x \notin \operatorname{dom} f$, we set $\partial f(x) = \emptyset$. If f is differentiable at $x \in \operatorname{dom} f$, then $\partial f(x) = \{\nabla f(x)\}$. Subdifferentials are very helpful in characterizing the set of minimizers. Consider a proper convex function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$, then x^* lies in the set of minimizers $\operatorname{argmin}_{x \in \mathbb{R}^N} f(x)$ if and only if $0 \in \partial f(x)$. This property is known as Fermat's rule or Fermat's optimality condition.

2.4 Convex optimization

Convex optimization is a vast topic in itself. In order to keep the presentation in line with rest of the chapters, here we will focus on a prominent algorithm, Proximal Gradient Method. We require few technical details before presenting the Proximal Gradient Method.

2.4.1 Lipschitz continuous gradient

We briefly extend the discussion regarding Lipschitz continuous gradient, given in Chapter 1. If f is twice continuously differentiable with a L-Lipschitz continuous gradient, then we have $\nabla^2 f(x) \preceq LI$ for all $x \in \mathbb{R}^N$. Instead of considering the full space $\mathbb{R}^N$ in the definition of L-smoothness, we now consider a more general variant that provides the notion of L-smoothness over a set. Consider a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ such that over a set $C \subset \operatorname{int}\operatorname{dom} f$ it is continuously differentiable. Then, f is said to be L-smooth over C (or has L-Lipschitz continuous gradient over C) if it satisfies the following condition:

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\| \,, \quad \forall x, y \in C \,.$$

As a consequence of the above definition, if the set C is convex, we have the following Descent Lemma [13, Lemma 5.7]:

$$|f(x) - f(y) - \langle \nabla f(y), x - y\rangle| \leq \frac{L}{2} \|x - y\|^2 \,, \text{ for any } x, y \in C \,.$$

2.4.2 Proximal Gradient Method

In order to present several algorithms in convex optimization in a unified manner, we focus on the Proximal Gradient Method. For the description of Proximal Gradient Methods we rely on [13, Chapter 10].

Consider the following optimization problem

$$\min_{x \in \mathbb{R}^N} \{f(x) := f_0(x) + f_1(x)\} \,,$$

where we assume that $f_0 : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper closed convex function, $f_1 : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper, closed function with convex domain $\operatorname{dom} f_1$, $\operatorname{dom} f_0 \subset \operatorname{int}\operatorname{dom} f_1$ and f_1 has a L-Lipschitz continuous gradient over $\operatorname{int}\operatorname{dom} f_1$. Additionally, we also assume that the set of minimizers $\operatorname{Argmin}_{x \in \mathbb{R}^N} f(x)$ is non-empty, and thus f is lower bounded, which we denote by f^*. In such a setting, the Proximal Gradient Method is given in Algorithm 1.

Algorithm 1: PGM: Proximal Gradient Method

- **Initialization:** Select $x_0 \in \operatorname{dom} f$.
- **For each** $k \geq 1$**:** Choose τ_k such that $\tau_k \in (0, 2/L)$ and compute

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^N} \left\{ f_0(x) + f_1(x_k) + \langle \nabla f_1(x_k), x - x_k\rangle + \frac{1}{2\tau_k} \|x - x_k\|^2 \right\} . \tag{2.4.1}$$

The update step in PGM basically involves the quadratic majorizer of f_1 and the function f_0 remains as it is, at the point x_k. As per [13, Lemma 10.4], PGM has the following descent property

$$f(x_{k+1}) \leq f(x_k) - \left(\frac{1}{\tau_k} - \frac{L}{2}\right) \|x_{k+1} - x_k\|^2 \,.$$

The above result implies that the function value is monotonically nonincreasing with the sequence generated by PGM. As per [13, Theorem 10.15], all the limit points of the sequence generated by PGM are stationary points of f. Under certain additional conditions, one can also show global convergence of the sequence generated by PGM [7], in the sense that the whole sequence converges to a single point that is a stationary point of f (for example, see [7]). Note that we do not assume convexity of f till here.

If f_1 is convex, then PGM attains the convergence rate of $O(1/k)$ in terms of function values to the optimal value f^*. In precise terms, set $\tau_k = \frac{1}{L}$, then for any $x^* \in \operatorname{Argmin}_{x\in\mathbb{R}^N} f$, the following condition holds true:

$$f(x_k) - f^* \leq \frac{L\left\|x_0 - x^*\right\|}{2k}\,. \tag{2.4.2}$$

If f_0 is an indicator function δ_C over a nonempty closed convex set $C \subset \mathbb{R}^N$, then Proximal Gradient Method reduces to Projected Gradient Descent. If f_0 is a zero function, then the Proximal Gradient Method reduces to the so-called Gradient Descent with the update $x_{k+1} = x_k - \tau_k \nabla f_1(x_k)$. If $f_0(x) = \lambda \left\|x\right\|_1$ for some $\lambda > 0$, then Proximal Gradient Method reduces to Iterative Shrinkage-Thresholding Algorithm (ISTA).

2.4.3 Backtracking

A major drawback of PGM is the requirement to know the value of L, which can be difficult to compute, in general. Even if L is known, it is often very large, thus resulting in small steps for PGM. However, one can locally estimate the value of L via the quadratic bounds in (1.1.3). The upper bound in (1.1.3) governs the step-size, thus at each iteration of PGD one can modify the step-size to incorporate local constant $\bar{L}_k > 0$ such that the following condition holds:

$$f_1(x_{k+1}) \leq f_1(x_k) + \langle \nabla f_1(x_k), x_{k+1} - x_k\rangle + \frac{\bar{L}_k}{2}\left\|x - x_k\right\|^2\,.$$

In particular, the step-size can be chosen such that it is non-increasing, which can be chosen, for example, by setting $\tau_k \leq \min\left\{\tau_{k-1}, \bar{L}_k^{-1}\right\}$. The full algorithm, Proximal Gradient Method with backtracking is provided in Algorithm 2.

Algorithm 2: PGM-WB: Proximal Gradient Method with Backtracking

- **Initialization:** Select $x_0 \in \operatorname{dom} f$ and $\bar{L}_0 > 0$.
- **For each $k \geq 1$:** Choose $\bar{L}_k \geq \bar{L}_{k-1}$, set $\tau_k \leq \min\left\{\tau_{k-1}, \bar{L}_k^{-1}\right\}$ and compute

$$x_{k+1} = \operatorname{argmin}_{x\in\mathbb{R}^N}\left\{f_0(x) + f_1(x_k) + \langle \nabla f_1(x_k), x - x_k\rangle + \frac{1}{2\tau_k}\left\|x - x_k\right\|^2\right\} \tag{2.4.3}$$

with $\bar{L}_k$ fulfilling

$$f_1(x_{k+1}) \leq f_1(x_k) + \langle \nabla f_1(x_k), x_{k+1} - x_k\rangle + \frac{\bar{L}_k}{2}\left\|x - x_k\right\|^2\,. \tag{2.4.4}$$

In order to find an appropriate $\bar{L}_k$ such that (2.4.4) holds true, one can use the backtracking procedure, which we describe now. In backtracking procedure, to find $\bar{L}_k$ such that the condition (2.4.4) is satisfied, start with an estimate for $\bar{L}_k > 0$ and if the condition (2.4.4) fails to hold, then increase the estimate of $\bar{L}_k$ by a scaling factor greater than one. By repetitively scaling the estimate, the condition (2.4.4) holds true for certain appropriate $\bar{L}_k$ after finite number of repetitions as the function f_1 has a Lipschitz continuous gradient. In order to obtain the condition on step-size such that the step-size is decreasing, we set $\tau_k \leq \min\left\{\tau_{k-1}, \bar{L}_k^{-1}\right\}$ and $\bar{L}_k \geq \bar{L}_{k-1}$. A similar variant was provided in [13, Chapter 10], for which similar convergence results as in (2.4.2) are obtained and there is no change in convergence rate when f_1 is assumed to be convex.

2.4.4 Accelerated Proximal Gradient Method

In the context of convex f_1, PGM acheives $O(1/k)$ convergence rate. It is possible to obtain a better rate $O(1/k^2)$ in terms of function values. In this regard, we provide Accelerated Proximal Gradient Method (APGM) in Algorithm 3. It is also referred to as "fast proximal method" or "FISTA". The strategy used relies on the classical Nesterov's Accelerated Gradient Method [126]. The idea is to develop an extrapolated point (2.4.6) before performing PGM like update. The extrapolated point is governed by an inertial parameter γ_k and a careful choice of the step-size and the inertial parameter leads to an accelerated convergence rate $O(1/k^2)$ in terms of function values.

Algorithm 3: APGM: Accelerated Proximal Gradient Method

- **Initialization:** Select $y_0 = x_0 \in \operatorname{dom} f$ and $t_0 = 1$.
- **For each** $k \geq 1$**:** Choose $\gamma_k \in [0,1]$ and compute

$$x_{k+1} \in \operatorname*{Argmin}_{x\in\mathbb{R}^N} \left\{ f_0(x) + f_1(y_k) + \langle \nabla f_1(y_k), x - y_k\rangle + \frac{L}{2}\left\|x - y_k\right\|^2 \right\}. \tag{2.4.5}$$

- Set $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$, $\gamma_k = \frac{t_k - 1}{t_{k+1}}$ and compute

$$y_k = x_k + \gamma_k(x_k - x_{k-1}). \tag{2.4.6}$$

If f_1 is convex such that $\operatorname{dom} f_1 = \mathbb{R}^N$ then for any $x^* \in \operatorname{Argmin}_{x\in\mathbb{R}^N} f$, APGM attains the following convergence rate:

$$f(x_k) - f^* \leq \frac{2L\left\|x_0 - x^*\right\|}{(k+1)^2}.$$

Note that PGM has only $O(1/k)$ convergence rate, whereas APGM attains $O(1/k^2)$ convergence rate, hence the term "accelerated". Moreover, the increase in computational effort in APGM compared to PGM is minimal. For a comprehensive review of PGM, APGM and related variants, we refer the reader to [13, Chapter 10]. The choice of the inertial parameter can be relaxed in the non-convex setting significantly, which we consider in Chapter 5.

2.4.5 Strong convexity

We recall the notion of a strongly convex function. Consider a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$. It is said to be strongly convex with modulus/parameter $\mu > 0$, or equivalently μ-strongly convex for certain $\mu > 0$, if $\operatorname{dom} f$ is convex and for any $x, y \in \operatorname{dom} f$, $\lambda \in [0,1]$ the following condition holds:

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y) - \frac{\mu}{2}\lambda(1-\lambda)\left\|x-y\right\|^2.$$

Another equivalent definition is the following. A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is said to be strongly convex with modulus $\mu > 0$, if $f - (\mu/2)\left\|\cdot\right\|^2$ is convex. Note that the above definitions of strong convexity do not require the differentiability of the function. We now provide variants of the strong convexity definition, when f is differentiable.

If f is a continuously differentiable convex function over $\mathbb{R}^N$, we can use the following definition. The function f is strongly convex with modulus $\mu > 0$, if for all $x, y \in \mathbb{R}^N$ we have

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{\mu}{2} \|x - y\|^2 .$$

Intuitively, this means that there exists a quadratic lower bound for f which can be generated using the local information at any point. If f is a twice continuously differentiable μ-strongly convex function, we can equivalently say that for all $x \in \mathbb{R}^N$ the condition $\nabla^2 f(x) \succeq \mu I$ holds. Clearly, simple functions like $x^2, x^4 + x^2$ are strongly convex. Let $A \in \mathbb{R}^{M \times N}, b \in \mathbb{R}^M$, then the function $f : \mathbb{R}^N \to \mathbb{R}$ given by $f(x) = \frac{1}{2} \|Ax - b\|^2$ is strongly convex if $\lambda_{\min}(A^T A) > 0$. Usually, when the function is strongly convex, one can obtain stronger convergence guarantees such as linear convergence rate for PGM and APGM variants [13, 124], which we skip discussing for brevity.

Chapter 3

Variational analysis

3.1 Variational analysis

For this chapter, we mainly refer to [150] and sparsely we refer also to [116]. For an extended real-valued function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$, it is of interest to know whether f attains its minimum and the properties of minimizers, if any. It is well-known that a continuous function defined on a compact set attains its minimum (Weierstrass Extreme Value Theorem). In the context of extended real-valued functions, can a similar statement be made? Before we answer this question, we require the following notion. A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is level-bounded if for any $\alpha \in \mathbb{R}$ the level set $\mathrm{lev}_{\leq\alpha} f$ is bounded. Here, $\mathrm{lev}_{\leq\alpha} f$ can possibly be empty. Note that if $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is level-bounded then $f(x) \to \infty$ as $\|x\| \to \infty$. Now, we answer the question regarding attainment of minimum for an extended real-valued function with the following theorem.

Theorem 3.1.0.1. *[150, Theorem 1.9] Let $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ be a lower semi-continuous, level-bounded and a proper function, then* $\inf f$ *is finite and* $\operatorname{argmin} f$ *is nonempty and compact.*

In order to attain the minimum of a function, the iterative algorithms that use the local first-order information are popular. For a differentiable function, the gradient of the function provides the local first-order information. For a convex function, the subgradient of the function provides with such information. However, for a generic extended real-valued function, the notion of subgradient has to be defined. Firstly, we require the following notion. The sequence $(x^\nu)_{\nu\in\mathbb{N}}$ is said to be f-attentive convergent if $x^\nu \to x$ and $f(x^\nu) \to f(x)$, which is denoted by $x^\nu \underset{f}{\to} x$.

3.2 Subgradients and subdifferentials

Regular subdifferential. We record the definition of subgradients from [150, Definition 8.3]. Consider a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ and a point $\bar{x}$ with $f(\bar{x})$ finite. A vector $v \in \mathbb{R}^N$ is a regular subgradient of f at $\bar{x}$, written $v \in \widehat{\partial} f(\bar{x})$, if

$$f(x) \geq f(\bar{x}) + \langle v, x - \bar{x}\rangle + o(\|x - \bar{x}\|)\,, \tag{3.2.1}$$

or equivalently

$$\liminf_{x\to\bar{x}, x\neq\bar{x}} \frac{f(x) - f(\bar{x} + \langle v, x - \bar{x}, \rangle)}{\|x - \bar{x}\|} \geq 0. \tag{3.2.2}$$

The set $\widehat{\partial} f(\bar{x})$ is called a regular subdifferential (or presubdifferential or Fréchet subdifferential or viscosity subdifferential). The regular subdifferential is equal to $\{\nabla f(\bar{x})\}$ when f is differentiable at $\bar{x}$. For a convex function f, the usual subdifferential and regular subdifferential coincide. We are interested in non-smooth non-convex problems, for which the regular subdifferential can possibly be empty (for example, when $f(x) = -|x|$ at $x = 0$ then $\widehat{\partial} f(\bar{x}) = \emptyset$). The regular subdifferential is a convex set.

Failure of calculus with regular subdifferential. Certain desirable properties such as the standard calculus, the closedness property do not hold for the regular subdifferential. For example, the sum rule of subdifferential $\widehat{\partial}(f_0 + f_1)(x) \subset \widehat{\partial} f_0(x) + \widehat{\partial} f_1(x)$ may fail to hold for certain $f_0 : \mathbb{R}^N \to \overline{\mathbb{R}}$, $f_1 : \mathbb{R}^N \to \overline{\mathbb{R}}$ (for $f_1(x) = -|x|$, $f_0(x) = |x|$ at $x = 0$, $\widehat{\partial}(f_0 + f_1)(x) = \{0\}$ whereas $\widehat{\partial} f_0(x) + \widehat{\partial} f_1(x) = \emptyset$). Another desirable property in optimization is the closedness property of subdifferential, where if there is a sequence $(x^\nu)_{\nu\in\mathbb{N}}$ such that $x^\nu \underset{f}{\to} x$ then the sequence $(v^\nu)_{\nu\in\mathbb{N}}$ for $v^\nu \in \widehat{\partial} f(x^\nu)$ is expected to converge to $v \in \widehat{\partial} f(x)$, however this may not hold for the regular subdifferential. Thus, a generalization of regular subdifferential known as general subdifferential was proposed.

Limiting subdifferential. A vector $v \in \mathbb{R}^N$ is a (general) subgradient of f at $\bar{x}$ with finite $f(\bar{x})$, written $v \in \partial f(\bar{x})$, if there are sequences $x^\nu \underset{f}{\to} \bar{x}$, $v^\nu \in \widehat{\partial} f(x^\nu)$ and $v^\nu \to v$. The set $\partial f(\bar{x})$ is called general or limiting subdifferential (also called as Mordukhovich subdifferential). Limiting subdifferential can be possibly non-convex and the standard calculus is applicable. Also, the set $\partial f(\bar{x})$ satisfies the closedness property by definition, and the condition $\widehat{\partial} f(\bar{x}) \subset \partial f(\bar{x})$ holds true.

Critical points. Equipped with the notion of limiting subdifferential, the set of critical points of a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is defined by

$$\mathrm{crit} f := \{x \in \mathbb{R}^N \,|\, 0 \in \partial f(x)\}\,.$$

Fermat's rule. We record now the Fermat's rule for extended real-valued function (see [134, Theorem 4.23]). For a proper function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$, if the point $\bar{x}$ is a local minimum then $0 \in \partial f(\bar{x})$. As a consequence, if $f = f_0 + g$ for certain smooth function f_0, then $0 \in \partial f(\bar{x})$ is equivalent to the condition $-\nabla f_0(x) \in \partial g(\bar{x})$ (see [134, Corallary 4.24]).

Horizon subdifferential. In addition to the regular subdifferential and the limiting subdifferential, we also consider the notion of horizon subdifferential. For an extended real-valued function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ at certain point there can exist certain directions such that function can get infinitely steep or possibly has a jump in function value. Such directions are characterized by the horizon subdifferential, which we define now. A vector $v \in \mathbb{R}^N$ is a horizon subgradient of f at $\bar{x}$ with finite $f(\bar{x})$, written $v \in \partial^\infty f(\bar{x})$, if there are sequences $x^\nu \xrightarrow[f]{} \bar{x}$, $v^\nu \in \widehat{\partial} f(x^\nu)$, one has $\lambda^\nu v^\nu \to v$ for some sequence $\lambda^\nu \searrow 0$. The set of all horizon subgradients $\partial^\infty f(\bar{x})$ is called horizon subdifferential.

Consequences. Based on the above defined notions, we have the following results from [150, Exercise 8.8]. If f_0 is differentiable at $\bar{x}$, then $\widehat{\partial} f(\bar{x}) = \{\nabla f_0(\bar{x})\}$, so $\nabla f_0(\bar{x}) \in \partial f_0(\bar{x})$. If f_0 is smooth on a neighborhood of $\bar{x}$, then $\partial f(\bar{x}) = \{\nabla f_0(\bar{x})\}$ and $\partial^\infty f(\bar{x}) = \{\nabla f_0(\bar{x})\}$. If $f = g + f_0$ with g finite at $\bar{x}$ and f_0 is smooth on a neighborhood of $\bar{x}$, then $\widehat{\partial} f(\bar{x}) = \widehat{\partial} g(\bar{x}) + \nabla f_0(\bar{x})$, $\partial f(\bar{x}) = \partial g(\bar{x}) + \nabla f_0(\bar{x})$, and $\partial^\infty f(\bar{x}) = \partial^\infty g(\bar{x})$.

Convex functions. In the context of convex functions, based on [150, Proposition 8.12], we have the following notions. For any proper, convex function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ and any point $\bar{x} \in \operatorname{dom} f$, one has

$$\begin{aligned}
\partial f(\bar{x}) &= \{v \,|\, f(x) \geq f(\bar{x}) + \langle v, x - \bar{x}\rangle \text{ for all } x\} = \widehat{\partial} f(\bar{x}) \,, \\
\partial^\infty f(\bar{x}) &\subset \{v \,|\, 0 \geq \langle v, x - \bar{x}\rangle \text{ for all } x \in \operatorname{dom} f\} = N_{\operatorname{dom} f}(\bar{x}) \,.
\end{aligned}$$

The horizon subgradient inclusion is an equation when f is locally lsc at $\bar{x}$ or when $\partial f(\bar{x}) \neq \emptyset$.

Subderivative. We consider the following definition of subderivative from [150, Definition 8.1]. For a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ and a point $\bar{x}$ with $f(\bar{x})$ finite, the subderivative function $df(\bar{x}) : \mathbb{R}^N \to \overline{\mathbb{R}}$ is defined by

$$df(\bar{x})(\bar{w}) := \liminf_{\substack{\tau \searrow 0 \\ w \to \bar{w}}} \frac{f(\bar{x} + \tau w) - f(\bar{x})}{\tau} \,.$$

The subderivative given above is actually the lower subderivative. If lim inf is replaced by lim sup, then one obtains the upper subderivative. For a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ and a point $\bar{x}$ with $f(\bar{x})$ finite, the regular subderivative function $df(\bar{x}) : \mathbb{R}^N \to \overline{\mathbb{R}}$ is defined by

$$\widehat{d}f(\bar{x})(\bar{w}) := \lim_{\delta \searrow 0} \left(\limsup_{\substack{x \xrightarrow[f]{} \bar{x} \\ \tau \searrow 0}} \left(\inf_{w \in B(\bar{w}, \delta)} \frac{f(x + \tau w) - f(x)}{\tau} \right) \right) \,,$$

where $B(\bar{w}, \delta) := \{w \in \mathbb{R}^N \mid \|w - \bar{w}\| \leq \delta\}$.

3.3 Set convergence

The limiting subdifferential and horizon subdifferential can be seen as certain evolutions of the sets pertaining to the regular subdifferential. However, in order to consider the evolution or the sequence of sets, we need to define the set convergence concepts. We require the following notations:

$$\begin{aligned}\mathcal{N}_\infty &:= \{N \subset \mathbb{N} \quad | \quad \mathbb{N}\backslash N \text{ is finite }\}\,,\\ \mathcal{N}_\infty^\# &:= \{N \subset \mathbb{N} \quad | \quad N \text{ is infinite }\}\,.\end{aligned}$$

Clearly, $\mathcal{N}_\infty \subset \mathcal{N}_\infty^\#$. We denote $\lim_{\nu\to\infty}$ when $\nu \to \infty$ and $\nu \in \mathbb{N}$ and $\lim_{\nu \underset{N}{\to} \infty}$ for convergence of a subsequence based on the index set N in $\mathcal{N}_\infty$ and $\mathcal{N}_\infty^\#$. Based on these notations, we require the following notions related to convergence of sets, namely, the outer limit and inner limit. Consider a sequence of sets $(C^\nu)_{\nu\in\mathbb{N}}$ in $\mathbb{R}^N$, then its outer limit is defined by

$$\limsup_{\nu\to\infty} C^\nu := \{x \in \mathbb{R}^N \,|\, \exists N \in \mathcal{N}_\infty^\#,\, \exists x^\nu \in C^\nu(\nu \in \mathbb{N}) \text{ with } x^\nu \underset{N}{\to} x\}\,, \tag{3.3.1}$$

and its inner limit is defined by

$$\liminf_{\nu\to\infty} C^\nu := \{x \in \mathbb{R}^N \,|\, \exists N \in \mathcal{N}_\infty,\, \exists x^\nu \in C^\nu(\nu \in \mathbb{N}) \text{ with } x^\nu \underset{N}{\to} x\}\,. \tag{3.3.2}$$

The limit of the sequence $(C^\nu)_{\nu\in\mathbb{N}}$ exists if the inner limit and the outer limit are equal, that is

$$\lim_{\nu\to\infty} C^\nu := \limsup_{\nu\to\infty} C^\nu := \liminf_{\nu\to\infty} C^\nu\,. \tag{3.3.3}$$

The outer limit and the inner limit of a sequence $(C^\nu)_{\nu\in\mathbb{N}}$ always exist and can possibly be empty. However, the limit of a sequence $(C^\nu)_{\nu\in\mathbb{N}}$ may not exist.

3.3.1 Subdifferentials with set convergence

Consider an extended real-valued function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ that is finite at $\bar{x} \in \mathbb{R}^N$. Then, equipped with the notions of outer and inner limits, the limiting subdifferential can be equivalently defined as following:

$$\partial f(\bar{x}) := \limsup_{x \underset{f}{\to} \bar{x}} \widehat{\partial} f(x)\,.$$

Similarly, the horizon subdifferential is expressed as

$$\partial^\infty f(\bar{x}) = \limsup_{\substack{x \underset{f}{\to} \bar{x} \\ \lambda^\nu \searrow 0}} \lambda^\nu \widehat{\partial} f(x)\,,$$

where

$$\limsup_{\substack{x \underset{f}{\to} \bar{x} \\ \lambda^\nu \searrow 0}} \widehat{\partial} f(x) = \{v \in \mathbb{R}^N \,|\, \exists\, x^\nu \to \bar{x}, f(x^\nu) \to f(x), \lambda^\nu \searrow 0, v^\nu \to v \text{ with } v^\nu \in \widehat{\partial} f(x^\nu) \text{ for all } \nu \in \mathbb{N}\}.$$

3.4 Normal cone and tangent cone

Another equivalent characterization of subdifferentials can be obtained using the notion of normal cones. A related notion to the normal cone is the tangent cone, thus we discuss both the concepts in conjunction.

Tangent cone. We list below few important notions from [150, Chapter 6]. A sequence $x^\nu \to \bar{x}$ is said to converge from the direction dir w if for some sequence of scalars $\tau^\nu \searrow 0$ the vectors $\frac{x^\nu - \bar{x}}{\tau^\nu}$ converge to w. We denote $x^\nu \underset{C}{\to} x$ if $x^\nu \to \bar{x}$ with $x^\nu \in C$. A vector $w \in \mathbb{R}^N$ is tangent to a set $C \subset \mathbb{R}^N$ at a point $\bar{x} \in C$, written $w \in T_C(\bar{x})$, if

$$\frac{x^\nu - \bar{x}}{\tau^\nu} \to w \quad \text{for some } x^\nu \underset{C}{\to} \bar{x},\, \tau^\nu \searrow 0\,.$$

The set of all the tangent vectors at $\bar{x}$ is the tangent cone $T_C(\bar{x})$.

Normal cone. Firstly, we define the normal vectors whose collection represents the normal cone. We record the definition as in [150, Definition 6.3]. Let $C \subset \mathbb{R}^N$ and $\bar{x} \in C$. A vector v is normal to C at $\bar{x}$ in the regular sense, or a regular normal, written $v \in \widehat{N}_C(\bar{x})$, if

$$\langle v, x - \bar{x}\rangle \leq o(\|x - \bar{x}\|) \quad \text{for } x \in C\,.$$

It is normal to C at $\bar{x}$ in the general sense, or simply a normal vector, written $v \in N_C(\bar{x})$, if there are sequences $x^\nu \underset{C}{\to} \bar{x}$ and $v^\nu \to v$ with $v^\nu \in \widehat{N}_C(x^\nu)$. As per [150, Proposition 6.5], we deduce that $N_C(\bar{x}), \widehat{N}_C(\bar{x})$ are closed cones. Also, $\widehat{N}_C(\bar{x})$ is convex and can be related to the tangent cone via the following equivalence:

$$v \in \widehat{N}_C(\bar{x}) \quad \Longleftrightarrow \quad \langle v, w\rangle \leq 0 \text{ for all } w \in T_C(\bar{x}).$$

Additionally, $N_C(\bar{x}) = \limsup\limits_{x \underset{C}{\to} \bar{x}} N_C(x) \supset N_C(\bar{x})$.

We record below the following crucial results on normal and tangent cones.

Proposition 3.4.0.1. *[150, Proposition 6.41] With $\mathbb{R}^N$ expressed as $\mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_m}$, write $x \in \mathbb{R}^N$ as $(x_1, \ldots, x_m)$ with components $x_i \in \mathbb{R}^{n_i}$. If $C = C_1 \times \ldots \times C_m$ for closed sets $C_i \in \mathbb{R}^{n_i}$, then at any $\bar{x} = (\bar{x}_1, \ldots, \bar{x}_m)$ with $\bar{x}_i \in C_i$ one has*

$$N_C(\bar{x}) = N_{C_1}(\bar{x}_1) \times \ldots \times N_{C_m}(\bar{x}_m),$$

$$\widehat{N}_C(\bar{x}) = \widehat{N}_{C_1}(\bar{x}_1) \times \ldots \times \widehat{N}_{C_m}(\bar{x}_m),$$

$$T_C(\bar{x}) \subset T_{C_1}(\bar{x}_1) \times \ldots \times T_{C_m}(\bar{x}_m),$$

$$\widehat{T}_C(\bar{x}) = \widehat{T}_{C_1}(\bar{x}_1) \times \ldots \times \widehat{T}_{C_m}(\bar{x}_m).$$

Furthermore, C is regular at $\bar{x}$ if and only if each C_i is regular at $\bar{x}_i$. In the regular case the inclusion for $T_C(\bar{x})$ becomes an equation like the others.

Proposition 3.4.0.2. *[150, Theorem 6.42] Let $C = C_1 \cap \ldots \cap C_m$ for closed sets $C_i \subset \mathbb{R}^N$, and let $\bar{x} \in C$. Then*

$$T_C(\bar{x}) \subset T_{C_1}(\bar{x}) \cap \ldots \cap T_{C_m}(\bar{x}),$$

$$\widehat{N}_C(\bar{x}) \supset \widehat{N}_{C_1}(\bar{x}) + \ldots + \widehat{N}_{C_m}(\bar{x}).$$

Under the condition that the only combination of vectors $v_i \in N_{C_i}(\bar{x})$ with $v_1 + \ldots + v_m = 0$ is $v_i = 0$ for all i (this being satisfied for $m = 2$ when C_1 and C_2 are convex and cannot be seperated), one also has

$$\widehat{T}_C(\bar{x}) \supset \widehat{T}_{C_1}(\bar{x}) \cap \ldots \cap \widehat{T}_{C_m}(\bar{x}),$$

$$N_C(\bar{x}) \subset N_{C_1}(\bar{x}) + \ldots + N_{C_m}(\bar{x}).$$

If in addition every C_i is regular at $\bar{x}$, then C is regular at $\bar{x}$ and

$$T_C(\bar{x}) = T_{C_1}(\bar{x}) \cap \ldots \cap T_{C_m}(\bar{x}),$$

$$N_C(\bar{x}) = N_{C_1}(\bar{x}) + \ldots + N_{C_m}(\bar{x}).$$

Clarke regularity. Another crucial notion is the Clarke regularity of sets, which we record from [150, Definition 6.4]. A set $C \subset \mathbb{R}^N$ is regular at one of its points $\bar{x}$ in the sense of Clarke if it is locally closed at $\bar{x}$ and every normal vector to C at $\bar{x}$ is a regular normal vector, i.e., $N_C(\bar{x}) = \widehat{N}_C(\bar{x})$.

We now record the notion of subdifferential regularity from [150, Definition 7.25]. A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is called subdifferentially regular at $\bar{x}$ if $f(\bar{x})$ is finite and epi$f$ is Clarke regular at $(\bar{x}, f(\bar{x}))$ as a subset of $\mathbb{R} \times \mathbb{R}$.

Optimality conditions in constrained optimization. Normal cones and tangent cones play an important role in optimization. For example, they characterize the optimality conditions in constrained optimization problems. In this regard, we record [150, Theorem 6.12]. Consider the problem of minimizing a differentiable function f_0 over a set $C \subset \mathbb{R}^N$. A necessary condition for $\bar{x}$ to be locally optimal is

$$\langle \nabla f_0(\bar{x}), w \rangle \geq 0 \text{ for all } w \in T_C(\bar{x})\,,$$

which is the same as $-\nabla f_0(\bar{x}) \in \widehat{N}_C(\bar{x})$ and implies

$$-\nabla f_0(\bar{x}) \in N_C(\bar{x}), \text{ or } 0 \in \nabla f_0(\bar{x}) + N_C(\bar{x}).$$

When C is convex, the above given conditions are equivalent and can also written as

$$\langle \nabla f_0(\bar{x}), x - \bar{x} \rangle \geq 0 \text{ for all } x \in C\,,$$

which means that the linearized function $f_0(\bar{x}) + \langle \nabla f_0(\bar{x}), \cdot - \bar{x} \rangle$ achieves its minimum over C at $\bar{x}$. When f_0 too is convex, the equivalent conditions are sufficient for $\bar{x}$ to be globally optimal.

Relation between subdifferentials and normal cones. The subgradients are closely are closely related to the normal cones generated from the epigraphs. We record below the relations as in [150, Theorem 8.9]. For a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ and a point $\bar{x}$ with $f(\bar{x})$ finite, we have

$$\begin{aligned}
\widehat{\partial} f(\bar{x}) &= \{ v \,|\, (v, -1) \in \widehat{N}_{\mathrm{epi}f}(\bar{x}, f(\bar{x})) \}\,, \\
\partial f(\bar{x}) &= \{ v \,|\, (v, -1) \in N_{\mathrm{epi}f}(\bar{x}, f(\bar{x})) \}\,, \\
\partial^\infty f(\bar{x}) &\subset \{ v \,|\, (v, 0) \in N_{\mathrm{epi}f}(\bar{x}, f(\bar{x})) \}\,.
\end{aligned}$$

The last relation holds with equality if f is locally lower semicontinuous at $\bar{x}$.

3.5 Lipschitz continuity and strict continuity

We record here the definitions of Lipschitz continuity and strict continuity given in [150, Definition 9.1]. Let F be a single-valued mapping defined on a set $D \subset \mathbb{R}^N$, with values in $\mathbb{R}^M$. Let $X \subset D$. F is Lipschitz continuous on X if there exists $\kappa \in \mathbb{R}_+ = [0, \infty)$ with

$$\left\|F(x') - F(x)\right\| \leq \kappa \left\|x' - x\right\| \qquad \text{for all } x, x' \in X.$$

Then, κ is called a Lipschitz constant for F on X. F is strictly continuous at $\bar{x}$ relative to X if $\bar{x} \in X$ and the value

$$\operatorname{lip}_X F(\bar{x}) := \limsup_{\substack{x, x' \xrightarrow[X]{} \bar{x} \\ x \neq x'}} \frac{\|F(x') - F(x)\|}{\|x' - x\|}$$

is finite. More simply, F is strictly continuous at $\bar{x}$, where $\operatorname{lip}_X F(\bar{x})$ is this modulus relative to X. F is strictly continuous relative to X if, for every point $\bar{x} \in X$, F is strictly continuous at $\bar{x}$ relative to X.

The horizon subdifferential characterizes the strict continuity property. From [150, Theorem 9.13], we have that for a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ that is locally lower semicontinuous at $\bar{x}$ with finite $f(\bar{x})$, the conditions $\partial^\infty f(\bar{x}) = \{0\}$ and the f being strictly continuous at $\bar{x}$ are equivalent.

3.5.1 Coercivity

The following is the standard definition (see [152, Definition 1.2.33]) of coercive and super-coercive functions. A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is called

- coercive if $\lim_{\|x\| \to \infty} f(x) = +\infty$,
- supercoercive if $\lim_{\|x\| \to \infty} \frac{f(x)}{\|x\|} = +\infty$.

Consider the two dimensional problem $f(x, y) := |a - xy|^2$, where $a \in \mathbb{R}$, $x \in \mathbb{R}$, $y \in \mathbb{R}$. Note that f is not coercive, because when $y = 0$ and $|x| \to \infty$, then $\sqrt{x^2 + y^2} \to \infty$ however f stays constant at $|a|^2$. However, a slight modification of the function given by $f(x, y) = |a - xy|^2 + \lambda x^2 + \lambda y^2$ results in coercivity, due to the additional quadratic term. Moreover, it is easy to see that f is supercoercive. Some examples of coercive functions that are super-coercive include $\|x\|_2^2$, $x^T A x$ for some symmetric positive definite matrix A. An example of a coercive function that is not super-coercive is $\|x\|_1$.

3.6 Subdifferentials based on the function structure

Depending on the structure of the function, the rules for the calculation of the subdifferentials can be simplified significantly. In this regard, we consider various structures, such as separable functions, composite functions, additive functions.

3.6.1 Results on separable functions

The following result pertains to the functions that have separability in the variables. In particular, the following result illustrates the construction of the subdifferentials when the subdifferentials of the components are known.

Proposition 3.6.1.1 (Proposition 10.5 in [150]). *Let $f(x) = f_1(x_1) + \ldots + f_m(x_m)$ for lsc functions $f_i : \mathbb{R}^{n_i} \to \overline{\mathbb{R}}$, where $x \in \mathbb{R}^N$ is expressed as $(x_1, \ldots, x_m)$ with $x_i \in \mathbb{R}^{n_i}$. Then at any $\bar{x} = (\bar{x}_1, \ldots, \bar{x}_m)$ with $f(\bar{x})$ finite and $df_i(\bar{x}_i)(0) = 0$, one has*

$$\begin{aligned} \widehat{\partial} f(\bar{x}) &= \widehat{\partial} f_1(\bar{x}_1) \times \ldots \times \widehat{\partial} f_m(\bar{x}_m), \\ \partial f(\bar{x}) &= \partial f_1(\bar{x}_1) \times \ldots \times \partial f_m(\bar{x}_m), \\ \partial^\infty f(\bar{x}) &= \partial^\infty f_1(\bar{x}_1) \times \ldots \times \partial^\infty f_m(\bar{x}_m), \end{aligned}$$

while on the other hand

$$\begin{aligned} df(\bar{x}) &\geq df_1(\bar{x}_1) + \ldots + df_m(\bar{x}_m)\,, \\ \widehat{d}f(\bar{x}) &\leq \widehat{d}f_1(\bar{x}_1) + \ldots + \widehat{d}f_m(\bar{x}_m)\,. \end{aligned}$$

Moroever, f is regular at $\bar{x}$ when f_i is regular at $\bar{x}_i$ for each i. Then the inclusions and inequalities become equations.

3.6.2 Results on additive functions

The following result pertains to the functions that comprises of additive components, in the sense that the function can be expressed as a summation of few other functions. In such a case, using the subdifferentials of the component functions, we can deduce the following result.

Corollary 3.6.2.1 (Corollary 10.9 in [150]). *Suppose $f = f_1 + \ldots + f_m$ for proper, lsc functions $f_i : \mathbb{R}^N \to \overline{\mathbb{R}}$, and let $\bar{x} \in \operatorname{dom} f$. Then*

$$\begin{aligned} \widehat{\partial} f(\bar{x}) &\supset \widehat{\partial} f_1(\bar{x}) + \ldots + \widehat{\partial} f_m(\bar{x}), \\ df(\bar{x}) &\geq df_1(\bar{x}) + \ldots + df_m(\bar{x}). \end{aligned}$$

Under the condition that the only combination of vectors $v_i \in \partial^\infty f_i(\bar{x})$ with $v_1 + \ldots + v_m = 0$ is $v_1 = v_2 = \ldots = v_m = 0$ (this being true in the case of convex functions f_1, f_2 when $\operatorname{dom} f_1$ and $\operatorname{dom} f_2$ cannot be separated), one also has that

$$\begin{aligned} \partial f(x) &\subset \partial f_1(x) + \ldots + \partial f_m(x)\,, \\ \partial^\infty f(x) &\subset \partial^\infty f_1(x) + \ldots + \partial^\infty f_m(x)\,, \\ \widehat{d}f(\bar{x}) &\leq \widehat{d}f_1(\bar{x}) + \ldots + \widehat{d}f_m(\bar{x})\,. \end{aligned}$$

If also each f_i is regular at $\bar{x}$, then f is regular at $\bar{x}$ and

$$\begin{aligned} \partial f(x) &= \partial f_1(x) + \ldots + \partial f_m(x)\,, \\ \partial^\infty f(x) &= \partial^\infty f_1(x) + \ldots + \partial^\infty f_m(x)\,, \\ \widehat{d}f(\bar{x}) &= \widehat{d}f_1(\bar{x}) + \ldots + \widehat{d}f_m(\bar{x})\,. \end{aligned}$$

In the above provided Corollary 3.6.2.1, the requirement that the only combination of vectors $v_i \in \partial^\infty f_i(\bar{x})$ with $v_1 + \ldots + v_m = 0$ is $v_1 = v_2 = \ldots = v_m = 0$ is also known as the qualification condition.

3.6.3 Chain rule

It is often the case that the functions have a composite structure, and in such a case one is interested in the chain rule or rules through which the subdifferentials of a function can be obtained. In this regard, we consider the following result.

Theorem 3.6.3.1 (Theorem 10.6 in [150]). *Suppose $f(x) = g(F(x))$ for a proper, lsc function $g : \mathbb{R}^M \to \overline{\mathbb{R}}$ and a smooth mapping $F : \mathbb{R}^N \to \mathbb{R}^M$. Then at any point $\bar{x} \in \operatorname{dom} f = F^{-1}(\operatorname{dom} g)$ one has*

$$\begin{aligned} \widehat{\partial} f(\bar{x}) &\supset \nabla F(\bar{x})^* \widehat{\partial} g(F(\bar{x})), \\ df(\bar{x})(w) &\geq dg(F(\bar{x}))(\nabla F(\bar{x})w). \end{aligned}$$

If the only vector $y \in \partial^\infty g(F(\bar{x}))$ with $\nabla F(\bar{x})^* y = 0$ is $y = 0$ (this being true for convex g when $\operatorname{dom} g$ cannot be separated from the range of the linearized mapping $w \to F(\bar{x}) + \nabla F(\bar{x})w$) one also has

$$\begin{aligned} \partial f(\bar{x}) \subset \nabla F(\bar{x})^* \partial g(F(\bar{x})), \quad \partial^\infty f(\bar{x}) \subset \nabla F(\bar{x})^* \partial^\infty g(F(\bar{x})), \\ \widehat{d} f(\bar{x})(w) \leq \widehat{d} g(F(\bar{x}))(\nabla F(\bar{x})w). \end{aligned}$$

If in addition g is regular at $F(\bar{x})$, then f is regular at $\bar{x}$ and

$$\begin{aligned} \partial f(\bar{x}) = \nabla F(\bar{x})^* \partial g(F(\bar{x})), \quad \partial^\infty f(\bar{x}) = \nabla F(\bar{x})^* \partial^\infty g(F(\bar{x})), \\ df(\bar{x})(w) = dg(F(\bar{x}))(\nabla F(\bar{x})w). \end{aligned}$$

Theorem 3.6.3.1 is a very generic result and the result stated in Corollary 3.6.2.1 can be seen as a special case of the above given theorem (see proof of Corollary 10.9 in [150]). There are many important consequences of Theorem 3.6.3.1 which can be found in [150, Chapter 10, Section B].

3.6.4 Results on parametric functions

One of the consequences of Theorem 3.6.3.1 is the chain rule result pertaining to the partial subdifferentiation. At times the function can have dependency on two (or more) variables. In such a case, one might be interested to know the subdifferentials with respect to one variable. The following result considers such a setting.

Corollary 3.6.4.1 (Corollary 10.11 in [150]). *For a proper, lsc function $f : \mathbb{R}^N \times \mathbb{R}^M \to \overline{\mathbb{R}}$ and a point $(\bar{x}, \bar{u}) \in \operatorname{dom} f$, let $\partial_x f(\bar{x}, \bar{u})$ denote the subgradients of $f(\,\cdot\,, \bar{u})$ at $\bar{x}$, and similarly $\widehat{\partial}_x f(\bar{x}, \bar{u})$ and $\partial_x^\infty f(\bar{x}, \bar{u})$. Likewise, let $d_x f(\bar{x}, \bar{u})$ and $\widehat{d}_x f(\bar{x}, \bar{u})$ denote the subderivative functions associated with $f(\,\cdot\,, \bar{u})$ at $\bar{x}$. One always has*

$$\begin{aligned} \widehat{\partial}_x f(\bar{x}, \bar{u}) &\supset \{v \,|\, \exists y \text{ with } (v, y) \in \widehat{\partial} f(\bar{x}, \bar{u})\}, \\ d_x f(\bar{x}, \bar{u})(w) &\geq df(\bar{x}, \bar{u})(w, 0) \text{ for all } w. \end{aligned}$$

Under the condition that $(0, y) \in \partial^\infty f(\bar{x}, \bar{u})$ implies $y = 0$ (this being true for convex f with $\operatorname{dom} f$ cannot be separated from $(\mathbb{R}^N, \bar{u})$), one also has

$$\begin{aligned} \partial_x f(\bar{x}, \bar{u}) &\subset \{v \,|\, \exists y \text{ with } (v, y) \in \partial f(\bar{x}, \bar{u})\}, \\ \partial_x^\infty f(\bar{x}, \bar{u}) &\subset \{v \,|\, \exists y \text{ with } (v, y) \in \partial^\infty f(\bar{x}, \bar{u})\}, \\ \widehat{d}_x f(\bar{x}, \bar{u})(w) &\leq \widehat{d} f(\bar{x}, \bar{u})(w, 0) \text{ for all } w. \end{aligned}$$

If also f is regular at $(\bar{x}, \bar{u})$, then $f(\,\cdot\,, \bar{u})$ is regular at $\bar{x}$ and

$$\begin{aligned}\partial_x f(\bar{x}, \bar{u}) &= \{v \,|\, \exists y \textit{ with } (v, y) \in \partial f(\bar{x}, \bar{u})\},\\ \partial_x^\infty f(\bar{x}, \bar{u}) &= \{v \,|\, \exists y \textit{ with } (v, y) \in \partial^\infty f(\bar{x}, \bar{u})\},\\ \widehat{d}_x f(\bar{x}, \bar{u})(w) &= \widehat{d} f(\bar{x}, \bar{u})(w, 0) \textit{ for all } w.\end{aligned}$$

Based on the above results, one can rewrite the Fermat's rule as below.

Example 3.6.4.1. [150, Example 10.12] For a proper, lsc function $f : \mathbb{R}^N \times \mathbb{R}^M \to \overline{\mathbb{R}}$ and vector $\bar{u} \in \mathbb{R}^M$, consider the problem of minimizing $f(x, \bar{u})$ over $x \in \mathbb{R}^N$. Suppose $\bar{x}$ is locally optimal and

$$\nexists\, y \neq 0 \quad \text{with } (0, \bar{y}) \in \partial^\infty f(\bar{x}, \bar{u}),$$

which in the case of convex f means that $\operatorname{dom} f$ does not have a supporting half-space at $(\bar{x}, \bar{u})$ with normal vector of the form $(0, y) \neq (0, 0)$. Then

$$\exists\, \bar{y} \quad \text{with } (0, \bar{y}) \in \partial f(\bar{x}, \bar{u}).$$

If f is regular at $(\bar{x}, \bar{u})$ and $f(\bar{x}, \bar{u})$ is convex in x, this condition is sufficient for $\bar{x}$ to be globally optimal.

3.7 KL framework

The convergence results of various algorithms in this thesis rely on the so-called Kurdyka–Łojasiewicz (KL) property. It has became a standard tool in recent years, and it is essentially satisfied by any function that appears in practice, we just state the definition here and refer to [7, 22, 23, 26, 97] for more details. The following definition is from [7].

Definition 3.7.0.1 (Kurdyka–Łojasiewicz property)**.** Let $f \colon \mathbb{R}^N \to \overline{\mathbb{R}}$ be an extended real valued function and let $\bar{x} \in \operatorname{dom} \partial f$. If there exists $\eta \in (0, \infty]$, a neighborhood U of $\bar{x}$ and a continuous concave function $\varphi \colon [0, \eta) \to \mathbb{R}_+$ such that

$$\varphi(0) = 0, \quad \varphi \in C^1(0, \eta), \quad \text{and} \quad \varphi'(s) > 0 \text{ for all } s \in (0, \eta),$$

and for all $x \in U \cap [f(\bar{x}) < f(x) < f(\bar{x}) + \eta]$ the Kurdyka–Łojasiewicz inequality

$$\varphi'(f(x) - f(\bar{x})) \|\partial f(x)\|_- \geq 1 \tag{3.7.1}$$

holds, then the function has the Kurdyka–Łojasiewicz property at $\bar{x}$. If, additionally, the function is lower semi-continuous and the property holds for each point in $\operatorname{dom} \partial f$, then f is called a Kurdyka–Łojasiewicz function.

We abbreviate Kurdyka–Łojasiewicz property as KL property. The function φ in the KL property is known as a desingularizing function. Many functions arising in practical problems satisfy the KL property, for example, semi-algebraic functions with a desingularizing function of the following form:

$$\varphi(s) = c s^{1-\theta}\,,$$

for certain $c > 0$ and $\theta \in [0, 1)$. The KL property is crucial in order to prove the global convergence of sequences generated by many algorithms, for example, PALM [26], iPALM [144], BPG [28], CoCaIn BPG (see Chapter 5) and many others. For the purpose of simplification of analysis, we use the following uniformization lemma for the KL property as detailed in [26].

Lemma 3.7.0.1 (Uniformized KL property [26, Lemma 3.6]). *Let Ω be a compact set and let $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ be proper and lower semicontinuous function. Assume that f is constant on Ω and satisfies KL property at each point on Ω. Then, there exist $\vartheta > 0$, $\eta > 0$, a continuous concave function $\varphi\colon [0, \eta) \to \mathbb{R}_+$ such that*

$$\varphi(0) = 0, \quad \varphi \in \mathcal{C}^1(0, \eta), \quad \textit{and} \quad \varphi'(s) > 0 \textit{ for all } s \in (0, \eta),$$

and for all $\bar{x} \in \Omega$ and x in the following intersection

$$\{x \in \mathbb{R}^N : \operatorname{dist}(x, \Omega) < \vartheta\} \cap [f(\bar{x}) < f(x) < f(\bar{x}) + \eta]$$

one has,

$$\varphi'(f(x) - f(\bar{x}))\|\partial f(x)\|_{-} \geq 1\,. \tag{3.7.2}$$

It is well known that the class of functions definable in an o-minimal structure satisfies KL property [23, Theorem 14]. The exact definition of o-minimal structure is given in [23, Definition 6], which we record below. We require the definition of canonical projection $\Pi : \mathbb{R}^{N+1} \to \mathbb{R}^N$ onto $\mathbb{R}^N$, which is defined by

$$\Pi(x_1, \ldots, x_N, t) = (x_1, \ldots, x_N)\,.$$

Definition 3.7.0.2 (o-minimal structure [23, Definition 6]). An o-minimal structure on $(\mathbb{R}, +, .)$ is a sequence of boolean algebras $\mathcal{O}_N$ of "definable" subsets of $\mathbb{R}^N$, such that for each $N \in \mathbb{N}$

(i) if A in $\mathcal{O}_N$, then $A \times R$ and $R \times A$ belong to $\mathcal{O}_{N+1}$;

(ii) if $\Pi : \mathbb{R}^{N+1} \to \mathbb{R}^N$ is the canonical projection onto $\mathbb{R}^N$ then for any A in $\mathcal{O}_{N+1}$, the set $\Pi(A)$ belongs to $\mathcal{O}_N$;

(iii) $\mathcal{O}_N$ contains a family of algebraic subsets of $\mathbb{R}^N$, that is, every set of the form

$$\{x \in \mathbb{R}^N : p(x) = 0\}\,,$$

where $p : \mathbb{R}^N \to \mathbb{R}$ is a polynomial function;

(iv) the elements of $\mathcal{O}_1$ are exactly the finite unions of intervals and points.

Definition 3.7.0.3 (definable function [23, Definition 7]). Given an o-minimal structure $\mathcal{O}$ (over $(\mathbb{R}, +, .)$), a function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is said to be definable in $\mathcal{O}$ if its graph belongs to $\mathcal{O}_{N+1}$.

Numerous functions and sets can be defined in an o-minimal structure, for example, sets and functions that are semi-algebraic and globally subanalytic. For a comprehensive discussion, we refer the reader to [21], [23, Section 4], [57] and [134, Section 4.5].

Semi-algebraic functions play a crucial role in this thesis. Hence, we briefly recall the definition of semi-algebraic sets and semi-algebriac functions as in [134, Definition 4.26].

Definition 3.7.0.4. A subset S of $\mathbb{R}^N$ is a real semi-algebraic set if it is expressible as

$$S = \bigcup_{j=1}^{p} \bigcap_{i=1}^{q} \{x \in \mathbb{R}^N \mid f_{i,j}(x) = 0, g_{i,j}(x) < 0\},$$

where $f_{i,j}, g_{i,j} : \mathbb{R}^N \to \mathbb{R},\ 1 \leq i \leq q,\ 1 \leq j \leq p,\ p, q \in \mathbb{N}$, are real polynomials. A function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is called a semi-algebraic function if its graph $\operatorname{Graph} f$ is a semi-algebraic subset of $\mathbb{R}^{N+1}$. A set-valued mapping $F : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ is semi-algebraic if its graph $\operatorname{Graph} F$ is a semi-algebraic subset of $\mathbb{R}^{N+M}$

There are numerous examples of semi-algebraic sets and functions. We list below few of them from [21, 26, 134].

- All real polynomial functions are semi-algebraic.
- Indicator functions of semi-algebraic sets are semi-algebraic.
- The sparsity measure of a vector $x \in \mathbb{R}^N$ defined by $\|x\|_0$, that is the number of non-zero terms in x, is semi-algebraic (see [26, Example 5.2]).
- The p-norm of a vector $x \in \mathbb{R}^N$ defined by

  $$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p \right)^{\frac{1}{p}}$$

  is semi-algebraic when p is a rational number and not semi-algebraic when p is an irrational number (see [26, Example 5.3]).
- We state the following results as in [54]. The semi-algebraic subsets of $\mathbb{R}$ are the unions of finitely many points and open intervals. If A is semi-algebraic subset of $\mathbb{R}^N$ and $L \subset \mathbb{R}^N$ a line, then $L \cap A$ is the union of finitely many points and open intervals.
- Consider the following sets:

  $$S_1 := \{(x, y) \in \mathbb{R}^2 \mid y = \exp(x)\},$$

  $$S_2 := \{(x, y) \in \mathbb{R}^2 \mid \exists n \in \mathbb{N}, y = nx\},$$

  $$S_3 := \{(x, y) \in \mathbb{R}^2 \mid y = \lfloor x \rfloor \text{ or } (x \in \mathbb{Z} \text{ and } x \leq y \leq x + 1)\}.$$

  The sets S_1, S_2, S_3 are not semi-algebraic.
- Then, the function given by $\frac{1}{2}\|A - UZ\|_F^2$, where $A \in \mathbb{R}^{M\times N}$, $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$, is semi-algebraic, as it is a real polynomial function. Even if A is a fixed matrix, the function still remains semi-algebraic. Such problems arise in the context of matrix factorization, which we consider later in this thesis.

Verifying that a given function satisfies the KL property could be difficult, however in their seminal work [22], Bolte, Daniilidis and Lewis prove that any proper, lower semicontinuous and semi-algebraic function satisfies the KL property on its domain. This important result makes this proof technique very powerful, since we are familiar with many semi-algebraic functions that appear very often in applications. In fact, the same result holds for (possibly non-smooth) functions that are definable in an o-minimal structure [22, 23]. For examples and more details about the relations between KL and other important notions, see [22, 24] and references therein. Instead of considering the further properties of semi-algebraic sets and functions, it is beneficial to consider the properties in the context of o-minimal structure as it is a more general notion. Hence, we now focus on the properties that arise in o-minimal structures. The following result shows that the functions definable in an o-minimal structure are closed under pointwise addition and multiplication. This is a standard result which can, for example, be found in [134, Corollary 4.32].

Lemma 3.7.0.2. *Let $S, T \subset \mathbb{R}^M$, $S \cap T = \emptyset$, and let $f : S \to \mathbb{R}^N$, $g : T \to \mathbb{R}^N$ be maps that belong to $\mathcal{O}$. Then, pointwise addition and multiplication, $f + g$ and $f \cdot g$, restricted to $S \cap T$ belongs to $\mathcal{O}$.*

The following result connects KL property to functions that are definable in an o-minimal structure.

Theorem 3.7.0.3 ([23, Theorem 14])**.** *Any proper lower semi-continuous function $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ that is definable in an o-minimal structure $\mathcal{O}$ has the Kurdyka–Łojasiewicz property at each point of $\operatorname{dom} \partial f$. Moreover the function φ in Lemma 3.7.0.1 is definable in $\mathcal{O}$.*

3.7.1 Discussion

KL property plays a crucial role in the convergence analysis of many optimization algorithms, such as Proximal Gradient Method [7], Bregman Proximal Gradient (BPG) [28], Proximal Alternating Linearized Minimization (PALM) based algorithms [26, 144] and many others. Usually, the essential conditions required for the global convergence analysis of certain optimization algorithms can be collected in an abstract manner, and are clearly summarized and studied in [7, 26]. Basically, the conditions that need to be verified are called "sufficient descent condition", "relative error condition", and "continuity condition". The sequence satisfying such conditions is at-times called gradient-like descent sequence [28], which we detail in Section 9.4. In the context of KL functions, to prove the global convergence of the full sequence of iterates generated by certain optimization algorithm, it mostly suffices to prove that it is a gradient-like descent sequence.

In this thesis, we use the same technique in the analysis of the proposed algorithms, CoCaIn BPG in Chapter 5, Model BPG in Chapter 9, Model CoCaIn BPG in Chapter 10.

Chapter 4

Bregman distances

4.1 Abstract

The L-smad property, a generalization of the Lipschitz continuous gradient property, relies on the so-called Bregman distances, which are generalized proximity measures. In this chapter, we review the Bregman distance concept and recall some known properties. We also recall the L-smad property. The focus here is on proposing suitable Bregman distances to some popular objectives that arise in real world applications. We mainly focus on objectives that arise in matrix factorization, deep matrix factorization and deep non-linear neural networks, for which we verify the L-smad property. For this purpose, we propose novel Bregman distances suitable for the considered setting.

4.2 Introduction

The Euclidean distance is a standard tool that is very prevalent in the optimization field. It is used in concepts such as Lipschitz continuous gradient property, strong convexity, proximal algorithms and many others. In the introduction of this thesis, we mentioned that the Lipschitz continuous gradient property is restrictive, as it is based on quadratic bounds. Such quadratic bounds arise due to the usage of the Euclidean distance. This is because the Euclidean distance cannot capture all the geometries that arise in real world applications. Hence, generalized proximity measures are sought after. In this thesis, we are mainly interested in Bregman distances that generalize the Euclidean distance. Bregman distances are generated from so-called Legendre function (Definition 4.3.0.1), say h. Essentially, the Bregman distance is the difference between h and the linear approximation of h. For illustrative purposes, let h be convex and continuously differentiable over $\mathbb{R}^N$. Then, the Bregman distance D_h between two points $x, y \in \mathbb{R}^N$ generated by h is given by

$$D_h(x,y) = h(x) - (h(y) + \langle \nabla h(y), x - y \rangle) ,$$

where $h(y) + \langle \nabla h(y), x - y \rangle$ is the linear approximation of h at y. We illustrate the Bregman distance using a simple function $h(x) = x^4$ in Figure 4.1.

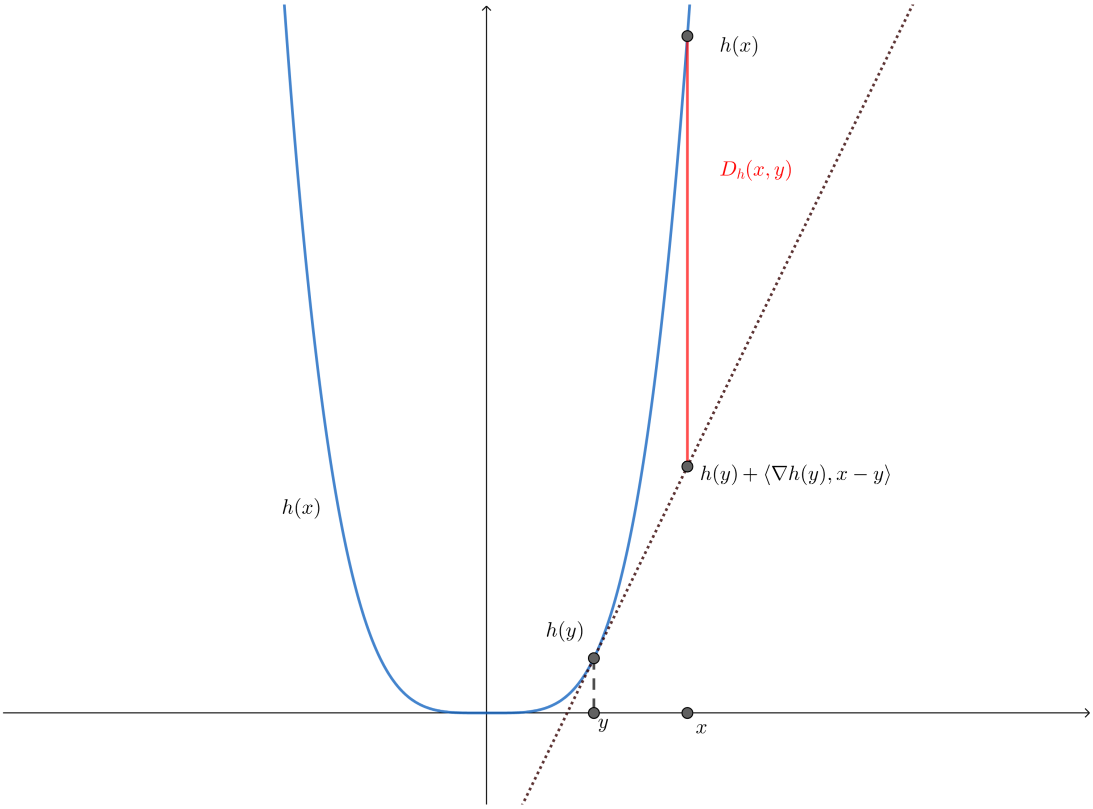


FIGURE 4.1: Illustration of the Bregman distance between two point $x, y \in \mathbb{R}$. Set $h(x) = x^4$. The Bregman distance is the difference between $h(x)$ and the linearization of h at y evaluated at x, which we denote via the red line segment.

Bregman distances were first considered in [34] in the context of projection onto the intersection of closed convex sets, which arises in the fields of image reconstruction, minimization of convex functions, statistical estimation and many others. Recently, Bregman distances became popular in various applications related to both convex and non-convex optimization, which we will discuss shortly. We are mainly interested in one thread of research that involves proposing extensions to the Lipschitz continuous gradient property and developing related algorithms. Notably, such algorithms are applicable to various problems arising in machine learning, computer vision, statistics and many other fields (see Chapter 5, 6, 7, 8, 9, 10).

In this regard, for non-convex and non-smooth optimization, the extension of the Lipschitz continuous gradient property known as the L-smad property was proposed in [28] and it will be key to this thesis. Based on the L-smad property, the Bregman Proximal Gradient (BPG) algorithm was proposed in [28], which we recall in Chapter 5. BPG forms the foundation for the developing various algorithms, which will be the focus of subsequent chapters. In this chapter, our goals are

- to introduce the concept of Bregman distance and its properties,
- to introduce the L-smad property and discuss its significance,
- to discuss the algorithmic implications of the L-smad property,
- to explore various practical applications and develop suitable Bregman distances such that the L-smad property is satisfied.

4.2.1 Contributions

Our main contributions in this chapter involves proposing appropriate Bregman distances suitable for the objectives that arise in the context of matrix factorization, deep matrix factorization and deep neural network settings such that the L-smad property is satisfied. In particular, we list our contributions below.

- We propose a novel Bregman distance for matrix factorization problem (1.2.7) with the following auxiliary function (called kernel generating distance) with certain $c_1, c_2 > 0$:

  $$h(U,Z) = c_1\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)^2 + c_2\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)\,,$$

  and verify the L-smad property. The generated Bregman distance embeds the crucial coupling between the variables U and Z.
- For the matrix factorization problems, we connect with few parallel works and show that our Bregman distance results in the tightest value for L in the L-smad property.
- Based on similar ideas used in the matrix factorization setting, we propose the Bregman distances suitable for deep matrix factorization problems and verify the L-smad property.
- Finally, we also consider both the regression and the classification problems that arise in the context of deep non-linear neural networks and propose suitable Bregman distances to satisfy the L-smad property.
- For the deep matrix factorization and the generic deep non-linear neural network settings, we rely on the Legendre functions that takes the following form:

  $$h(x) = \sum_{i=1}^{N} a_i \|x\|_2^{2i}\,,$$

  where $a_i \geq 0$ for all $i \in \{1, \ldots, N\}$. The choice of a_i's vary according to the setting in consideration.

4.2.2 Related work

As mentioned earlier, Bregman distances were first considered in [34] in the context of projection onto the intersection of closed convex sets. Earlier methods relied on Euclidean distances to generate an orthogonal projection, whereas with Bregman distances, non-orthogonal projections are possible. Various related settings were considered in [44, 45] that popularized Bregman distances further. Bregman distances also became popular in the context of proximal point algorithms [46, 160]. In the related works on maximal monotone operator setting, the Bregman distances were popularized due to [65]. The seminal Mirror Descent algorithm [14] incorporates Bregman distances in the update step, and such an update step can be interpreted as a Gradient Descent like step in the dual space based on the reference function that generates the Bregman distance.

Recently, there has been huge surge of work on Bregman distances [16, 37, 53, 66, 71, 91, 141, 161]. This is due to the flexibility one gains in modelling the proximity measures and the ability to design algorithms that are suitable for objectives that arise in machine learning, computer vision and many others contemporary research areas. In this regard, the so-called Bregman Proximal Gradient [28] and related algorithms (see Chapter 5, 9, 10) are increasingly becoming popular. A major drawback of using Bregman Proximal Gradient algorithms is that the update step is non-trivial to solve. However, at times the special structure of the Bregman distance can result in closed form update steps, simple case being Gradient Descent with the Euclidean distance. Also, for instance in the minimization problem obtained for deblurring an image under Poisson noise, one can obtain a closed form expression for an optimization subproblem using a Bregman distance generated by Burg's entropy [139]. Various closed form update steps were also proposed in Chapter 6, 7, 8.

The crucial observation that Bregman distances can indeed be used to generalize the notion of Lipschitz continuous gradient was considered in [10]. However, their setting was restricted to convex problems. This was later mitigated in [28], via the L-smad property for non-convex problems. During the same time, closely related notions such as relative smoothness [109], and relative continuity [108] were also proposed based on Bregman distances. Before [10] and [109], the work in [20] also considered a generalization of the Lipschitz continuous gradient notion. More related references also include [98, 130]. We later see in this thesis that the L-smad property defined above can also be restrictive, and thus we propose the MAP property in Chapter 9 (based on a closely related work is [55]) to generalize the L-smad property even further. The Bregman Proximal Gradient algorithms detailed later in Chapter 5 rely on the L-smad property. Model BPG variants in Chapter 9 and Chapter 10 rely on the MAP property.

In order for the BPG or Model BPG based algorithms to be applicable, it is required to verify the L-smad or the MAP property. In this regard, we mainly tackle the objectives arising in matrix factorization, deep matrix factorization and deep neural networks. Bregman distances for structured matrix factorization problems were considered in [58, 84, 103, 162] along with our work in Section 4.5. Extensions to deep linear neural networks were considered in Section 4.6. We also propose suitable Bregman distances to various practical deep neural network settings in Section 4.7, 4.8. Bregman distances allow for many optimization algorithms, which were previously thought to be completely different to co-exist in a single algorithm, thus making the analysis simpler.

4.3 Bregman distances

In Chapter 1, it was mentioned that the Lipschitz continuous gradient property is restrictive and more general notion such as L-smad property is sought after. However, the L-smad property relies on generalized proximity measures known as Bregman distances, which generalize the standard Euclidean distance. Bregman distances are generated from so-called Legendre functions, which is defined below.

Definition 4.3.0.1 (Legendre function [149, Section 26])**.** Let $h : \mathbb{R}^N \to \overline{\mathbb{R}}$ be a proper lsc convex function. It is called:

(i) essentially smooth, if h is differentiable on $\operatorname{int}\operatorname{dom} h$, with moreover $\|\nabla h(x_k)\| \to \infty$ for every sequence $(x_k)_{k\in\mathbb{N}} \in \operatorname{int}\operatorname{dom} h$ converging to a boundary point of $\operatorname{dom} h$ as $k \to \infty$;

(ii) of Legendre type if h is essentially smooth and strictly convex on $\operatorname{int}\operatorname{dom} h$.

Some properties of Legendre function include the following:

$$\operatorname{dom}\partial h = \operatorname{int}\operatorname{dom} h, \text{ and } \partial h(x) = \{\nabla h(x)\}, \forall x \in \operatorname{int}\operatorname{dom} h.$$

Additional properties can be found in [11, Section 2.3]. Legendre function has variants such as kernel generating distance [28], or a reference function [109]. Generic reference functions used in [109] are more general compared to Legendre functions, as they do not require essential smoothness. In this thesis, we only consider Legendre functions or kernel generating distances, which are almost equivalent and the discussion is provided below. We provide below the definition of kernel generating distance, which was recently stated in [28] (in this respect see also [8]).

Definition 4.3.0.2. (Kernel Generating Distance) Let C be a nonempty, convex and open subset of $\mathbb{R}^N$. Associated with C, a function $h : \mathbb{R}^N \to (-\infty, +\infty]$ is called a kernel generating distance if it satisfies the following:

(i) h is proper, lower semicontinuous and convex, with $\operatorname{dom} h \subset \overline{C}$ and $\operatorname{dom}\partial h = C$.

(ii) h is C^1 on $\operatorname{int}\operatorname{dom} h \equiv C$.

We denote the class of kernel generating distances by $\mathcal{G}(C)$.

A strictly convex kernel generating distance function is equivalent to the Legendre function, due to the following standard result (for example, see [148, Theorem 26.1]).

Definition 4.3.0.3. Let $h : \mathbb{R}^N \to \overline{\mathbb{R}}$ be a proper, lsc and convex function. The following statements are equivalent:

(i) h is essentially smooth.

(ii) $\partial h(x) = \{\nabla h(x)\}$ when $x \in \operatorname{int}\operatorname{dom} h$, while $\partial h(x) = \emptyset$ for $x \notin \operatorname{int}\operatorname{dom} h$.

(iii) $\operatorname{dom}\partial h = \operatorname{int}\operatorname{dom} h \neq \emptyset$.

The significance of kernel generating distance is as follows. In several optimization problems that arise in practical applications, it is the case that the objective function is optimized over a nonempty, convex and open set $C \subset \mathbb{R}^N$. At times, it is conducive for optimization if the set C is assigned to a kernel generating function h, such that the sub-problems that arise in algorithms (for example, see Chapter 5) essentially become unconstrained.

The Bregman distance associated with any Legendre function h or a kernel generating distance is defined by

$$D_h(x,y) = h(x) - h(y) - \langle x - y, \nabla h(y)\rangle\,, \quad \forall\, x \in \operatorname{dom} h,\; y \in \operatorname{int}\operatorname{dom} h\,. \tag{4.3.1}$$

This object is not a distance according to the classical definition (for example, it is not symmetric in general). However, the Bregman distance between two points is nonnegative if and only if the function h is convex. If h is known to be strictly convex, we have that $D_h(x,y) = 0$ if and only if $x = y$. The classic example of a Bregman distance is the squared Euclidean distance, which is generated by $h(x) = (1/2)\,\|x\|^2$. For more examples, results and applications of Bregman distances, see [11, 46, 65, 160, 161] and references therein.

4.3.1 Properties

For this section, we use the results stated in [139, Section 3]. The class of proper, closed, convex Legendre functions is denoted by $\mathscr{L}$.

Proposition 4.3.1.1. *Let $h \in \mathscr{L}$ and D_h be the associate Bregman distance.*

(i) *D_h is strictly convex on every convex subset of* $\operatorname{dom}\partial h$ *with respect the first argument.*

(ii) *For $y \in \operatorname{int}\operatorname{dom} h$, it holds that $D_h(x,y) = 0$ if and only if $x = y$.*

(iii) *For $x \in \mathbb{R}^N$ and $u, v \in \operatorname{int}\operatorname{dom} h$ the following three point identity holds:*

$$D_h(x,u) = D_h(x,v) + D_h(v,u) + \langle x - v, \nabla h(v) - \nabla h(u)\rangle\;. \tag{4.3.2}$$

Proof. (i) and (ii) follow directly from the definition of h being essentially strictly convex. (iii) is stated in [12, Prop. 2.3]. It follows from the definition of a Bregman distance. □

4.3.2 Examples

Prominent examples of Bregman distances can be found in [10, Example 1, 2]. We provide some examples below. For any vector $x \in \mathbb{R}^N$, the i^{th} coordinate is denoted by x_i.

- Bregman distance generated from $h(x) = \frac{1}{2}\|x\|^2$ is equivalent to the Euclidean distance. Here, the conjugate is given by $h^*(y) = \frac{1}{2}\|y\|^2$.

- Let $x, \bar{x} \in \mathbb{R}^N_{++}$, for $h(x) = -\sum_{i=1}^N \log(x_i)$ (Burg's entropy), the generated Bregman distance is

  $$D_h(x,\bar{x}) = \sum_{i=1}^N \left(\frac{x_i}{\bar{x}_i} - \log\left(\frac{x_i}{\bar{x}_i}\right) - 1\right)\,.$$

  Such distances are helpful in Poisson linear inverse problems [10, 139] (Chapter 9). Here, the conjugate is given by $h^*(y) = -\sum_{i=1}^N \log(-y_i) - 1$ with $\operatorname{dom} h^* = \mathbb{R}^N_{--}$, where $\mathbb{R}^N_{--} := (-\infty, 0) \times \ldots \times (-\infty, 0)$.

- Let $x \in \mathbb{R}^N_+$, $\bar{x} \in \mathbb{R}^N_{++}$, for $h(x) = \sum_{i=1}^N x_i \log(x_i)$ (Boltzmann–Shannon entropy), with $0\log(0) := 0$, the Bregman distance is given by

  $$D_h(x,\bar{x}) = \sum_{i=1}^N x_i(\log(x_i) - \log(\bar{x}_i)) - (x_i - \bar{x}_i)\,.$$

Such distances are helpful to handle simplex constraints [14]. Here, the conjugate is given by $h^*(y) = \sum_{i=1}^N \exp(y_i) - N$ with $\operatorname{dom} h^* = \mathbb{R}^N$.

- Phase retrieval problems [28] (standard phase retrieval problem mentioned in Section 1.2.1) use the Bregman distance based on the Legendre function $h : \mathbb{R}^N \to \mathbb{R}$ that is given by

$$h(x) = \frac{1}{4}\|x\|_2^4 + \frac{1}{2}\|x\|_2^2 .$$

  Here, the conjugate is given by $h^*(y) = \frac{3}{4}\|y\|_2^{\frac{4}{3}} + \frac{1}{2}\|y\|_2^2$ with $\operatorname{dom} h^* = \mathbb{R}^N$.

- In Section 4.5, we show that matrix factorization problems use the Bregman distance based on the Legendre function $h : \mathbb{R}^{N_1} \times \mathbb{R}^{N_2} \to \mathbb{R}$ that is given by

$$h(x_1, x_2) = c_1 \left(\frac{\|x_1\|^2 + \|x_2\|^2}{2} \right)^2 + c_2 \left(\frac{\|x_1\|^2 + \|x_2\|^2}{2} \right) ,$$

  with certain $c_1, c_2 > 0$ and $N_1, N_2 \in \mathbb{N}$. Here, the conjugate function $h^* : \mathbb{R}^{N_1} \times \mathbb{R}^{N_2} \to \mathbb{R}$ is given by

$$h^*(y_1, y_2) = \frac{3c_1}{4(c_1^{\frac{4}{3}})} \left(\|y_1\|^2 + \|y_2\|^2 \right)^{\frac{4}{3}} + \frac{1}{2c_2} \left(\|y_1\|^2 + \|y_2\|^2 \right) ,$$

  with full domain.

- In deep neural networks (see Section 4.6, 4.7, 4.8), the following Legendre functions are prominent

$$h(x) = \sum_{i=1}^N a_i \|x\|_2^{2i} ,$$

  where $a_i \geq 0$ for all $i \in \{1, \ldots, N\}$. Here, the value of N varies according to the setting under consideration. Here, the conjugate is given

$$h^*(y) = \sum_{i=1}^N a_i(2i-1) \left(\frac{1}{c(2i)} \right)^{\frac{2i}{2i-1}} \|y\|_2^{\frac{2i}{2i-1}} ,$$

  with full domain.

- Fermi-Dirac entropy is given by $h(x) = x\log(x) + (1-x)\log(1-x)$ with $\operatorname{dom} h = [0,1]$. The conjugate of the Fermi-Dirac entropy is given by $h^*(y) = \log(1 + \exp(y))$ with $\operatorname{dom} h^* = \mathbb{R}$.

- Hellinger function is given by $h(x) = -\sqrt{1-x^2}$ with $\operatorname{dom} h = [-1, 1]$. The conjugate is given by $h^*(y) = \sqrt{1+y^2}$ with $\operatorname{dom} h^* = \mathbb{R}$.

4.4 The Bregman framework

In this section we detail the recent concept of smooth adaptable functions (functions satisfying the L-smad property), which in some sense extends and generalizes the class of smooth functions with globally Lipschitz continuous gradient.

We focus on additive composite problems given by:

$$(\mathcal{P}) \qquad \inf\left\{f \equiv f_0\left(x\right) + f_1\left(x\right) : \ x \in \overline{C}\right\},$$

where f_0, f_1 satisfy Assumption A given below. One important feature of using Bregman distances in optimization algorithms is the ability of relate the constraint set C to a certain kernel generating distances function $h \in \mathcal{G}(C)$. For the rest of the chapter, we make the following assumption.

Assumption A. (i) $h \in \mathcal{G}(C)$ with $\overline{C} = \overline{\operatorname{dom} h}$.

(ii) $f_0 : \mathbb{R}^N \to (-\infty, +\infty]$ is a proper and lower semicontinuous function (possibly non-convex) with $\operatorname{dom} f \cap C \neq \emptyset$.

(iii) $f_1 : \mathbb{R}^N \to (-\infty, +\infty]$ is a proper and lower semicontinuous function (possibly non-convex) with $\operatorname{dom} h \subset \operatorname{dom} f_1$, which is continuously differentiable on C.

(iv) $v(\mathcal{P}) := \inf\left\{f\left(x\right) : \ x \in \overline{C}\right\} > -\infty$.

4.4.1 Smooth adaptable functions

Here, we deal with the non-convex optimization model $(\mathcal{P})$ where the gradient of the smooth function f_1 is not globally Lipschitz. Recently, Bauschke, Bolte and Teboulle [10], observed that the property of having a Lipschitz continuous gradient can be interpreted equivalently as a certain convexity condition on the function itself (see description above (1.1.5) in Chapter 1). This opens the gate for generalizing known results in the convex setting. It was extended to the non-convex setting in [28] with the concept of smooth adaptable functions given below.

Definition 4.4.1.1 (L-smooth adaptable)**.** A pair (f_1, h) is called L-smooth adaptable (L-smad) on C if there exists $L > 0$ such that $Lh - f_1$ and $Lh + f_1$ are convex on C.

Note that the L-smad property considered in Chapter 1 is a special case of the above definition with $C = \mathbb{R}^N$. The optimization model $(\mathcal{P})$ appears with a smooth term in the objective function which is very common in many fields of applications. For L-smooth adaptable functions, we will use the following extended version of the Descent Lemma (see [28, Lemma 2.1, p. 2134]).

Lemma 4.4.1.1 (Extended Descent Lemma)**.** *The pair of functions (f_1, h) is L-smooth adaptable on C if and only if:*

$$\left|f_1\left(x\right) - f_1\left(y\right) - \left\langle \nabla f_1\left(y\right), x - y\right\rangle\right| \leq LD_h\left(x, y\right), \quad \forall\ x, y \in \operatorname{int}\operatorname{dom} h. \tag{4.4.1}$$

Remark 4.4.1.1 (Invariance to strong convexity). We would like to note that the L-smooth adaptable property is invariant when h is additionally assumed to be σ-strongly convex. Indeed, as described in [28], since convexity of f_1 is not needed, we can define $\omega(x) := (\sigma_1/2)\left\|x\right\|^2$, and then for any $0 < \sigma_1 < \sigma$, we have

$$Lh - f_1 = L\left(h - \omega\right) - \left(f_1 - L\omega\right) := L\bar{h} - \bar{f}_1,$$

namely, the new pair $\left(\bar{f}_1, \bar{h}\right)$ satisfies the L-smad property on C.

Based on the L-smad property, provably globally convergent algorithms can be developed, which is the main focus on Chapter 5. We already illustrated the L-smad property in Figure 1.2. Now, we focus on providing few practical examples of the L-smad property. To this regard, we focus on objectives that arise in the context of matrix factorization, deep matrix factorization and deep non-linear neural networks. We design the Bregman distances for each of the mentioned setting and verify the L-smad property.

4.5 Bregman distance for matrix factorization

Matrix factorization has numerous applications in machine learning [112, 156], computer vision [48, 82, 157, 170], bio-informatics [35, 155] and many others. Given a matrix $A \in \mathbb{R}^{M\times N}$, one is interested in the factors $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$ such that $A \approx UZ$ holds. This is usually cast into the following non-convex optimization problem

$$\min_{U\in\mathcal{U},Z\in\mathcal{Z}} \left\{ f \equiv \frac{1}{2}\left\|A - UZ\right\|_F^2 + \mathcal{R}_1(U) + \mathcal{R}_2(Z) \right\} , \tag{4.5.1}$$

where $\mathcal{R}_1, \mathcal{R}_2$ are regularization terms, $\frac{1}{2}\left\|A - UZ\right\|_F^2$ is the data-fitting term, and $\mathcal{U}, \mathcal{Z}$ are the constraint sets for U and Z respectively. Here, $\mathcal{R}_1(U)$ and $\mathcal{R}_2(Z)$ can be potentially non-convex extended real valued functions and possibly non-smooth.

Denote $f_1(U,Z) := \frac{1}{2}\left\|A - UZ\right\|_F^2$ and $f_0(U,Z) := \mathcal{R}_1(U) + \mathcal{R}_2(Z)$. We prove the L-smad property for f_1. The kernel generating distance is a linear combination of

$$h_1(U,Z) := \left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2} \right)^2 \quad \text{and} \quad h_2(U,Z) := \frac{\|U\|_F^2 + \|Z\|_F^2}{2} , \tag{4.5.2}$$

Proposition 4.5.0.1. *Let f_1, h_1, h_2 be as defined above. Then, for $L \geq 1$, the function f_1 satisfies the L-smad property with respect to the following kernel generating distance*

$$h_a(U,Z) = 3h_1(U,Z) + \|A\|_F\, h_2(U,Z) . \tag{4.5.3}$$

The proof is given in Section A.2 in the appendix. The Bregman distances considered in [28] are separable and are not applicable for matrix factorization problems. The inherent coupling between two subsets of variables U, Z is the main source of non-convexity in the objective f_1. The kernel generating distance (in particular h_1) contains the interaction/coupling terms between U and Z which makes it amenable for matrix factorization problems.

4.5.1 Connection to related work in 2D setting

We briefly consider a two dimensional matrix factorization problem to compare various related strategies [28, 103] to compute the appropriate Bregman distance. In this regard, we use three strategies to develop suitable Bregman distances, namely, method 1 from [28], method 2 from [103] and method 3 from our setting.

Method 1. Consider the setting of standard phase retrieval from Section 1.2.1 (also see Chapter 5). Denote $A_i := a_i a_i^T$,

$$f_1\left(x\right) = \frac{1}{4}\sum_{i=1}^{m} \left(\langle x, A_i x\rangle - b_i^2 \right)^2 , \quad \text{and } h\left(x\right) = \frac{1}{4}\|x\|_2^4 + \frac{1}{2}\|x\|_2^2 .$$

As per [28, Lemma 5.1], the function $Lh - f_1$ is convex, where

$$L \geq \sum_{i=1}^{m}\left(3\left\|A_i\right\|^2 + \left\|A_i\right\| \left|b_i^2\right| \right) .$$

In the setting of two dimensions using $x = (x_1, x_2)$, $m = 1$ and $A_1 = \frac{1}{2}\begin{pmatrix} 0, & 1 \\ 1, & 0 \end{pmatrix}$, we obtain the matrix factorization problem in 2D. Then, with

$$L \geq \left(\frac{3}{2} + \frac{1}{\sqrt{2}}\left|b_1^2\right|\right), \quad h(x_1, x_2) = \frac{1}{4}(x_1^2 + x_2^2)^2 + \frac{1}{2}(x_1^2 + x_2^2),$$

the function $Lh - f_1$ is convex. In other words, the following function is convex, with certain $\theta \geq 1$,

$$\begin{aligned} &\theta\left(\frac{3}{2} + \frac{1}{\sqrt{2}}\left|b_1^2\right|\right)\left(\frac{1}{4}\left\|x\right\|_2^4 + \frac{1}{2}\left\|x\right\|_2^2\right) - f_1, \\ &= \theta\left(\frac{3}{2} + \frac{1}{\sqrt{2}}\left|b_1^2\right|\right)\left(\frac{1}{c_1}\frac{c_1}{4}\left\|x\right\|_2^4 + \frac{1}{c_2}\frac{c_2}{2}\left\|x\right\|_2^2\right) - f_1. \end{aligned}$$

Then, with $c_1 = \frac{3}{2}$, $c_2 = \frac{b_1^2}{2}$, we can deduce that $L_1 h_1 - f_1$ is convex, where

$$L_1 \geq \left(\frac{3}{2} + \frac{1}{\sqrt{2}}\left|b_1^2\right|\right)\frac{1}{\min\{c_1, c_2\}}, \quad h_1(x) := \frac{c_1}{4}\left\|x\right\|_2^4 + \frac{c_2}{2}\left\|x\right\|_2^2.$$

Rewriting L_1 lower bound, we obtain that

$$L_1 \geq (\sqrt{2}b_1^2 + 3)\max\left\{\frac{1}{3}, \frac{1}{b_1^2}\right\}.$$

Method 2. From the method in [103], after a brief calculation, we obtain that with the following Legendre function

$$h_2(x_1, x_2) = \frac{3}{2}\frac{\left({x_1}^2 + {x_2}^2\right)^2}{4} + \frac{b_1^2}{2}\left(\frac{{x_1}^2 + {x_2}^2}{2}\right) + 1,$$

and with any

$$L_2 \geq (\sqrt{2}b_1^2 + 3)\max\left\{\frac{1}{3}, \frac{1}{b_1^2}\right\},$$

the function $L_2 h_2 - f_1$ is convex. Note that the constant in the Legendre function is not affecting the convexity of $L_2 h_2 - f_1$.

Method 3. With the matrix factorization setting which we proposed in this chapter, we deduce that with

$$h_3(x_1, x_2) = c_1\left(\frac{x_1^2 + x_2^2}{2}\right)^2 + c_2\left(\frac{x_1^2 + x_2^2}{2}\right),$$

$c_1 = \frac{3}{2}$ and $c_2 = \frac{b_1^2}{2}$, the function $L_3 h_3 - f_1$ is convex for $L_3 \geq 1$.

As illustrated above, it is clear that L_3 gives the tightest value on the choice of scaling factor required for the Legendre function. This implies that our analysis in the context of matrix factorization is the tightest. In regard to method 1, the immediate relation to the matrix factorization problem was not clear. Additionally, the matrix A_i is required to be symmetric, thus generalization of above mentioned strategy in method 1 may fail in higher dimensions.

We continue the matrix factorization setting in Chapter 6, where we illustrate the efficient applicability of BPG based algorithms based on the proposed Bregman distances.

4.6 Bregman distances for deep matrix factorization

Matrix factorization problems consider only two factors and it is natural to consider extensions that involve arbitrary number of factors. Deep matrix factorization deals with this setting, which is our main focus in this section. We note that the Bregman distance proposed for matrix factorization is not valid for the deep matrix factorization setting involving an arbitrary number of factors. The main contribution of this section is to derive Bregman distances suitable for performing deep matrix factorization with a quadratic loss. Such a problem is equivalent to training a so-called deep linear neural network, which is an important and interesting optimization problem. As remarked by [77] and in view of [49, 92, 169, 175], it is well justified to study the theoretically more tractable deep linear neural networks instead of the more challenging deep nonlinear networks (Section 4.7, 4.8). Deep matrix factorization model also has applications in matrix completion (c.f. Section 7.6).

Based on the additive composite problem setting in Section 4.4, the deep matrix factorization problem or equivalently training a so-called deep linear neural network (DLNN) model involves solving the following optimization problem:

$$\min_{W_i \in \mathcal{W}_i, \forall i \in \{1,\ldots,N\}} f_1(W) + f_0(W)\,, \tag{4.6.1}$$

where

$$f_1(W) := \frac{1}{2}\left\|W_1W_2\cdots W_NX - Y\right\|_F^2\,,$$

f_0 is possibly a regularization term, and N denotes the number of layers. Here, we set f_0 to be a zero function, as our main focus here is to provide suitable Bregman distances such that f_1 is L-smad with respect to h. Furthermore, we denote by $\mathcal{W}_i = \mathbb{R}^{d_i \times d_{i+1}}$ where $d_i \in \mathbb{N}$ for all $i \in \{1,\ldots,N\}$. Let $d_{N+1} = d$ and $X \in \mathbb{R}^{d \times n_T}$ be fixed, where $n_T \in \mathbb{N}$, which typically corresponds to the number of training samples. Similarly we have fixed $Y \in \mathbb{R}^{d_1 \times n_T}$, which typically corresponds to the labels of the inputs in X. We denote by $W := (W_1,\ldots,W_N)$, meaning W lies in the product space $\mathcal{W} := \mathcal{W}_1 \times \cdots \times \mathcal{W}_N$, equipped with the norm $\|W\|_F^2 := \sum_{i=1}^N \|W_i\|_F^2$. We focus on $N \geq 2$ in this section.

To prove the L-smad property we consider its characterization via the Hessian. More precisely, if h and f_1 are twice continuously differentiable, $Lh - f_1$ and $f_1 + Lh$ are convex if and only if $L\nabla^2 h(x) \succeq \nabla^2 f_1(x)$ and $-L\nabla^2 h(x) \preceq \nabla^2 f_1(x)$, i.e., the eigenvalues of the Hessian of f_1 are bounded by the eigenvalues of the Hessian of Lh. Our analysis suggests that the odd and the even case have to be considered separately.

Even number of layers. Let N be even and define the following functions:

$$H_1(W) := \left(\frac{\|W\|_F^2}{N}\right)^N, \quad H_2(W) := \left(\frac{\|W\|_F^2}{N}\right)^{\frac{N}{2}}.$$

Then, we have the following result, which shows that for an appropriate linear combination of H_1 and H_2 we obtain the L-smad property for f_1 in (4.6.1).

Proposition 4.6.0.1. *Let H_1, H_2 be as defined above and let f_1 be as in (4.6.1). Then, for $L = 1$, the function f_1 satisfies the L-smad property with respect to the following kernel generating distance*

$$H_a(W) = c_1(N)H_1(W) + c_2(N)H_2(W)\,, \tag{4.6.2}$$

where we have

$$c_1(N) = \frac{(2N-1)N^N}{2N!}\|X\|_F^2\,, \quad c_2(N) = \frac{\|Y\|_F\,\|X\|_F\,(N-1)N^{\frac{N-2}{2}}}{(N-2)^{\frac{N-2}{2}}}\,.$$

The proof is given in Section A.3.1 in the appendix.

Note that H_a is a polynomial of order $2N$ as a linear combination of a degree $2N$ and a degree N polynomial. The resulting Bregman distances are data-dependent. More precisely, the coefficients $c_1(N)$ and $c_2(N)$ are dependent on the number of layers, X and Y. We remark that for $N = 2$ and $\|X\|_F = 1$, this matches the results from Section 4.5 for the matrix factorization problems.

Odd number of layers. Let N be odd and denote

$$H_3(W) := \left(\frac{\|W\|_F^2 + 1}{N+1}\right)^{\frac{N+1}{2}}\,. \tag{4.6.3}$$

As the following proposition reveals, the loss function for the odd case is L-smooth adaptable with respect to a degree $2N$ polynomial H_b which is given as a linear combination of H_1 and H_3.

Proposition 4.6.0.2. *Let H_1, H_3 be as defined above and let f_1 be as in (4.6.1). Then, for $L = 1$, the function f_1 satisfies the L-smad property with respect to the following kernel generating distance*

$$H_b(W) = c_1(N)H_1(W) + c_3(N)H_3(W)\,, \tag{4.6.4}$$

where we have

$$c_1(N) = \frac{(2N-1)N^N}{2N!}\|X\|_F^2\,, \quad c_3(N) = \frac{\|Y\|_F\,\|X\|_F\,(N-1)(N+1)^{\frac{N-1}{2}}}{(N-1)^{\frac{N-1}{2}}}\,.$$

The proof is given in Section A.3.3 in the appendix.

Like in the even case H_1 is a polynomial of order $2N$. However, here H_2 is not applicable as N is odd. We fix this issue using H_3, a polynomial of order $N+1$. Note that the analysis of the objective results in a polynomial of degree only N. This is automatically resolved with H_3, because the constant term 1 in H_3 allows for certain terms to be of order N, while preserving the convexity of H_3. Note that this is just one potential way to obtain polynomials of order N. We show that the proposed Bregman distances are efficient to implement in practice.

Strong convexity of h. The global convergence result of BPG in [28] (also Theorem 7.3.2.1) relies on the strong convexity of h. We denote σ as the strong convexity parameter. For $N = 2$ the strong convexity is satisfied directly by H_a. Denote the following:

$$H_4(W) := \frac{\|W\|_F^2}{N}\,.$$

For even $N > 2$, with $\rho > 0$, we use the following h:

$$h(W) = H_a(W) + \rho H_4(W)\,, \tag{4.6.5}$$

for which $\sigma = \frac{2\rho}{N}$. For odd $N > 2$, with $\rho > 0$, we use the following h:

$$h(W) = H_b(W) + \rho H_4(W)\,, \tag{4.6.6}$$

thus $\sigma = \frac{1}{(N+1)^{\frac{N-1}{2}}} + \frac{2\rho}{N}$. We fix ρ in the initialization phase of the algorithms.

In Chapter 7, we explore the application of BPG and other BPG based methods based on the Bregman distances proposed in this section, to solve deep matrix factorization problems. We now embark on deep non-linear neural networks with possibly more than two factors.

4.7 Bregman distances for deep neural networks - Regression setting

Deep non-linear neural networks form a major chunk of the research in the field of machine learning in the recent times [77, 96, 105, 146]. This is due to the state of the art performance attained by deep neural networks in various research areas of machine learning, such as computer vision, natural language processing and many others. For an introduction to deep neural networks, we recommend the reader to the book [77]. Deep non-linear neural networks rely on so-called non-linear activation functions, whereas in deep matrix factorization setting we use only the linear activation functions. Our focus here is on the regression problems that arise in the context of deep learning, whereas in the next section we focus on the classification problems.

We describe below the objective that arises in regression setting with deep (non-linear) neural networks. Denote $\mathcal{W}_i = \mathbb{R}^{d_{i+1}\times d_i}$ where $d_i \in \mathbb{N}$ for all $i \in \{1, 2, \ldots, N\}$ where N is a positive integer such that $N \geq 2$. Also, $d_{N+1} = d$, $X \in \mathbb{R}^{d_1 \times n_T}$, $Y \in \mathbb{R}^{d\times n_T}$ be fixed, where $n_T \in \mathbb{N}$. Typically, Y denotes the labels/targets for the inputs X and n_T corresponds to the number of training samples. Moreover, $W := (W_1, \ldots, W_N)$ and $W \in \mathcal{W} := \mathcal{W}_1 \times \cdots \times \mathcal{W}_N$. Also, we denote $\|W\|_F^2 := \sum_{i=1}^N \|W_i\|_F^2$, which is the induced norm on the product space $\mathcal{W}$. The optimization problem suitable for the regression setting with deep non-linear neural networks is:

$$\min_{W_i\in\mathcal{W}_i\,\forall i\in[N]} \left\{ f_1(W) := \frac{1}{2}\left\|\sigma_N(W_N\ldots\sigma_1(W_1X)) - Y\right\|_F^2 \right\}\,, \tag{4.7.1}$$

where $\sigma_i : \mathbb{R} \to \mathbb{R}$ for $i \in \{1, \ldots, N\}$ are activation functions, which are applied element-wise. We will discuss the exact properties of the activation functions later in this section. Note that when $\sigma(x) = x$, we obtain the deep matrix factorization setting.

4.7.1 Activation functions

Henceforth, we consider the following assumption on the so-called activation functions $\sigma_1, \ldots, \sigma_N$.

Assumption B. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an activation function that is twice continuously differentiable. There exist certain constants $D, E, F > 0$, $C \geq 0$ such that the following conditions hold true for any $x \in \mathbb{R}$:

$$\sigma(x) \leq C|x| + D\,, \quad \sigma'(x) \leq E\,, \quad \sigma''(x) \leq F\,.$$

The following are examples of popular activation functions that satisfy Assumption B.

Sigmoid activation function. The sigmoid activation function $\sigma_1 : \mathbb{R} \to \mathbb{R}$ is given by

$$\sigma_1(t) = \frac{1}{1 + e^{-t}}\,.$$

The first and second order derivatives of σ_1 are given by

$$\sigma_1'(t) = \sigma_1(t)(1 - \sigma_1(t))\,, \quad \sigma_1''(t) = \sigma_1'(t) - 2\sigma_1'(t)\sigma_1(t)\,.$$

It is easy to see that σ_1 satisfies Assumption B with $C = 0$, $D = 1$, $E = 1$, $F = 1$.

Tanh activation function. The tanh activation function is $\sigma_2 : \mathbb{R} \to \mathbb{R}$ is given by

$$\sigma_2(t) = \tanh(t)\,.$$

It's first and second order derivatives are given by

$$\sigma_2'(t) = 1 - \sigma_2(t)^2\,, \quad \sigma_2''(t) = -2\sigma_2(t)(1 - \sigma_2(t)^2)\,.$$

It is easy to see that σ_2 satisfies Assumption B with $C = 0$, $D = 1$, $E = 1$, $F = \frac{4}{3\sqrt{3}}$.

Softplus activation function. Let $\alpha > 0$, the softplus activation function is $\sigma_3 : \mathbb{R} \to \mathbb{R}$ is given by

$$\sigma_3(t) = \frac{1}{\alpha}\log(1 + e^{\alpha t})\,.$$

It's first and second order derivatives are given by

$$\sigma_3'(t) = \frac{e^{\alpha t}}{1 + e^{\alpha t}}\,, \quad \sigma_3''(t) = \alpha\frac{e^{\alpha t}}{1 + e^{\alpha t}} - \alpha\frac{e^{2\alpha t}}{(1 + e^{\alpha t})^2} = \alpha\frac{e^{\alpha t}}{(1 + e^{\alpha t})^2}\,.$$

Following the calculation in [129], we deduce that $\sigma_3(t) \leq \frac{\log 2}{\alpha} + |t|$. It is easy to see that σ_3 satisfies Assumption B with $C = 1$, $D = \frac{\log 2}{\alpha}$, $E = 1$, $F = \alpha$.

4.7.2 Regression setting

Recall the following Generalized AM-GM inequality. Let $a_1, \ldots, a_N$ be non-negative real numbers then the following inequality holds true

$$a_1 a_2 \ldots a_N \leq \left(\frac{a_1 + a_2 + \ldots + a_N}{N}\right)^N\,.$$

The mapping $S_N : W \to \mathbb{R}^{d_1 \times n_T}$ is given by

$$S_N(W_1, \ldots, W_N) := \sigma_N(W_N \ldots \sigma_1(W_1 X)).$$

Similarly, we denote the mappings $S_{N-1}, \ldots, S_1$ and $S_0 := X$.

In order to analyse the L-smad property, we need to be aware of the second order terms that arise in the Taylor expansion of the objective. The objective in (4.7.1) relies on S_N. Thus, we initially consider both the first and second order terms of S_N, using the following result. For the purpose of ease of understanding, we now consider $N = 2$. However, we later consider a generic positive integer N.

Lemma 4.7.2.1. *Consider $N = 2$ and let $H_1 \in \mathcal{W}_1$ and $H_2 \in \mathcal{W}_2$. Consider the mapping $S_2(W_1, W_2) := \sigma_2(W_2 \sigma_1(W_1 X))$ and in the expansion $S_2(W_1 + H_1, W_2 + H_2)$, the first order term containing H_1 is given by*

$$\sigma_2'(W_2\sigma_1(W_1X)) \circ (W_2(\sigma_1'(W_1X) \circ (H_1X)))\,,$$

the first order term containing H_1 is given by

$$\sigma_2'(W_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X))\,,$$

the second order term containing H_1 is given by

$$\sigma_2'(W_2\sigma_1(W_1X)) \circ \left(W_2 \left(\frac{1}{2}\sigma_1''(W_1X) \circ (H_1X) \circ (H_1X)\right)\right)\,,$$

the second order term containing H_2 is given by

$$\frac{1}{2}\sigma_2''(W_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X))\,,$$

and the second order term which couples H_1 and H_2 is given by

$$\sigma_2'(W_2\sigma_1(W_1X)) \circ (H_2(\sigma_1'(W_1X) \circ (H_1X)))\,.$$

Proof. Considering the expansion

$$S_2(W_1 + H_1, W_2 + H_2) := \sigma_2((W_2 + H_2)\sigma_1((W_1 + H_1)X))\,.$$

We find the first order term containing H_1 of the above given expansion by setting $H_2 = 0$. Similarly, we obtain the first order term containing H_2 by setting $H_1 = 0$. We provide the calculation for first order term containing H_1. Considering the first order expansion of $\sigma_1((W_1 + H_1)X)$ ignoring the higher order terms, we obtain the following:

$$\sigma_1(W_1X + H_1X) = \sigma_1(W_1X) + \sigma_1'(W_1X) \circ (H_1X)\,,$$

where we used the fact that σ_1 is applied element-wise. Then, we obtain the following:

$$\begin{aligned}
&\sigma_2(W_2\sigma_1(W_1X + H_1X)) \\
&= \sigma_2(W_2(\sigma_1(W_1X) + \sigma_1'(W_1X) \circ (H_1X)))\,, \\
&= \sigma_2(W_2\sigma_1(W_1X) + W_2(\sigma_1'(W_1X) \circ (H_1X)))\,, \\
&= \sigma_2(W_2\sigma_1(W_1X)) + \sigma_2'(W_2\sigma_1(W_1X)) \circ (W_2(\sigma_1'(W_1X) \circ (H_1X)))\,.
\end{aligned}$$

where in the first step we used the first order expansion of σ_1 and in the last step we used the first order expansion of σ_2. Similarly, to find the first order term containing H_2, we set $H_1 = 0$. Then, we obtain the following:

$$\begin{aligned}&\sigma_2((W_2 + H_2)\sigma_1(W_1X))\\ &= \sigma_2(W_2\sigma_1(W_1X) + H_2\sigma_1(W_1X))\,,\\ &= \sigma_2(W_2\sigma_1(W_1X)) + \sigma_2'(W_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X))\,,\end{aligned}$$

where in the last step we used the first order expansion of σ_2. In order to find the second order term containing only H_1, using the following second order expansion

$$\sigma_1(W_1X + H_1X) = \sigma_1(W_1X) + \sigma_1'(W_1X) \circ (H_1X) + \frac{1}{2}\sigma_1''(W_1X) \circ (H_1X) \circ (H_1X)\,.$$

We are only interested in the second order term, thus we ignore the $\sigma_1'(W_1X) \circ (H_1X)$ in the above expansion. Now, we obtain

$$\begin{aligned}&\sigma_2(W_2(\sigma_1(W_1X) + \frac{1}{2}\sigma_1''(W_1X) \circ (H_1X) \circ (H_1X)))\\ &= \sigma_2\left(W_2\sigma_1(W_1X) + W_2\left(\frac{1}{2}\sigma_1''(W_1X) \circ (H_1X) \circ (H_1X)\right)\right)\,,\\ &= \sigma_2(W_2\sigma_1(W_1X)) + \left(\sigma_2'(W_2\sigma_1(W_1X)) \circ \left(W_2\left(\frac{1}{2}\sigma_1''(W_1X) \circ (H_1X) \circ (H_1X)\right)\right)\right)\,,\end{aligned}$$

where in the last step we used second order Taylor expansion of σ_2 element-wise. Now, considering the second order expansion containing H_2 we obtain

$$\begin{aligned}&\sigma_2((W_2 + H_2)\sigma_1(W_1X))\\ &= \sigma_2(W_2\sigma_1(W_1X) + H_2\sigma_1(W_1X))\,,\\ &= \sigma_2(W_2\sigma_1(W_1X)) + \sigma_2'(W_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X)) + \frac{1}{2}\sigma_2''(W_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X)) \circ (H_2\sigma_1(W_1X))\,.\end{aligned}$$

In order to find the coupling term containing H_1, H_2, we consider the following

$$\begin{aligned}&\sigma_2((W_2 + H_2)\sigma_1(W_1X + H_1X))\\ &= \sigma_2((W_2 + H_2)(\sigma_1(W_1X) + \sigma_1'(W_1X) \circ (H_1X)))\,,\\ &= \sigma_2(W_2\sigma_1(W_1X) + H_2(\sigma_1'(W_1X) \circ (H_1X)))\,,\\ &= \sigma_2(W_2\sigma_1(W_1X)) + \sigma_2'(W_2\sigma_1(W_1X)) \circ (H_2(\sigma_1'(W_1X) \circ (H_1X)))\,.\end{aligned}$$

Thus, we arrive at the proposed result. □

Using the same logic as Lemma 4.7.2.1, we obtain the following result for a generic positive integer N.

Lemma 4.7.2.2. *Let $H_i \in \mathcal{W}_i$, for $i \in \{1, \ldots, N\}$. Considering the following expansion*

$$S_N(W_1 + H_1, \ldots, W_N + H_N)\,,$$

the first order term is given by $\Delta_{i,N}$ for $i \in \{1, \ldots, N\}$ in (A.4.1), the second order terms are given by $\Delta_{i,j,N}$ for $i, j \in \{1, \ldots, N\}$ in (A.4.2) and $\Delta_{i,i,N}$ for $i \in \{1, \ldots, N\}$ in (A.4.3).

The proof is provided in Section A.4 in the appendix.

Henceforth, we use the notions provide in Lemma 4.7.2.2. Now, we consider the first and second order terms that arise in the Taylor expansion of the objective in (4.7.1).

Lemma 4.7.2.3. *Let $H_i \in \mathcal{W}_i$, for $i \in \{1, \ldots, N\}$. Consider the following expansion*

$$f_1(W_1 + H_1, \ldots, W_N + H_N)\,,$$

then there exist $\Lambda_u \geq 0$ for $u \in \{1, \ldots, N\}$ such that the second order form is given by

$$\frac{1}{2}\langle (H_1, \ldots, H_N), \nabla^2 f_1(W)(H_1, \ldots, H_N)\rangle \leq \left(\sum_{u=0}^{N} \Lambda_u \left(\sum_{p=1}^{N} \|W_p\|^2\right)^u\right)\left(\sum_{i=1}^{N} \|H_i\|^2\right).$$

The proof is provided in Section A.5 in the appendix.

We record the following variant of Lemma A.3.0.2. The following result states that first order term and bounds on second order terms that arise in the Taylor expansion of the function H defined below.

Lemma 4.7.2.4. *Let $H_i \in \mathcal{W}_i$, for $i \in \{1, \ldots, N\}$. Consider the following kernel generating distance:*

$$H(W_1, \ldots, W_N) := \left(\frac{\sum_{j=1}^{N} \|W_j\|^2}{N}\right)^N.$$

It's gradient with respect to W_i for $i \in \{1, \ldots, N\}$ is given by

$$\nabla_{W_i} H(W) = \frac{2}{N^N}\binom{N}{N-1, 1}\left(\sum_{j=1}^{N} \|W_j\|^2\right)^{N-1} W_i\,,$$

and the following lower bound holds true

$$\big\langle (H_1, \ldots, H_N), \nabla^2 H(W_1, \ldots, W_N)(H_1, \ldots, H_N)\big\rangle \geq 2\left(\frac{\sum_{j=1}^{N} \|W_j\|^2}{N}\right)^{N-1}\left(\sum_{k=1}^{N} \|H_k\|_F^2\right),$$

and the following upper bound holds true

$$\big\langle (H_1, \ldots, H_N), \nabla^2 H(W_1, \ldots, W_N)(H_1, \ldots, H_N)\big\rangle \leq \left(\frac{2(2N-1)}{N^{N-1}}\right)\left(\sum_{k=1}^{N} \|H_k\|_F^2\right)\left(\sum_{k=1}^{N} \|W_k\|_F^2\right)^{N-1}.$$

The proof follows from the proof of Lemma A.3.0.2. Based on the above result, we have the following lemma which is crucial to prove the L-smad property.

Lemma 4.7.2.5. *Based on the notions in Lemma 4.7.2.3, consider the following kernel generating distance:*

$$h(W) := \sum_{u=1}^{2N} \Gamma_u \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{u}\right)^u, \tag{4.7.2}$$

where

$$\Gamma_u = \begin{cases} \left(\sum_{i=1}^{9} \Theta_i\right) (u)^{u-1}\,, & \text{if } 1 \leq u \leq N-1\,, \\ (\Theta_1 + \Theta_2 + \Theta_3 + \Theta_4 + \Theta_5 + \Theta_6)\, N^{N-1}\,, & \text{if } u = N\,, \\ (\Theta_5 + \Theta_9)\,(N+1)^N\,, & \text{if } u = N+1, \\ \Theta_5\,, & \text{if } u \in \{N+2, \ldots, 2N\}\,. \end{cases} \tag{4.7.3}$$

Then, it's gradient with respect to W_i for $i \in \{1, \ldots, N\}$ is given by

$$\nabla_{W_i} h(W) = \sum_{u=1}^{2N} \frac{2\Gamma_u}{u^u} \binom{u}{u-1, 1} \left(\sum_{j=1}^{N} \|W_j\|^2\right)^{u-1} W_i\,.$$

Also, the following lower bound holds true:

$$\left\langle (H_1, \ldots, H_N), \nabla^2 h(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \sum_{u=1}^{2N} 2\Gamma_u \left(\sum_{i=1}^{N} \|H_i\|_F^2\right) \left(\frac{\sum_{j=1}^{N} \|W_j\|^2}{u}\right)^{u-1}\,,$$

and the following upper bound holds true:

$$\left\langle (H_1, \ldots, H_N), \nabla^2 h(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \leq \sum_{u=1}^{2N} (2\Gamma_u(2u-1)) \left(\sum_{k=1}^{N} \|H_k\|_F^2\right) \left(\frac{\sum_{j=1}^{N} \|W_j\|^2}{u}\right)^{u-1}\,.$$

Proof. The proof is a simple consequence of Lemma 4.7.2.4 and Lemma 4.7.2.3. □

Note that h is strongly convex as $\Gamma_1 \neq 0$. Now, we provide our main result that is the L-smad property.

Proposition 4.7.2.1 (L-smad property)**.** *Consider h in* (4.7.2) *and f_1 in* (4.7.1)*, then the function $Lh - f_1$ is convex with $L = 1$.*

Proof. Combining the results of Lemma 4.7.2.5 and Lemma 4.7.2.3, we obtain

$$\frac{1}{2} \left\langle (H_1, \ldots, H_N), \nabla^2 h(W)(H_1, \ldots, H_N) \right\rangle \geq \frac{1}{2} \left\langle (H_1, \ldots, H_N), \nabla^2 f_1(W)(H_1, \ldots, H_N) \right\rangle\,.$$

□

A similar calculation leads to the convexity of $\underline{L}h + f_1$ for certain $\underline{L}$, however, such a condition is not crucial for this chapter. Thus, we skip it. In Chapter 8, we explore the application of BPG and other BPG based methods based on the Bregman distances proposed in this section.

4.8 Bregman distances for deep neural networks - Classification setting

Classification problems based on deep neural networks are very popular in machine learning and related fields. Several practical objectives such as hand written recognition [99], image classification [96], spam detection [168] and many other problems [77] rely on classification problems. In classification setting, essentially we are given a training data with inputs and corresponding class labels. The goal is to develop a classifier (function) where an input is passed to obtain the class label. We now describe the objective that arises in the classification setting. For the purpose of self-containedness of the chapter, we repeat the text from Chapter 1. Let K be the number of classes. Given a training dataset with M inputs, denoted $x_j \in \mathbb{R}^{d_1}$ for

$j \in \{1, \ldots, M\}$, and the corresponding class j_k in $\{1, 2 \ldots, K\}$ for each input. Continuing the notation in the regression setting, x_j is the j^{th} column of X and set $K = d$, $M = n_T$. Here, the label for the j^{th} sample would be $y_j \in \mathbb{R}^N$, such that all the elements are zero except the j_k^{th} element which is set to one. The goal is find a model which uses this training dataset to predict the class labels for new unseen datapoints. In this regard, we consider the following objective:

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in [N]} \left\{ f_1(W) := \sum_{j=1}^{M} \left(-\log \left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^{K} e^{z_{j,k}}} \right) \right) \right\} . \tag{4.8.1}$$

where the vector $z_j \in \mathbb{R}^N$ is generated via certain deep neural network, which can be possibly be a linear network or a non-linear network for the j^{th} sample and z_{j,j_k} is the j_k^{th} coordinate of z_j, j_k denotes the class of j^{th} sample and it lies in $\{1, 2 \ldots, K\}$. For $j \in \{1, \ldots, N\}$, with deep linear neural networks we have $z_j = W_1 \ldots W_N x_j$, and with generic deep non-linear neural network we have $z_j := \sigma_N(W_N \ldots \sigma_1(W_1 x_j))$.

We recall few properties of the cross-entropy loss given above. Let $i \in \{1, \ldots, K\}$ and consider the following function $\tilde{f} : \mathbb{R}^K \to \mathbb{R}$ given by

$$\tilde{f}(x) = -\log \left(\frac{e^{x_i}}{\sum_{i=1}^{K} e^{x_i}} \right) . \tag{4.8.2}$$

For convenience denote $\tilde{S}_i(x) := \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$. The gradient of $\tilde{S}_i$ for $i \in \{1, \ldots, K\}$ is given by

$$(\nabla \tilde{S}_i(x))_j = \begin{cases} \tilde{S}_i(x))(1 - \tilde{S}_i(x)) & \text{if } j = i \,, \\ -\tilde{S}_i(x)\tilde{S}_j(x) & \text{if } j \neq i \,. \end{cases} \tag{4.8.3}$$

Thus, the gradient of $\tilde{f}$ is given by

$$(\nabla \tilde{f}(x))_j = \begin{cases} -(1 - \tilde{S}_j(x)) & \text{if } j = i \,, \\ \tilde{S}_j(x) & \text{if } j \neq i \,. \end{cases} \tag{4.8.4}$$

Note that we have the following:

$$\left\| \nabla^2 \tilde{f}(x) \right\|_F \leq 2\sqrt{K} \,, \quad \forall x \in \mathbb{R}^K .$$

This is because of the following manipulations:

$$\begin{aligned} \left\langle h, \nabla^2 \tilde{f}(x) h \right\rangle &= \sum_{j=1}^{K} h_j \left\langle \nabla \tilde{S}_j(x), h \right\rangle , \\ &\leq \|h\| \sqrt{\sum_{j=1}^{K} \left\langle \nabla \tilde{S}_j(x), h \right\rangle^2} , \\ &\leq \|h\|^2 \sqrt{\sum_{j=1}^{K} \left\| \nabla \tilde{S}_j(x) \right\|^2} , \\ &\leq 2\sqrt{K} \, \|h\|^2 \, , \end{aligned}$$

where $\left\|\nabla\tilde{S}_j(x)\right\| \leq 2$ and in the second last step we use Cauchy-Schwarz inequality.

4.8.1 Deep linear neural networks

We first consider the deep linear neural network model via the following result.

Lemma 4.8.1.1. *Denote the following*

$$f_1(W) := \sum_{j=1}^{M} \left(-\log\left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^{K} e^{z_{j,k}}} \right) \right) , \tag{4.8.5}$$

where $z_j = W_1 \dots W_N x_j$ for all $j = 1, \dots, M$. Let $H_i \in \mathcal{W}_i$, for $i \in \{1, \dots, N\}$. Consider the following expansion

$$f_1(W_1 + H_1, \dots, W_N + H_N) ,$$

then there exist $\Lambda_u \geq 0$ for $u \in \{1, \dots, N\}$ such that the second order form is given by

$$\frac{1}{2}\langle (H_1, \dots, H_N), \nabla^2 f_1(W)(H_1, \dots, H_N)\rangle \tag{4.8.6}$$

$$\leq \frac{1}{2} 2\sqrt{K} N \left(\sum_{j=1}^{M} \|x_j\|^2 \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right) \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{N-1} \right)^{N-1} . \tag{4.8.7}$$

Proof. Considering the second order term of $f_1(W_1 + H_1, \dots, W_N + H_N)$ we have the following calculation after simple manipulations:

$$\begin{aligned}
&\frac{1}{2}\langle (H_1, \dots, H_N), \nabla^2 f_1(W)(H_1, \dots, H_N)\rangle \\
&\leq \frac{1}{2} 2\sqrt{K} \sum_{j=1}^{M} \left\| \sum_{i=1}^{N} \Delta_{i,N}^{(j)} \right\|^2 , \\
&\leq \frac{1}{2} 2\sqrt{K} \sum_{j=1}^{M} \left\| \sum_{i=1}^{N} \left(\prod_{p=1}^{i-1} W_p \right) H_i \left(\prod_{p=i+1}^{N} W_p \right) x_j \right\|^2 , \\
&\leq \frac{1}{2} 2\sqrt{K} N \sum_{j=1}^{M} \sum_{i=1}^{N} \|x_j\|^2 \left(\prod_{p=1}^{i-1} \|W_p\|^2 \right) \|H_i\|^2 \left(\prod_{p=i+1}^{N} \|W_p\|^2 \right) , \\
&\leq \frac{1}{2} 2\sqrt{K} N \left(\sum_{j=1}^{M} \|x_j\|^2 \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right) \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{N-1} \right)^{N-1} .
\end{aligned}$$

□

Lemma 4.8.1.2. *Based on the notions in Lemma 4.7.2.3, consider the following kernel generating distance:*

$$h(W) := \widehat{\Gamma}_N \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{N} \right)^N \tag{4.8.8}$$

where

$$\widehat{\Gamma}_N = 2\sqrt{K}\frac{N\left\|X\right\|^2}{2(N-1)^{N-1}}N^N \quad \text{if} \quad 1 \leq j \leq N\,. \tag{4.8.9}$$

Then, it's gradient with respect to W_i *for* $i \in \{1,\dots,N\}$ *is given by*

$$\nabla_{W_i} h(W) = \frac{2\widehat{\Gamma}_N}{N^N}\binom{N}{N-1,1}\left(\sum_{j=1}^{N}\left\|W_j\right\|^2\right)^{N-1} W_i\,.$$

Also, the following lower bound holds true:

$$\left\langle (H_1,\dots,H_N), \nabla^2 h(W_1,\dots,W_N)(H_1,\dots,H_N)\right\rangle \geq 2\hat{\Gamma}_N\left(\sum_{i=1}^{N}\left\|H_i\right\|_F^2\right)\left(\frac{\sum_{j=1}^{N}\left\|W_j\right\|^2}{N}\right)^{N-1},$$

and the following upper bound holds true:

$$\left\langle (H_1,\dots,H_N), \nabla^2 h(W_1,\dots,W_N)(H_1,\dots,H_N)\right\rangle \leq \left(2\hat{\Gamma}_N(2N-1)\right)\left(\sum_{k=1}^{N}\left\|H_k\right\|_F^2\right)\left(\frac{\sum_{j=1}^{N}\left\|W_j\right\|^2}{N}\right)^{N-1}.$$

Proof. The proof is a simple consequence of Lemma 4.7.2.4 and Lemma 4.7.2.3. □

Using the above notions and let $\rho > 0$, denote the following:

$$h(W) := \widehat{\Gamma}_N\left(\frac{\sum_{p=1}^{N}\left\|W_p\right\|^2}{N}\right)^N + \frac{\rho}{2}\left\|W\right\|^2, \tag{4.8.10}$$

where the additional quadratic term is required in order for the strong convexity to hold. The following results states the L-smad property.

Proposition 4.8.1.1 (L-smad property)**.** *Consider* h *in* (4.8.10) *and* f_1 *in* (4.8.5), *then the function* $Lh - f_1$ *is convex with* $L = 1$.

Proof. Combining the above results we obtain

$$\frac{1}{2}\left\langle (H_1,\dots,H_N), \nabla^2 h(W)(H_1,\dots,H_N)\right\rangle \geq \frac{1}{2}\left\langle (H_1,\dots,H_N), \nabla^2 f_1(W)(H_1,\dots,H_N)\right\rangle.$$

□

4.8.2 Deep non-linear neural networks

Using the same notions as before and set $d_N = K$, we consider the following optimization problem:

$$\min_{W_i\in\mathcal{W}_i\,\forall i\in[N]} f_1(W) := \sum_{j=1}^{M}\left(-\log\left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^{K}e^{z_{j,k}}}\right)\right), \tag{4.8.11}$$

where z_{j,j_k} is the j_k^{th} coordinate of z_j, j_k denotes the class in $\{1,2\dots,K\}$ to which the sample x_j belongs to, and $z_j := \sigma_N(W_N\dots\sigma_1(W_1x_j))$ for $j \in \{1,\dots,M\}$. Note that z_j is obtained via a deep non-linear neural network.

Lemma 4.8.2.1. *Let $H_i \in \mathcal{W}_i$, for $i \in \{1, \dots, N\}$. Consider the following expansion*

$$f_1(W_1 + H_1, \dots, W_N + H_N)\,,$$

then there exist $\Lambda_u \geq 0$ for $u \in \{1, \dots, N\}$ such that the second order form is given by

$$\frac{1}{2}\langle (H_1, \dots, H_N), \nabla^2 f_1(W)(H_1, \dots, H_N)\rangle \leq 2\sqrt{K}\Theta_1 \left(\sum_{j=0}^{N-1} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-j-1} \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right) .$$

Proof. Considering the second term of $f_1(W_1 + H_1, \dots, W_N + H_N)$ we have

$$\begin{aligned}
&\frac{1}{2}\langle (H_1, \dots, H_N), \nabla^2 f_1(W)(H_1, \dots, H_N)\rangle \\
&\leq \frac{1}{2} 2\sqrt{K} \sum_{j=1}^{M} \left\| \sum_{i=1}^{N} \Delta_{i,N}^{(j)} \right\|^2 , \\
&\leq 2\sqrt{K} \frac{N}{2} \sum_{j=1}^{M} \sum_{i=1}^{N} \left\| \Delta_{i,N}^{(j)} \right\|^2 , \\
&\leq 2\sqrt{K}\Theta_1 \left(\sum_{j=0}^{N-1} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-j-1} \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right) ,
\end{aligned}$$

where $\Delta_{i,N}^{(j)}$ denotes the expression of $\Delta_{i,N}$ with X replaced by x_j and Θ_1 is exactly as in (A.5.2). □

Lemma 4.8.2.2. *Based on the notions in Lemma 4.7.2.3, consider the following kernel generating distance:*

$$h(W) := \sum_{j=1}^{N} \tilde{\Gamma}_j \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{j} \right)^j \tag{4.8.12}$$

where

$$\tilde{\Gamma}_j = 2\sqrt{K}\Theta_1 j^j \quad \text{if} \quad 1 \leq j \leq N\,. \tag{4.8.13}$$

Then, it's gradient with respect to W_i for $i \in \{1, \dots, N\}$ is given by

$$\nabla_{W_i} h(W) = \sum_{u=1}^{N} \frac{2\tilde{\Gamma}_u}{u^u} \binom{u}{u-1, 1} \left(\sum_{j=1}^{N} \|W_j\|^2 \right)^{u-1} W_i\,.$$

Also, the following lower bound holds true:

$$\big\langle (H_1, \dots, H_N), \nabla^2 h(W_1, \dots, W_N)(H_1, \dots, H_N)\big\rangle \geq \sum_{u=1}^{N} 2\tilde{\Gamma}_u \left(\sum_{i=1}^{N} \|H_i\|_F^2 \right) \left(\frac{\sum_{j=1}^{N} \|W_j\|^2}{u} \right)^{u-1} ,$$

and the following upper bound holds true:

$$\left\langle (H_1,\ldots,H_N), \nabla^2 h(W_1,\ldots,W_N)(H_1,\ldots,H_N)\right\rangle \le \sum_{u=1}^{N} \left(2\tilde{\Gamma}_u(2u-1)\right)\left(\sum_{k=1}^{N}\|H_k\|_F^2\right)\left(\frac{\sum_{j=1}^{N}\|W_j\|^2}{u}\right)^{u-1} .$$

Proof. The proof is a simple consequence of Lemma 4.7.2.4 and Lemma 4.7.2.3. □

Note that (4.8.12) is strongly convex for $N \geq 2$.

Proposition 4.8.2.1 (L-smad property)**.** *Consider h in* (4.8.12) *and f_1 in* (4.8.11)*, then the function $Lh - f_1$ is convex with $L = 1$.*

Proof. Combining the above results we obtain

$$\frac{1}{2}\left\langle (H_1,\ldots,H_N), \nabla^2 h(W)(H_1,\ldots,H_N)\right\rangle \ge \frac{1}{2}\left\langle (H_1,\ldots,H_N), \nabla^2 f_1(W)(H_1,\ldots,H_N)\right\rangle .$$

□

In Chapter 8 we explore the application of BPG and other BPG based methods based on the Bregman distances proposed in this section to solve the classification problems arising with deep neural networks.

4.9 Chapter conclusion

In this chapter, we briefly reviewed various concepts and properties related to Bregman distances, which are generalized proximity measures. The extension of the Lipschitz continuous gradient property, namely, the L-smad property is also described. We proposed Bregman distances that are suitable for objectives that arise in the context of matrix factorization, deep matrix factorization and deep non-linear neural networks. As we will see in the later chapters, the proposed Bregman distances play a key role in the application of BPG based algorithms for the above-mentioned objectives. The ideas used to develop the Bregman distances in this chapter can be used for various other problems with similar structure, such as tensor factorization, tensor completion, matrix recovery problems, which requires further exploration.

Chapter 5

CoCaIn BPG

5.1 Abstract

Backtracking line-search is an old yet powerful strategy for finding a better step sizes to be used in Proximal Gradient algorithms. The main principle is to locally find a simple convex upper bound of the objective function, which in turn controls the step size that is used. In case of inertial Proximal Gradient algorithms, the situation becomes much more difficult and usually leads to very restrictive rules on the extrapolation parameter. In this chapter, we show that the extrapolation parameter can be controlled by locally finding also a simple concave lower bound of the objective function. This gives rise to a double convex-concave backtracking procedure which allows for an adaptive choice of both the step size and extrapolation parameters. We apply this procedure to the class of inertial Bregman Proximal Gradient methods, and prove that any sequence generated by these algorithms converges globally to a critical point of the function at hand. Numerical experiments on a number of challenging non-convex problems in image processing and machine

learning were conducted and show the power of combining inertial step and double backtracking strategy in achieving improved performances.

5.2 Introduction

We continue the setting from Section 4.4. Firstly, we recall the problem setting here. Consider the non-convex additive composite minimization problems, which include the sum of two extended-valued functions: a non-smooth function denoted by f_0 (possibly non-convex) and a smooth function denoted by f_1 (possibly non-convex) of the following form

$$(\mathcal{P}) \qquad \inf\left\{ f\left(x\right) \equiv f_0\left(x\right) + f_1(x) : \, x \in \overline{C} \right\},$$

where $\overline{C}$ is a nonempty, closed and convex set in $\mathbb{R}^N$. In Chapter 4, we introduced the L-smad property. In this chapter, we recall Bregman Proximal Gradient algorithm and discuss that it is suitable for above-mentioned problems. The goal of this chapter is to incorporate inertia into BPG and propose a new algorithm with global convergence guarantees, while relying on the upper and lower bounds that arise in the L-smad property.

The convexity condition in the L-smad property (Definition 4.4.1.1) easily yields an approximation of the objective function at hand by a convex function from above (majorant) and a concave function from below (minorant). In the traditional setting, where the gradient of the smooth function f_1 is Lipschitz continuous, the majorant and the minorant are quadratic functions. In this case, it is well-known that the tightness of the quadratic approximations is directly related to restrictions on the step size to be used in the algorithm. The same relation is true for the convexity condition. In addition to their global existence, these approximations can be locally improved by backtracking (line search) strategies and it is well-known that tight approximations are advantageous. The significance of the minorants is not clear, in general. The goal of this chapter is to leverage the minorant functions to incorporate inertia into the Bregman Proximal Gradient method, where the step-size is already governed by the majorants. For improved local approximations, we rely on backtracking procedures for both the upper and lower bounds using the convex-concave backtracking strategy.

We would like to give the reader a first intuition about the convex-concave backtracking strategy on a simple instance of problem $(\mathcal{P})$. In the following, we consider the following particular instance of problem $(\mathcal{P})$: $C = \mathbb{R}^N$, $f_0 \equiv 0$ and the gradient of f_1 is L-Lipschitz continuous. Even in this simpler setting, the convex-concave backtracking strategy is novel. In this smooth and non-convex setting, an update step of a classical inertial based gradient method, starting with some $x^0 \in \mathbb{R}^N$, reads as follows

$$\begin{aligned} y^k &= x^k + \gamma_k (x^k - x^{k-1}), \\ x^{k+1} &= y^k - \frac{1}{\bar{L}_k} \nabla f_1(y^k), \end{aligned}$$

where $\gamma_k \in [0,1]$, $k \in \mathbb{N}$, is an extrapolation parameter and $\bar{L}_k > 0$. If f_1 is convex and the extrapolation parameter γ_k is carefully chosen, this recovers the popular Nesterov's Accelerated Gradient Method [126] (for $f \neq 0$, again in the convex setting, see [15]). It is well-known that the gradient step above, can be equivalently written as follows

$$x^{k+1} = \operatorname{argmin}_{x\in\mathbb{R}^N} \left\{ f_1(y^k) + \left\langle \nabla f_1(y^k), x - y^k \right\rangle + \frac{\bar{L}_k}{2} \left\| x - y^k \right\|^2 \right\}.$$

For a proper $\bar{L}_k$, the function to be minimized above is a convex quadratic majorant of the function f_1 (due to the classical Descent Lemma), which is a property that is also crucial for the convergence analysis of the algorithm. Classically, $\bar{L}_k \geq L$, $k \in \mathbb{N}$, is a sufficient condition to guarantee the existence of a quadratic majorant. However, locally, i.e., between the points y^k and x^{k+1}, the parameter $\bar{L}_k$ may be significantly smaller than the global Lipschitz constant L (which will immediately affect the step size of the algorithm). More precisely, note that the Descent Lemma,

$$\left| f_1(x) - f_1(y^k) - \left\langle \nabla f_1(y^k), x - y^k \right\rangle \right| \leq \frac{L}{2} \left\| x - y^k \right\|^2, \qquad \forall\, x \in \mathbb{R}^N, \tag{5.2.1}$$

actually guarantees the existence of a quadratic minorant and a quadratic majorant that are determined by the same (global) parameter L. However, only the majorant limits the step size that is used in the algorithm. As shown in Figure 5.1, tighter approximations can be computed if the parameters of the minorant and the majorant are allowed to differ:

$$-\frac{\underline{L_k}}{2} \left\| x - y^k \right\|^2 \leq f_1(x) - f_1(y^k) - \left\langle \nabla f_1(y^k), x - y^k \right\rangle \leq \frac{\bar{L}_k}{2} \left\| x - y^k \right\|^2, \tag{5.2.2}$$

i.e., the minorant parameter $\underline{L_k}$ could be different from the majorant parameter $\bar{L}_k$.

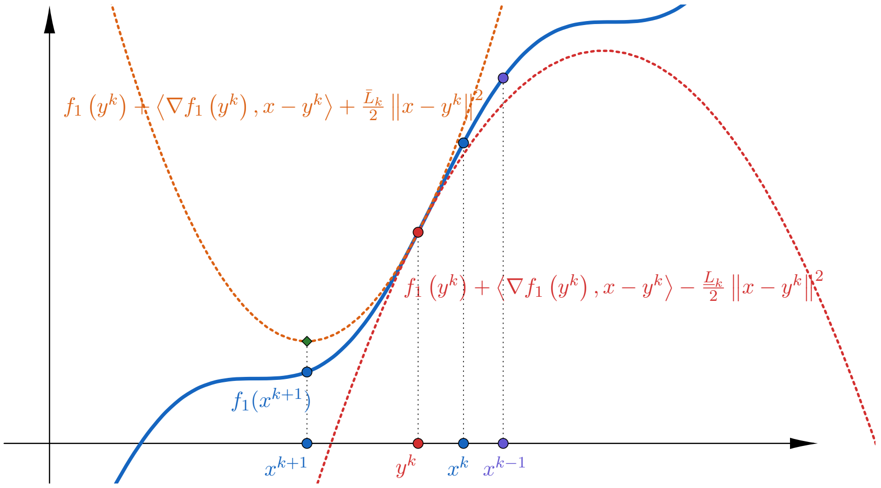


FIGURE 5.1: The inequalities in (5.2.2) guarantee that the objective function has a quadratic concave minorant and a quadratic convex majorant. The proposed convex-concave backtracking strategy locally estimates both the lower and the upper approximations using a double backtracking procedure.

While the step size of the algorithm only depends on the majorant parameter $\bar{L}_k$, the extrapolation parameter γ_k also depends on the minorant parameter $\underline{L_k}$. When $\bar{L}_k = \bar{L}$ and $\underline{L_k} = \underline{L}$, for all $k \in \mathbb{N}$, it was established in [166] that for any $0 \leq \gamma_k \leq \overline{\gamma}$, when

$$\overline{\gamma} < \sqrt{\frac{\bar{L}}{\underline{L} + \bar{L}}} \qquad \left(= \frac{1}{\sqrt{2}} \quad \text{for } \bar{L} = \underline{L} \right),$$

the generated sequence converges linearly (under certain error bound condition).

If the minorant parameter $\underline{L_k}$ is close to 0, which means that the function f_1 is “locally convex”, the extrapolation parameter γ_k can be taken close to 1, which makes the algorithm we present “similar" to an

Accelerated gradient method in the non-convex setting.

Below, we will show that using the minorant and the majorant in a local fashion (instead of their global counterparts) is very useful in developing the inertial Bregman Proximal Gradient method.

5.2.1 Contributions

Our contributions are the following.

- Interestingly, while the step size is usually restricted by the quality of the majorant, the extrapolation (also known as inertia or over-relaxation) parameter is affected by the quality of the minorant. This observation suggests to adapt the majorant and the minorant independently. In this chapter we propose an efficient backtracking strategy that locally determines a tight majorant and minorant to exploit as much information as possible from the objective function, to be used in the proposed algorithm. This leads to a highly efficient algorithm, which is able to detect "the degree of local convexity" of the objective function (see Section 5.4 for details). As the backtracking procedure seeks for tight convex majorants and concave minorants, our idea is to combine it with an inertial step.

- We propose an inertial version of the Bregman Proximal Gradient (BPG) algorithm, which uses a convex-concave backtracking procedure to dynamically adjust the step size and the extrapolation parameter. Therefore, we call our algorithm Convex-Concave Inertial BPG (CoCaIn BPG in short).

- We prove a global convergence result of this algorithm (see Section 5.6 for the details) to critical points of the objective function.

- The efficiency, which we demonstrate on several practical applications, comes from combining the inertial step with the novel convex-concave backtracking strategy, which fully exploits the power of tight local approximations in achieving large step sizes and large extrapolation parameters that can be used at the same time.

5.2.2 Related work

Our proposed algorithm belongs to the class of inertial based optimization methods. The most well-known method in this class is the so-called Heavy-ball method, which was introduced by Polyak [145] to minimize convex and smooth functions. A popular variant of the method based on Nesterov's technique (see Section 2.4.4), when applied to the additive composite model ($\mathcal{P}$) with $C = \mathbb{R}^N$, takes the following form. Start with any $x^0 = x^1 \in \mathbb{R}^N$, and generate iteratively a sequence $\{x^k\}_{k\in\mathbb{N}}$ via

$$y^k = x^k + \gamma_k(x^k - x^{k-1}), \tag{5.2.3}$$

$$x^{k+1} \in \operatorname{argmin}_x \left\{ f_0\left(x\right) + f_1(y^k) + \left\langle \nabla f_1(y^k), x - y^k \right\rangle + \frac{1}{2\tau_k} \left\| x - y^k \right\|^2 \right\}, \tag{5.2.4}$$

where $\gamma_k \in [0,1]$ is an extrapolation parameter and $\tau_k > 0$ is a step size parameter. In [137], an inertial Proximal Gradient algorithm, called iPiano, was proposed[1]. It was shown that under Assumption A, if f_0 is convex and f_1 has a globally Lipschitz continuous gradient, the sequence $\{x^k\}_{k\in\mathbb{N}}$ converges globally to a critical point (in this setting, under additional error-bound condition, a linear rate of convergence was

[1] With a small modification that the proximity term is centered around the extrapolated point y^k, while the gradient of f_1 is evaluated at x^k.

proved in [166]). The case where also the function f_0 is not necessarily convex was treated in [29, 134]. Two years later, in [144] a block version of the method, called iPALM was proposed and analyzed in the fully non-convex setting, i.e., both f_0 and f_1 are non-convex. In this case, a global convergence result to critical points was also established. A unified analysis was presented in [136]. Our goal is to incorporate the Bregman distances along with the inertial scheme mentioned in (5.2.3), (5.2.4) .

5.3 The Bregman Proximal Gradient algorithm

In this section we review the basic notations and results needed to study Bregman based optimization methods. We first recall the definition of the Bregman proximal mapping [160], which is associated with a proper and lower semi-continuous function $f : \mathbb{R}^N \to (-\infty, +\infty]$, and is defined by

$$\operatorname{prox}_f^h(x) := \operatorname{argmin}\left\{f_0\left(u\right) + D_h\left(u, x\right) : \, u \in \mathbb{R}^N\right\}, \quad \forall\, x \in \operatorname{int}\operatorname{dom} h.$$

With $h \equiv (1/2)\left\|\cdot\right\|^2$, the above boils down to the classical set-valued Moreau proximal mapping introduced in [117]. In this regard, more discussion can be found in the recent survey paper [161], and references therein. Here, we will focus on the Bregman Proximal Gradient mapping, which will take a central role in the algorithm to be developed in the next section. Given $x \in \operatorname{int}\operatorname{dom} h$ and a step size parameter $\tau > 0$, the Bregman Proximal Gradient mapping is defined by

$$\begin{aligned} T_\tau\left(x\right) &:= \operatorname{argmin}\left\{f_0\left(u\right) + f_1(u) + \left\langle \nabla f_1\left(x\right), u - x\right\rangle + \frac{1}{\tau} D_h\left(u, x\right) : \, u \in \overline{C}\right\} \\ &= \operatorname{argmin}\left\{f_0\left(u\right) + f_1(u) + \left\langle \nabla f_1\left(x\right), u - x\right\rangle + \frac{1}{\tau} D_h\left(u, x\right) : \, u \in \mathbb{R}^N\right\}, \end{aligned} \tag{5.3.1}$$

where the second equality follows from the fact that $\operatorname{dom} h \subset \overline{C}$. Note that here with $h \equiv (1/2)\left\|\cdot\right\|^2$, the above recovers the classical Proximal Gradient mapping. Now, we record below the Bregman Proximal Gradient (BPG) algorithm in Algorithm 4 from [28].

Algorithm 4: BPG: Bregman Proximal Gradient

- **Input:** $\tau > 0$.
- **Initialization:** $x^1 \in \operatorname{int}\operatorname{dom} h \cap \operatorname{dom} f_0$.
- **For each** $k \geq 1$**:** compute

$$x^{k+1} \in T_\tau(x^k) \tag{5.3.2}$$

Since f_0 could be non-convex, the mapping T_τ is not, in general, single-valued. This mapping emerges from the usual approach, which consists of linearizing the differentiable function f_1 around a point x and regularizing it with a proximal distance from that point. Similar to [28], the following assumption guarantees that the Bregman Proximal Gradient mapping is well-defined.

Assumption C. (i) The function $h + \tau f_0$ is supercoercive for all $\tau > 0$, that is,

$$\lim_{\|u\| \to \infty} \frac{h\left(u\right) + \tau f_0\left(u\right)}{\|u\|} = \infty.$$

(ii) For all $x \in C$, we have $T_{\tau}(x) \subset C$.

Assumption C(i) is a standard coercivity condition, which is for instance automatically satisfied when $\overline{C}$ is compact. On the other hand, Assumption C(ii) can be shown to hold under a classical constraint qualification condition. It also holds automatically when f_0 is convex or when $C = \mathbb{R}^N$. The following result from [28], ensures that the Bregman Proximal Gradient mapping is well-defined.

Lemma 5.3.0.1 (Well-posedness of T_τ). *Suppose that Assumptions A and C hold, and let $x \in \operatorname{int}\operatorname{dom} h$. Then, the set $T_\tau(x)$ is a nonempty and compact subset of* $\operatorname{int}\operatorname{dom} h$.

We record below the assumptions and the convergence result of BPG as stated in [28].

Theorem 5.3.0.2. *Let Assumption A and C hold. Assume that $\operatorname{dom} h = \mathbb{R}^N$, h is σ-strongly convex on $\mathbb{R}^N$, and let ∇h and ∇g be Lipschitz continuous on any bounded subset on $\mathbb{R}^N$. Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by BPG which is assumed to be bounded and let $0 < \lambda L < 1$. The following assertions hold.*

- ***Subsequential convergence.*** *Any limit point of the sequence $\{x^k\}_{k\in\mathbb{N}}$ is a critical point of f.*
- ***Global convergence.*** *Suppose that f satisfies the KL property on $\operatorname{dom} f$. Then, the sequence $\{x^k\}_{k\in\mathbb{N}}$ has finite length and converges to a critical point x^* of f.*

In essence, the above theorem states that the sequence generated by BPG converges to a single point which in turn is a critical point of the function f. A main drawback of BPG is that only the upper bound (see (4.4.1)) of the L-smad property is used and it governs the update step. The role of the lower bound in (4.4.1) is not clear. In the next section, we develop the CoCaIn BPG algorithm, which takes into consideration both the upper bound and the lower bound in (4.4.1) to perform the update and incorporate inertia. We obtain similar convergence result as BPG, while additionally gaining on the empirical performance.

We later see in Chapter 9 that BPG is restrictive and cannot be applied to the objectives with generic composite structure are out of scope. In order to enhance the applicability, we provide Model BPG algorithm in Chapter 9, which is more general than BPG and also retains the global convergence result. We now focus on proposing the inertial variant of BPG.

5.4 The inertial Bregman Proximal Gradient method

We aim to propose a Bregman variant of the method mentioned above in (5.2.3) and (5.2.4), which also handles the two involved parameter γ_k and τ_k, $k \in \mathbb{N}$, in a dynamic fashion. The update step is essentially the same as that of BPG, except that the update is performed at the extrapolated point. To this end we incorporate into our basic steps two routines aiming at controlling and updating these parameters.

5.4.1 The convex-concave backtracking procedure

As illustrated on a simple example in the introduction, the origin of this procedure comes from the fact that for smooth adaptable functions we can build lower and upper approximations as given in Lemma 4.4.1.1:

$$-\underline{L}D_h(x,y) \le f_1(x) - f_1(y) - \langle \nabla f_1(y), x-y\rangle \le \bar{L}D_h(x,y), \quad \forall\, x,y \in \operatorname{int}\operatorname{dom} h. \tag{5.4.1}$$

Even though the existence of the parameters $\underline{L}$ and $\bar{L}$ could be globally guaranteed, in practice it is often difficult or computationally expensive to evaluate them. In such cases it is recommended to apply a backtracking procedure that can locally verify the validity of the inequalities given in (5.4.1). However, in most cases only the upper approximation and the corresponding parameter $\bar{L}$ are used. Here, we will develop a double backtracking procedure that locally verifies both the lower and the upper approximations, in order to better control and update the extrapolation parameter γ_k and the step size parameter τ_k at each iteration $k \in \mathbb{N}$. To the best of our knowledge, this is the first attempt to use the lower approximation in algorithms for tackling non-convex problems. It should be noted that in the case that f_1 is convex we have by definition $\underline{L} = 0$, or even a convex quadratic lower approximation can be found when f_1 is strongly convex (see [161] for a discussion and references about a strong convexity property with respect to a Bregman distance). Based on the concepts described above, we will make the following additional assumptions on the involved functions.

Assumption D. (i) The function $h : \mathbb{R}^N \to (-\infty, +\infty]$ is σ-strongly convex on C.

(ii) The pair of functions (f_1, h) is L-smooth adaptable on C.

(iii) There exists $\alpha \leq 0$ such that $f(\cdot) - (\alpha/2)\left\|\cdot\right\|^2$ is convex[2].

A few comments on the assumption above are now in order. The first item is related to Remark 4.4.1.1, which says that the smooth adaptable property is invariant to strongly convex kernel generating distance functions h. The third assumption allows us to deal with non-convex functions f_0 since α could be negative. Also for functions that are strongly convex, we set $\alpha = 0$, as our analysis does not benefit from a positive parameter in Assumption D(iii). See Section 5.7 for examples of functions that satisfy all these assumptions. Now we are ready to present our algorithm, which is called Convex-Concave inertial (CoCaIn) Bregman Proximal Gradient (see Algorithm 5).

The two input parameters δ and ϵ are free to be chosen by the user. As we will see later the parameter ϵ measures the descent to be achieved at each iteration of the algorithm. We describe here each step of the CoCaIn BPG algorithm and defer certain implementation details to Section 5.6.4. The steps (5.4.2) and (5.4.5) are the classical steps of the Inertial Proximal Gradient Method, while here since we are dealing with the Bregman variant, it must be guaranteed that the auxiliary vector y^k as defined in (5.4.2) belongs to $\operatorname{int} \operatorname{dom} h$. Otherwise the Bregman Proximal Gradient step (5.4.5) is not defined (see Section 5.3). Even though, in general, it is not easy to guarantee that, in our case this will not be an issue. Indeed, in order to derive global convergence results of Bregman based algorithms in the non-convex setting an essential assumption seems to be that the kernel generating distance function h has a full domain, i.e., $\operatorname{dom} h = \mathbb{R}^N$ (see, for instance, [28] for more details about this limitation). The steps (5.4.4) and (5.4.6) implement the double backtracking procedure (see Section 5.6.4). The step (5.4.3) is designed to control the extrapolation parameter γ_k, $k \in \mathbb{N}$, and should be validated at each iteration. However, a natural question would be if such a parameter always exists? We postpone the positive answer to this question, to Section 5.5, and conclude this section with a list of our theoretical contributions.

5.5 Well-posedness of CoCaIn BPG

Now, we verify the well-posedness of the CoCaIn BPG algorithm. An important tool in achieving our goal is the recently introduced symmetry coefficient of a Bregman distance, which measures the lack of symmetry in $D_h(\cdot, \cdot)$, see [10].

[2] Such functions are called semi-convex with modulus α (see [134, 135]).

Algorithm 5: CoCaIn BPG: Convex-Concave inertial BPG

- **Input.** $\delta, \epsilon > 0$ with $1 > \delta > \epsilon$.
- **Initialization:** $x^0 = x^1 \in \operatorname{int}\operatorname{dom} h \cap \operatorname{dom} f_0$, $\bar{L}_0 > \frac{-\alpha}{(1-\delta)\sigma}$ and $\tau_0 \leq \bar{L}_0^{-1}$.
- **For each** $k \geq 1$**:** compute

  $$y^k = x^k + \gamma_k(x^k - x^{k-1}) \in \operatorname{int}\operatorname{dom} h, \tag{5.4.2}$$

  where γ_k is chosen such that

  $$(\delta - \epsilon)\, D_h(x^{k-1}, x^k) \geq (1 + \underline{L}_k \tau_{k-1})\, D_h(x^k, y^k) \tag{5.4.3}$$

  holds and such that $\underline{L}_k$ satisfies

  $$f_1(x^k) \geq f_1(y^k) + \left\langle \nabla f_1(y^k), x^k - y^k \right\rangle - \underline{L}_k D_h(x^k, y^k). \tag{5.4.4}$$

  Now, choose $\bar{L}_k \geq \bar{L}_{k-1}$, set $\tau_k \leq \min\left\{\tau_{k-1}, \bar{L}_k^{-1}\right\}$ and compute

  $$x^{k+1} \in \operatorname{argmin}_u \left\{ f_0\left(u\right) + \left\langle \nabla f_1(y^k), u - y^k \right\rangle + \frac{1}{\tau_k} D_h(u, y^k) \right\} \tag{5.4.5}$$

  with $\bar{L}_k$ fulfilling

  $$f_1(x^{k+1}) \leq f_1(y^k) + \left\langle \nabla f_1(y^k), x^{k+1} - y^k \right\rangle + \bar{L}_k D_h(x^{k+1}, y^k). \tag{5.4.6}$$

Definition 5.5.0.1 (Symmetry coefficient)**.** Given $h \in \mathcal{G}(C)$, its symmetry coefficient is defined by

$$\alpha\left(h\right) := \inf\left\{ \frac{D_h\left(x, y\right)}{D_h\left(y, x\right)} : \, x, y \in \operatorname{int}\operatorname{dom} h, \, x \neq y \right\} \in [0, 1]\,.$$

An important and immediate consequence of this definition is the fact that for all $x, y \in \operatorname{int}\operatorname{dom} h$ we have

$$\alpha\left(h\right) D_h\left(x, y\right) \leq D_h\left(y, x\right) \leq \alpha\left(h\right)^{-1} D_h\left(x, y\right), \tag{5.5.1}$$

where we have adopted the convention that $0^{-1} = +\infty$ and $+\infty \times r = +\infty$ for all $r \geq 0$. Clearly, the closer is $\alpha\left(h\right)$ to 1, the more symmetric D_h is with perfect symmetry when $\alpha\left(h\right) = 1$ (which holds if and only if $h = \|\cdot\|^2$).

To this end, we need to convince the reader about the existence of γ_k, $k \in \mathbb{N}$, which satisfies (5.4.3), i.e., that

$$(\delta - \epsilon)\, D_h(x^{k-1}, x^k) \geq (1 + \underline{L}_k \tau_{k-1})\, D_h(x^k, y^k),$$

holds true. The following result provides a positive answer to the existences question and information on the relevant extrapolation parameters that satisfy this inequality.

Lemma 5.5.0.1 (General extrapolation behavior)**.** *Given $h \in \mathcal{G}(C)$ with $\alpha\left(h\right) > 0$. Let $x_1, x_2, y \in \operatorname{int}\operatorname{dom} h$ and $y := x_1 + \gamma\left(x_1 - x_2\right)$ with $\gamma \geq 0$. Then, for a given $\kappa > 0$, there exists $\gamma^* > 0$ such that*

$$D_h\left(x_1, y\right) \leq \kappa D_h\left(x_2, x_1\right), \quad \forall\, \gamma \in [0, \gamma^*]\,. \tag{5.5.2}$$

The proof of Lemma 5.5.0.1 is given in Section B.1 in the appendix.

Remark 5.5.0.1. Note that in the above lemma, γ^* depends only on the symmetry coefficient $\alpha\left(h\right)$. Therefore, for the Euclidean distance with $\alpha\left(h\right) = 1$, this implies that,

$$\gamma^* = \frac{-1+\sqrt{1+8\kappa}}{4}\,.$$

However, for the Euclidean distance, the expression in (5.5.2), can be simplified significantly. Indeed, since we take $h = (1/2)\left\|\cdot\right\|^2$, then using the fact that $y^k - x^k = \gamma_k(x^k - x^{k-1})$ we obtain that $\gamma_k \leq \sqrt{\kappa}$. In the case of CoCaIn BPG, we have the following restriction on the maximal extrapolation parameter that can be used $\gamma_k \leq \sqrt{\frac{\delta-\epsilon}{1+\underline{L}_k\tau_{k-1}}} = \sqrt{\frac{(\delta-\epsilon)\bar{L}_{k-1}}{\bar{L}_{k-1}+\underline{L}_k}}$ with $\tau_{k-1} = \bar{L}_{k-1}^{-1}$. A related bound also appeared in [166] as we discussed in the introduction. When, the values of $\underline{L}_k$ and $\bar{L}_{k-1}$ are almost equal and $\delta - \epsilon \approx 1$, then it is possible to choose the inertial parameter γ_k such that $\gamma_k \approx 1/\sqrt{2}$. We discuss more about bounds of γ_k, $k \in \mathbb{N}$, in Section 5.6.3.

5.6 Convergence analysis of CoCaIn BPG

Before we proceed to the convergence analysis, we need the following technical lemma.

Lemma 5.6.0.1 (Function descent property). *Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by CoCaIn BPG. Then, for all $k \in \mathbb{N}$, we have*

$$f(x^k) \geq f(x^{k+1}) + \frac{1}{\tau_k}D_h\left(x^k, x^{k+1}\right) + \frac{\alpha}{2}\left\|x^{k+1} - x^k\right\|^2 - \left(\frac{1}{\tau_k} + \underline{L}_k\right)D_h\left(x^k, y^k\right). \tag{5.6.1}$$

The proof of Lemma 5.6.0.1 is given in Section B.2 in the appendix.

Since we are dealing with inertial based methods, which belong to the class of non-descent methods, we can not expect to use classical convergence techniques for non-convex problems (see below for more information about it). In order to overcome the lack of descent, we will use the Lyapunov technique, which involves the construction of a sequence of new functions, which will be used to “better" measure the progress of the algorithm, where by progress we mean a decrement in the Lyapunov function values. In several cases a trivial Lyapunov function would be to use the function itself, however in the case of non-descent methods, it is not a good choice, since it does not capture well the behavior of the iterates. The behavior of two subsequent iterates must be taken into consideration along with the function, as observed in [137, 159].

5.6.1 Lyapunov function descent property of CoCaIn BPG

Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by CoCaIn BPG. We define, at iterate $k \in \mathbb{N}$, the following Lyapunov function

$$f_\delta^k\left(x^k, x^{k-1}\right) = \tau_{k-1}\left(f(x^k) - v(\mathcal{P})\right) + \delta D_h(x^{k-1}, x^k). \tag{5.6.2}$$

This Lyapunov function involves two terms: (i) the term $\tau_{k-1}\left(f(x^k) - v(\mathcal{P})\right)$, which measures the progress in original function values f with respect to the global optimal value of problem $(\mathcal{P})$ and (ii) the term given by $\delta D_h(x^{k-1}, x^k)$, which ensures that the iterates stay close enough, with respect to the Bregman distance. Before we motivate further the usage of this Lyapunov function, we show its descent property.

Proposition 5.6.1.1. *Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by CoCaIn BPG. Then, for all $k \in \mathbb{N}$, we have*

$$f_\delta^k\left(x^k, x^{k-1}\right) \geq f_\delta^{k+1}\left(x^{k+1}, x^k\right) + \epsilon D_h(x^{k-1}, x^k). \tag{5.6.3}$$

The proof of Proposition 5.6.1.1 is given in Section B.3 in the appendix.

Proposition 5.6.1.2. *Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by CoCaIn BPG. Then, the following assertions hold:*

(i) *The sequence $\left\{f_\delta^{k+1}\left(x^{k+1}, x^k\right)\right\}_{k\in\mathbb{N}}$ is nonincreasing.*

(ii) *$\sum_{k=1}^{\infty} D_h(x^{k-1}, x^k) < \infty$, and hence the sequence $\left\{D_h(x^{k-1}, x^k)\right\}_{k\in\mathbb{N}}$ converges to zero.*

(iii) *$\min_{1\leq k\leq n} D_h(x^{k-1}, x^k) \leq f_\delta^1\left(x^1, x^0\right) / (\epsilon n)$.*

The proof of Proposition 10.4.1.2 is given in Section B.4 in the appendix.
In order to proceed with the global convergence analysis of CoCaIn BPG, we will need throughout the rest of this section, to additionally assume the following.

Assumption E. (i) $\operatorname{dom} h = C = \mathbb{R}^N$.

(ii) ∇h and ∇f_1 are Lipschitz continuous on any bounded subset of $\mathbb{R}^N$.

5.6.2 Global convergence for CoCaIn BPG

In this subsection we show the global convergence result of CoCaIn BPG. The goal is to show that the whole sequence $\{x^k\}_{k\in\mathbb{N}}$, that is generated by CoCaIn BPG, converges to a critical point, in terms of the limiting subdifferential which contains only general subgradients [150, Definition 8.3]. To this end, we denote the set of critical points by

$$\operatorname{crit} f = \left\{x \in \mathbb{R}^N : \ 0 \in \partial f(x) \equiv \partial f_0(x) + \nabla f_1(x)\right\}.$$

Note that, such a set is well-defined due to Fermat's rule [150, Theorem 10.1, p. 422] and due to the concept of limiting subdifferential.

From now on we will make the following assumption regarding the sequence of majorant parameters $\left\{\bar{L}_k\right\}_{k\in\mathbb{N}}$: there exists an integer $K \in \mathbb{N}$ such that $\bar{L}_k = \bar{L}$ for all $k \geq K$ (K can be as large as the user wishes). It should be noted that thanks to Assumption D(ii) and Lemma 4.4.1.1, there exists a global majorant parameter $\bar{L}$ such that (5.4.6) holds true for all $k \in \mathbb{N}$. On the other hand, since in anyway we require that the parameters do not decrease between two successive iterations, it makes sense that at some point we will stop changing them and continue with a fixed value. However, it is very important not using the global parameter $\bar{L}$ right from the beginning since in practice the parameter $\bar{L}_k$ determined by (5.4.6) might be much smaller (especially in early stages of the algorithm).

In the second phase of the algorithm, i.e., when $k \geq K$, it also makes sense to assume that $\tau_k = \tau$ for all $k \geq K$ where $\tau \leq \bar{L}^{-1}$. This immediately suggests that our Lyapunov function can also be simplified. More precisely, we define the following new Lyapunov function:

$$f_{\delta_1}(x, y) = \begin{cases} f_\delta^k(x, y), & x = x^k,\ y = x^{k-1}, \text{for some } k < K, \\ f(x) + \delta_1 D_h(y, x), & \text{otherwise}, \end{cases} \tag{5.6.4}$$

where $\delta_1 = \delta/\tau$.

The global convergence result is based on showing that CoCaIn BPG generates a gradient-like descent sequence according to Definition 5.6.2.1 (see below). This involves three properties which need to be verified: "sufficient descent condition", "relative error condition" and "continuity condition". Such a convergence analysis is based on a recent technique, which was initiated by Attouch and Bolte [6], and later on was simplified and unified in [26]. A more general framework was proposed in [136].

The main tool that stands behind this technique is the Kurdyka-Łojasiewicz (KL) property [97, 106] (see [22] for the non-smooth case), which is properly defined before in Chapter 3. In order to derive the global convergence of our algorithm, we prove that the sequence generated by BPG is a gradient-like descent sequence, which we recall below. For the interested readers we refer to [28, Appendix 6, p. 2147], where a short and self-contained summary of this proof methodology can be found. It should be noted again that here we consider a modification, which fits non-descent methods like CoCaIn BPG.

Definition 5.6.2.1 (Gradient-like descent sequence)**.** A sequence $\{x^k\}_{k\in\mathbb{N}}$ is called a gradient-like descent sequence for minimizing f_{δ_1} if the following three conditions hold:

(C1) Sufficient decrease condition. There exists a positive scalar ρ_1 such that

$$\rho_1 \left\|x^k - x^{k-1}\right\|^2 \leq f_{\delta_1}\left(x^k, x^{k-1}\right) - f_{\delta_1}\left(x^{k+1}, x^k\right), \quad \forall\ k \in \mathbb{N}.$$

(C2) Relative error condition. There exist an integer $K \in \mathbb{N}$ and a positive scalar ρ_2 such that

$$\left\|w^{k+1}\right\| \leq \rho_2 \left(\left\|x^k - x^{k-1}\right\| + \left\|x^{k+1} - x^k\right\|\right), \quad w^{k+1} \in \partial f_{\delta_1}\left(x^{k+1}, x^k\right), \quad \forall\ k \geq K.$$

(C3) Continuity condition. Let $\overline{x}$ be a limit point of a subsequence $\left\{x^k\right\}_{k\in\mathcal{K}}$, then $\limsup_{k\in\mathcal{K}\subset\mathbb{N}} f(x^k) \leq f\left(\overline{x}\right)$.

Based on Definition 5.6.2.1 and the KL property, the following global convergence result holds true. We provide its proof in Section B.5 in the appendix.

Theorem 5.6.2.1 (Global convergence)**.** *Let $\{x^k\}_{k\in\mathbb{N}}$ be a bounded gradient-like descent sequence for minimizing f_{δ_1}. If f satisfies the KL property, then the sequence $\{x^k\}_{k\in\mathbb{N}}$ has finite length, i.e., $\sum_{k=1}^{\infty} \left\|x^{k+1} - x^k\right\| < \infty$ and it converges to $x^* \in \operatorname{crit} f$.*

Now, in a sequence of lemmas, we prove that CoCaIn BPG generates a gradient-like descent sequence for minimizing f_{δ_1}. Moreover, the boundedness of the sequence is guaranteed with the coercivity of the objective, which is typically satisfied in practice. In order to prove condition (C1), we first note that Proposition 10.4.1.2 is also valid for the new Lyapunov function f_{δ_1} as recorded now (for the sake of simplicity we omit the exact details of the proof, which is almost identical to the proof above).

Proposition 5.6.2.1. *Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence generated by CoCaIn BPG. Then, the following assertions hold:*

(i) *The sequence $\left\{f_{\delta_1}\left(x^{k+1}, x^k\right)\right\}_{k\in\mathbb{N}}$ is nonincreasing, converging and condition (C1) of Definition 5.6.2.1 holds true.*

(ii) $\sum_{k=1}^{\infty} D_h(x^{k-1}, x^k) < \infty$*, and hence the sequence* $\left\{D_h(x^{k-1}, x^k)\right\}_{k\in\mathbb{N}}$ *converges to zero.*

(iii) $\min_{1\leq k\leq n} D_h(x^{k-1},x^k) \leq \left(f_{\delta_1}\left(x^1,x^0\right)-f_*\right)/\left(\epsilon n\right)$ *where* $f_* = v(\mathcal{P}) > -\infty$ *(by Assumption A(iv)).*

Now we can prove the following result, which means that condition (C2) holds true.

Proposition 5.6.2.2. *Let* $\{x^k\}_{k\in\mathbb{N}}$ *be a bounded sequence generated by CoCaIn BPG. Then, there exist* $w^{k+1} \in \partial f_{\delta_1}\left(x^{k+1},x^k\right)$ *and a positive scalar* ρ_2 *such that*

$$\left\|w^{k+1}\right\| \leq \rho_2\left(\left\|x^k - x^{k-1}\right\| + \left\|x^{k+1}-x^k\right\|\right), \quad \forall\, k \geq K.$$

The proof of Proposition 5.6.2.2 is given in Section B.6 in the appendix.
Now we are left with showing that CoCaIn BPG generates a sequence that satisfies condition (C3).

Proposition 5.6.2.3. *Let* $\{x^k\}_{k\in\mathbb{N}}$ *be a bounded sequence generated by CoCaIn BPG. Let* x^* *be a limit point of a subsequence* $\left\{x^k\right\}_{k\in\mathcal{K}}$*, then* $\limsup_{k\in\mathcal{K}\subset\mathbb{N}} f(x^k) \leq f\left(x^*\right)$.

The proof of Proposition 5.6.2.3 is given in Section B.7 in the appendix.

The global convergence of CoCaIn BPG now easily follows from our general result on gradient-like descent sequences (see Theorem 5.6.2.1)

Theorem 5.6.2.2 (Global convergence of CoCaIn BPG)**.** *Let* $\{x^k\}_{k\in\mathbb{N}}$ *be a bounded sequence generated by CoCaIn BPG. If* f_0 *and* f_1 *satisfy the KL property, then the sequence* $\{x^k\}_{k\in\mathbb{N}}$ *has finite length, i.e.,* $\sum_{k=1}^{\infty}\left\|x^{k+1}-x^k\right\| < \infty$ *and it converges to* $x^* \in \operatorname{crit} f$.

Before we conclude this section, we provide a simplified variant of CoCaIn BPG.

5.6.3 CoCaIn BPG without backtracking

Note that CoCaIn BPG uses a local estimate of the minorant and majorant parameters $\underline{L}_k$ and $\bar{L}_k$, $k\in\mathbb{N}$, determined by the backtracking steps (5.4.4) and (5.4.6), respectively. However, when the global parameter L is known (guaranteed in Assumption D(ii)), we can skip the backtracking steps, and provide a simplified variant of CoCaIn BPG.

Algorithm 6: CoCaIn BPG without backtracking

- **Input.** $\delta,\epsilon > 0$ with $1 > \delta > \epsilon$.
- **Initialization.** $x^0 = x^1 \in \operatorname{int}\operatorname{dom} h \cap \operatorname{dom} f_0$, $L \geq \max\{\frac{-\alpha}{(1-\delta)\sigma}, L\}$ and $\tau_0 \leq L^{-1}$.
- **General Step.** For $k = 1,2,\ldots,$ compute

$$y^k = x^k + \gamma_k(x^k - x^{k-1}) \in \operatorname{int}\operatorname{dom} h, \tag{5.6.5}$$

$$x^{k+1} \in \operatorname{argmin}_u\left\{f_0\left(u\right) + \left\langle \nabla f_1(y^k), u - y^k\right\rangle + \frac{1}{\tau_k}D_h(u,y^k)\right\}, \tag{5.6.6}$$

where $\tau_k \leq \min\{\tau_{k-1}, L^{-1}\}$ and $\gamma_k \geq 0$ satisfies

$$\left(\delta-\epsilon\right)D_h(x^{k-1},x^k) \geq 2D_h(x^k,y^k)\,. \tag{5.6.7}$$

For the inertial step (5.6.7), when $h = (1/2)\left\|\cdot\right\|^2$ we can obtain that $\gamma_k \leq \sqrt{\frac{\delta-\epsilon}{2}}$ with $\bar{L} = \underline{L}$. Using Remark 5.5.0.1, if $\delta - \epsilon \approx 1$, one could choose the extrapolation parameter as follows $\gamma_k \approx 1/\sqrt{2}$. However, in general, the closed form expression for γ_k is difficult to obtain, for which backtracking line-search strategy can be used. In later chapters (Chapter 6, 7, 8), we develop techniques to obtain closed form inertia for problem specific Bregman distances. We use their technique later in the context of quadratic inverse problems to propose a new variant of CoCaIn BPG with closed form inertia.

5.6.4 Implementing the double backtracking procedure

The update steps of CoCaIn BPG are based on the double backtracking strategy (see steps (5.4.4) and (5.4.6)). Here, we describe some implementation details of these two steps. Note that the inner loops for finding the minorant and the majorant parameters $\underline{L}_k$ and $\bar{L}_k$, $k \in \mathbb{N}$, are implemented in a sequential fashion. By this, we mean that at iteration $k \in \mathbb{N}$ we first execute the steps (5.4.2), (5.4.3) and (5.4.4) in order to compute an appropriate y^k, only then we proceed to steps (5.4.5) and (5.4.6) in order to compute x^{k+1}. Note that the fact that the sequence $\left\{\bar{L}_k\right\}_{k\in\mathbb{N}}$ does not decrease is crucial in order to decouple the steps (5.4.2) and (5.4.5). More precisely, we now describe the backtracking procedure to find $\underline{L}_k$. Let $\underline{\nu} > 1$ be a scaling parameter and arbitrarily initialize $\underline{L}_{k,0} > 0$. Then, we find the smallest $\underline{L}_k \in \left\{\underline{\nu}^0\underline{L}_{k,0}, \underline{\nu}^1\underline{L}_{k,0}, \underline{\nu}^2\underline{L}_{k,0}, \ldots\right\}$ that satisfies (5.4.4) and such that $\gamma_k \geq 0$ satisfies

$$D_h(x^k, y^k) \leq \frac{\delta - \epsilon}{\underline{L}_k\tau_{k-1} + 1} D_h(x^{k-1}, x^k).$$

We can now describe the procedure to find $\bar{L}_k$. Let $\bar{\nu} > 1$ and initialize $\bar{L}_{k,0} := \bar{L}_{k-1}$, then we take the smallest $\bar{L}_k \in \left\{\bar{\nu}^0\bar{L}_{k,0}, \bar{\nu}^1\bar{L}_{k,0}, \bar{\nu}^2\bar{L}_{k,0}, \ldots\right\}$ that satisfies (5.4.6). Therefore, $\left\{\bar{L}_k\right\}_{k\in\mathbb{N}}$ is monotonically non-decreasing. Note, however, we do not require any monotonicity of the sequence $\{\underline{L}_k\}_{k\in\mathbb{N}}$.

The double backtracking strategy preserves the sign of $\underline{L}_k$, however, only $-\underline{L}_k \leq \bar{L}_k$ is required. Changing the sign of $\underline{L}_k$ when the function is locally strongly convex might lead to additional acceleration. However, we leave this kind of adaptation for future work.

5.7 Numerical experiments

Our goal in this section is to illustrate the performance of CoCaIn BPG in various situations. We start with minimization of univariate functions, which emphasizes the power of incorporating inertial terms into the BPG algorithm and using the double backtracking procedure. Then we provide some insights on the following practical applications: quadratic inverse problems in phase retrieval and non-convex robust denoising with non-convex total variation regularization. The efficiency of CoCaIn BPG will be demonstrated for matrix factorization in Chapter 7, for deep linear neural networks in Chapter 7. For the experiments, we use Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz machine with 8 CPUs with x86_64 architecture. We use Python programming language along with popular numerical computing libraries such as Numpy[3] and Scipy[4]. For Section 5.7.4 we additionally use Numba, an open source just in time compiler[5].

5.7.1 Finding global minima of univariate functions

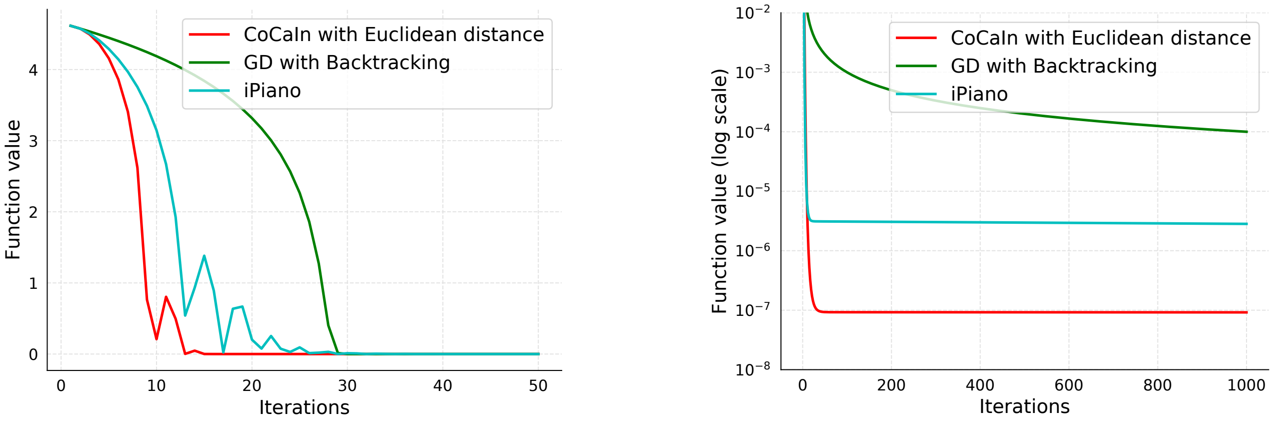
We begin with two examples of minimizing univariate non-convex functions, which shed some light on the two main features of our algorithm: (i) inertial term and (ii) double backtracking procedure. We consider

[3] https://numpy.org/

[4] https://www.scipy.org/

[5] http://numba.pydata.org/

unconstrained minimization of functions $f_1 : \mathbb{R} \to \mathbb{R}$, with Lipschitz continuous gradient, i.e., model $(\mathcal{P})$ with $d = 1$, $f_0 \equiv 0$ and $C = \mathbb{R}$. The two functions are: $f_1(x) = \log\left(1 + x^2\right)$ and $f_1(x) = (1 + e^x)^{-1}$. We compare three methods: CoCaIn BPG with $h = (1/2)\left\|\cdot\right\|^2$ and refer to it as CoCaIn with Euclidean distance, classical Gradient descent (GD) method with backtracking (which is actually CoCaIn with Euclidean distance and with $\gamma_k = 0$ for all $k \in \mathbb{N}$), and iPiano[6] [137] (with the inertial parameter set to 0.7). When using a backtracking procedure in GD and iPiano methods, we mean that only the majorant parameter is varied. We use the same initialization for all the algorithms and report the performance in Figure 5.2.



(A) $f_1(x) = \log\left(1 + x^2\right)$

(B) $f_1(x) = \frac{1}{1+e^x}$

FIGURE 5.2: Better performance by CoCaIn. In the left-hand side plot, the function has a unique critical point. CoCaIn BPG finds it faster than the other two methods. In the right-hand side plot, the function has a very small gradient and CoCaIn BPG reaches a significantly lower function value than the two other methods. These plots hint that CoCaIn BPG can significantly accelerate the convergence speed with comparison to GD and iPiano which use only a simple backtracking procedure.

In the second experiment, we illustrate the robustness of CoCaIn BPG to local minima and critical points. We consider the non-smooth and non-convex function $f(x) = |x| + \sin(x) + \cos(x)$, with many critical points as shown in the center plot of Figure 5.3, and set $f_0(x) = |x|$ and $f_1(x) = \sin(x) + \cos(x)$ (which is obviously a non-convex function with Lipschitz continuous gradient). Here again we take $h = (1/2)\left\|\cdot\right\|^2$. In order to apply CoCaIn BPG, the main computational step is of the following form:

$$x^{k+1} \in \operatorname{argmin}_x \left\{ |x| + \left\langle x - y^k, \cos\left(y^k\right) - \sin\left(y^k\right) \right\rangle + \frac{1}{2\tau_k}\left(x - y^k\right)^2 \right\}, \tag{5.7.1}$$

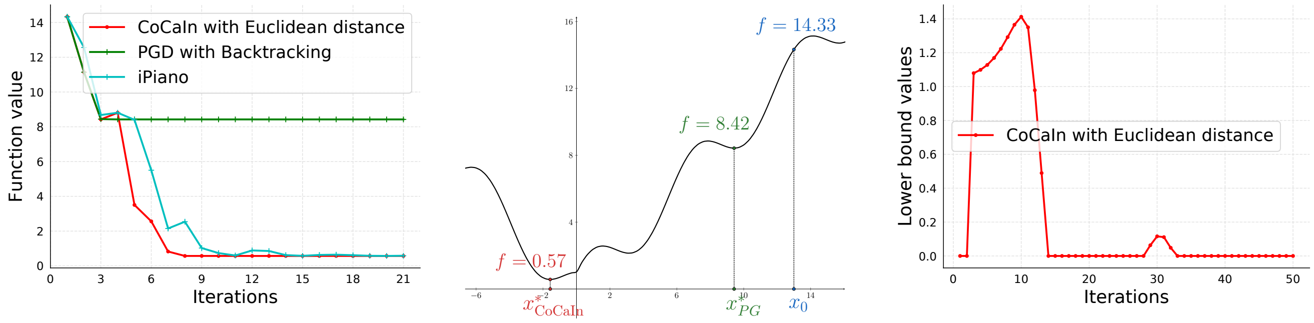
which results in the following update step

$$x^{k+1} = \max\left\{0, \left|y^k - \tau_k \nabla f_1(y^k)\right| - \tau_k\right\} \operatorname{sgn}\left(y^k - \tau_k \nabla f_1(y^k)\right). \tag{5.7.2}$$

We compare CoCaIn BPG with Euclidean distance to the classical Proximal Gradient (PG) method with backtracking (CoCaIn BPG with Euclidean distance and $\gamma_k = 0$, $k \in \mathbb{N}$), and iPiano. As mentioned in the first experiment, when using a backtracking procedure in PG and iPiano methods we mean that only the majorant parameter is varied.

As shown in Figure 5.3, CoCaIn BPG achieves the global minimum, whereas the PG with backtracking gets stuck in a local minimum. We performed the same experiment starting at 100 equidistant points sampled

[6] In this particular case, the method coincides with the Heavy-ball method [145].



(A) Function value plot

(B) $f(x) = |x| + \sin(x) + \cos(x)$

(C) $\underline{L}_k$ value

FIGURE 5.3: CoCaIn can find the global minimum. The left-hand side plot explicitly shows the behavior in terms of function values versus the iterations counter. In the center plot, we use x^*_{PG} as a short hand notation for the critical point achieved by the Proximal Gradient Method with backtracking, and for CoCaIn BPG method we use x^*_{CoCaIn}. The iPiano method achieves the same critical point as the CoCaIn BPG method however it is slower. In the right-hand side plot, we plot $\underline{L}_k$ (the minorant parameter) obtained by CoCaIn BPG method versus the iterations counter. The hilly structures represent that CoCaIn BPG can bypass local maxima and eventually converge to zero. Meaning that CoCaIn BPG adapts to the “local convexity" of the function.

from the interval $[-15, 15]$. The average final function value for CoCaIn was 2.75, whereas for PG method with backtracking it was 3.21 and for the iPiano it was 3.37. This means that CoCaIn BPG reaches the global minimum from 52 points, PG method with backtracking achieves the global minimum only from 27 points and iPiano from 39 points. Hence, the behavior illustrated in Figure 5.3 is not due to the choice of initialization, instead it is due to additional features of the CoCaIn BPG algorithm. This illustrates the great power of using double backtracking procedure in minimizing univariate non-convex functions.

5.7.2 Escaping spurious stationary points

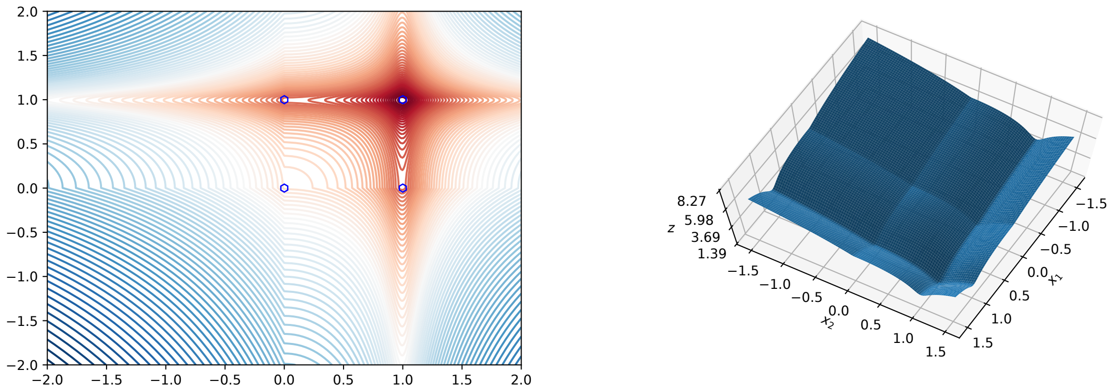
Here, we provide evidence that CoCaIn BPG can escape spurious stationary points in minimizing non-convex functions of two variables. Let $b_i \in \mathbb{R}$, $i = 1, 2, \ldots, m$, be samples of a noisy signal with additive Gaussian noise. A very common task in signal processing is to recover the true data. However, due to the noise, data can be prone to several outliers. In such cases, a robust loss [70] is used. Moreover, prior information about the data, can be embedded through a regularizing term (for instance, a sparsity promoting regularizer). Given $\lambda, \rho > 0$, we consider minimization of

$$f(x) = \lambda \sum_{i=1}^{m} \log\left(1 + \rho\left(x_i - b_i\right)^2\right) + \sum_{i=1}^{m} \log\left(1 + |x_i|\right), \tag{5.7.3}$$

with

$$f_0(x) := \sum_{i=1}^{m} \log\left(1 + |x_i|\right) \quad \text{and} \quad f_1(x) := \lambda \sum_{i=1}^{m} \log\left(1 + \rho\left(x_i - b_i\right)^2\right).$$

The function f_0 is a non-convex sparsity promoting regularizer (also known as the log-sum penalty term [42, 131]) and the function f_1 is a robust loss. For illustration purposes, we consider a simple instance of problem (5.7.3) where $m = 2$, $\lambda = 0.5$ and $\rho = 100$. For minimizing this function we set $C = \mathbb{R}^2$ and $h(x) := (1/2)\left(x_1^2 + x_2^2\right)$ to be used in the CoCaIn BPG method.



(A) Function contour

(B) Function surface

FIGURE 5.4: Function with spurious stationary points. The left-hand side plot shows the contours of the objective function, and the four critical points (denoted with blue diamond). In the right-hand side plot, we show the objective function, where the z-axis represents the function value. Here, the critical points appear as downward kink.

Before presenting the numerical results, we would like to note that in this example, the function $f_0(x) - (\alpha/2)\,h(x)$ is convex for any $\alpha \leq -1$ and $Lh - f_1$ is convex for all $L \geq 100$. Each iteration of CoCaIn BPG would require to compute the Bregman Proximal Gradient mapping, which in this case reduces to the classical Proximal Gradient mapping (due to the choice of h). Note that due to the separability of the functions f_0 and f_1, the needed minimization problem can be split into two individual minimizations with respect to x_1 and x_2. These two optimization problems (after simple manipulations) reduces to computation of the proximal mapping of the univariate function $\tilde{f}(x) := \log(1 + |x|)$. A closed form formula can be found in [76] and reads as follows:

$$\operatorname{prox}_{\tau \tilde{f}(x)}(y) = \begin{cases} \operatorname{sgn}(y)\operatorname{argmin}_{x \in E}\left\{\tilde{f}(x) + \frac{1}{2\tau}(x - |y|)^2\right\}, & \text{if } (|y|-1)^2 - 4(\tau - |y|) \geq 0, \\ 0, & \text{otherwise,} \end{cases}$$

where

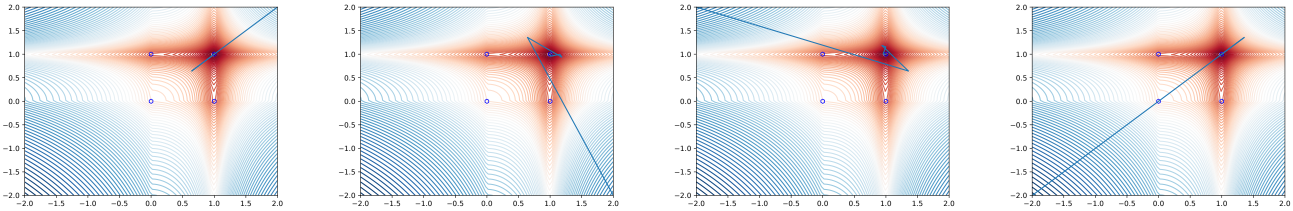
$$E = \left\{0, \left[\frac{|y| - 1 + \sqrt{(|y|-1)^2 - 4(\tau - |y|)}}{2}\right]_+, \left[\frac{|y| - 1 - \sqrt{(|y|-1)^2 - 4(\tau - |y|)}}{2}\right]_+\right\},$$

with $[x]_+ := \max\{0, x\}$.

Now we can apply CoCaIn BPG method and the function behavior is described in Figure 5.4. The performance of CoCaIn BPG is illustrated in Figure 5.5, which shows that CoCaIn BPG can indeed escape spurious critical points to reach the global minimum.

5.7.3 Quadratic inverse problems in phase retrieval

Phase retrieval has been an active research topic for several years [40, 64, 110, 164]. It gained a lot of attention from the optimization community, due to resulting hard non-convex problems [28, 47, 64]. The



(A) From $(2, 2)$ (B) From $(-2, 2)$ (C) From $(2, -2)$ (D) From $(-2, -2)$

FIGURE 5.5: CoCaIn can find the global minimum. The CoCaIn BPG algorithm finds the global minimum at $(1, 1)$, from various initialization points.

phase retrieval problem can be described as follows. Given sampling vectors $a_i \in \mathbb{R}^N$, $i = 1, 2, \ldots, m$, and measurements $b_i > 0$, we seek to find a vector $x \in \mathbb{R}^N$ such that the following system of quadratic equations is approximately satisfied,

$$|\langle a_i, x\rangle|^2 \approx b_i^2, \quad \forall\, i = 1, 2, \ldots, m. \tag{5.7.4}$$

The values in sampling vectors and measurements are taken from a uniform distribution over $[0, 1)$ interval[7]. One typical way to tackle this system is by solving an optimization problem that seeks to minimize a certain error/noise measure in accomodating the equations. The objective function also depends on the type of noise [47] in the system (for instance, Gaussian or Poisson noise). We assume additive Gaussian noise and the squared error measure.

$$f\,(x) = f_0\,(x) + \frac{1}{4}\sum_{i=1}^{m}\left(\langle a_i, x\rangle^2 - b_i^2\right)^2, \quad f_1(x) = \frac{1}{4}\sum_{i=1}^{m}\left(\langle a_i, x\rangle^2 - b_i^2\right)^2. \tag{5.7.5}$$

The function f_0 acts as a regularizing term and is used to incorporate certain prior information on the wished solution. We conduct experiments with two options of regularizing functions: (i) squared ℓ_2-norm, $f_0\,(x) = (\lambda/2)\,\|x\|^2$ and (ii) ℓ_1-norm, $f_0\,(x) = \lambda\,\|x\|_1$. When applying here the CoCaIn BPG method we use the following kernel generating distance function

$$h\,(x) = \frac{1}{4}\,\|x\|_2^4 + \frac{1}{2}\,\|x\|_2^2\,. \tag{5.7.6}$$

We obviously have that $\operatorname{dom} h = \mathbb{R}^N$ and we record below a result [28, Lemma 5.1, p. 2143], which shows that the pair (g, h) satisfies the L-smad property (see Definition 4.4.1.1).

Lemma 5.7.3.1. *Let f_1 and h be as defined above. Then, for any L satisfying*

$$L \geq \sum_{i=1}^{m}\left(3\left\|a_i a_i^T\right\|^2 + \left\|a_i a_i^T\right\|\left|b_i^2\right|\right),$$

the function $Lh - g$ is convex on $\mathbb{R}^N$.

By the design of CoCaIn BPG algorithm, the inertial parameter γ_k must satisfy (5.4.3). However, this involves backtracking over γ_k, which can computationally expensive for high dimensional problems. To this regard, we propose closed form expression for γ_k which satisfies (5.4.3). We also illustrate with our numerical

[7] https://docs.scipy.org/doc/numpy-1.15.1/reference/generated/numpy.random.rand.html

experiments, that CoCaIn BPG variant with closed form inertia is competitive to our main algorithm CoCaIn BPG.

Proposition 5.7.3.1. *Denote $\Delta_k := x^k - x^{k-1}$, for any $k \geq 1$ the following holds*

$$D_h(x^k, y^k) \leq \gamma_k^2 \left\|\Delta\right\|^2 \left(3 \left\|x^k\right\|^2 + \frac{7}{2}\right).$$

The proof of Proposition 5.7.3.1 is given in Section B.9 in the appendix.
Therefore, in this case, Assumptions A, C, D and E are valid. We now discuss the update step of CoCaIn BPG, which requires the solution of the following subproblem

$$x^{k+1} \in \mathrm{argmin}_x \left\{ f_0\left(x\right) + \left\langle \nabla f_1(y^k), x - y^k \right\rangle + \frac{1}{\tau_k} D_h(x, y^k) \right\}. \tag{5.7.7}$$

Following [28], we provide closed form formulas for these optimization problems when f_0 is either the squared ℓ_2-norm or the ℓ_1-norm.

ℓ_1-norm Here we use the following closed form solution, derived in [28, Proposition 5.1, p. 2145]. First, we define the soft-thresholding operator with respect to the parameter $\theta > 0$, as follows

$$\mathcal{S}_\theta\left(y\right) = \mathrm{argmin}_{x \in \mathbb{R}^N} \left\{ \theta \left\|x\right\|_1 + \frac{1}{2}\left\|x - y\right\|^2 \right\} = \max\left\{\left|y\right| - \theta, 0\right\} \mathrm{sgn}\left(y\right), \tag{5.7.8}$$

where all operations are applied coordinate-wise. Then the closed form solution of problem (5.7.7) is given by

$$x^{k+1} = t^* \mathcal{S}_{\lambda\tau_k}\left(\nabla h(y^k) - \tau_k \nabla f_1(y^k)\right),$$

where t^* is the unique positive real root of the following cubic equation

$$t^3 \left\|\mathcal{S}_{\lambda\tau_k}\left(\nabla h(y^k) - \tau_k \nabla f_1(y^k)\right)\right\|_2^2 + t - 1 = 0\,.$$

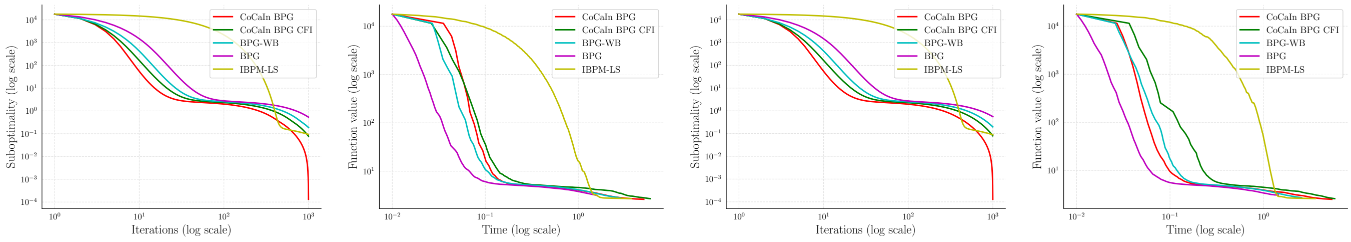
Squared ℓ_2-norm Using similar arguments as of [28, Proposition 5.1, p. 2145], we can easily derive that the solution of problem (5.7.7) is given by

$$x^{k+1} = t^* \left(\tau_k \nabla f_1(y^k) - \nabla h(y^k)\right),$$

where t^* is the unique real root of the following cubic equation

$$t^3 \left\|\tau_k \nabla f_1(y^k) - \nabla h(y^k)\right\|^2 + \left(2\lambda\tau_k + 1\right) t + 1 = 0.$$

We illustrate, in Figure 5.7, the performance of CoCaIn BPG and CoCaIn BPG with closed form inertia (CoCaIn BPG CFI), compared with two other algorithms: (i) the Bregman Proximal Gradient method with backtracking (denoted by BPG-WB) using the same kernel generating distance function (which is exactly CoCaIn BPG with $\gamma_k = 0$ for all $k \in \mathbb{N}$) and (ii) the Inexact Bregman proximal minimization line search algorithm (denoted by IBPM-LS) of [139]. We also compare with the Bregman Proximal Gradient (BPG) method of [28] without backtracking and with the parameter L as derived in Lemma 5.7.3.1.



(A) ℓ_1-norm (B) ℓ_1-norm (C) Squared ℓ_2-norm (D) Squared ℓ_2-norm

FIGURE 5.7: CoCaIn BPG for phase retrieval. The plots illustrate that CoCaIn BPG, CoCaIn BPG CFI and BPG with backtracking performances are competitive to other state of the art optimization algorithms. By suboptimality we mean the difference between the function value and the minimum function value attained by any of the algorithms. The difference is very significant when compared with BPG (without backtracking). This is due to the large L used in the algorithm, thus resulting in smaller steps. On the other hand, CoCaIn BPG uses the local parameters $\underline{L}_k$ and $\bar{L}_k$, thus enjoys larger steps. The function values versus the time plots reveal that CoCaIn BPG rapidly attains a lower function value in a very early stage. Note that CoCaIn BPG and CoCaIn BPG CFI perform very similarly, thus illustrating the benefits of closed form solutions.

5.7.4 Non-convex robust denoising with non-convex TV regularization

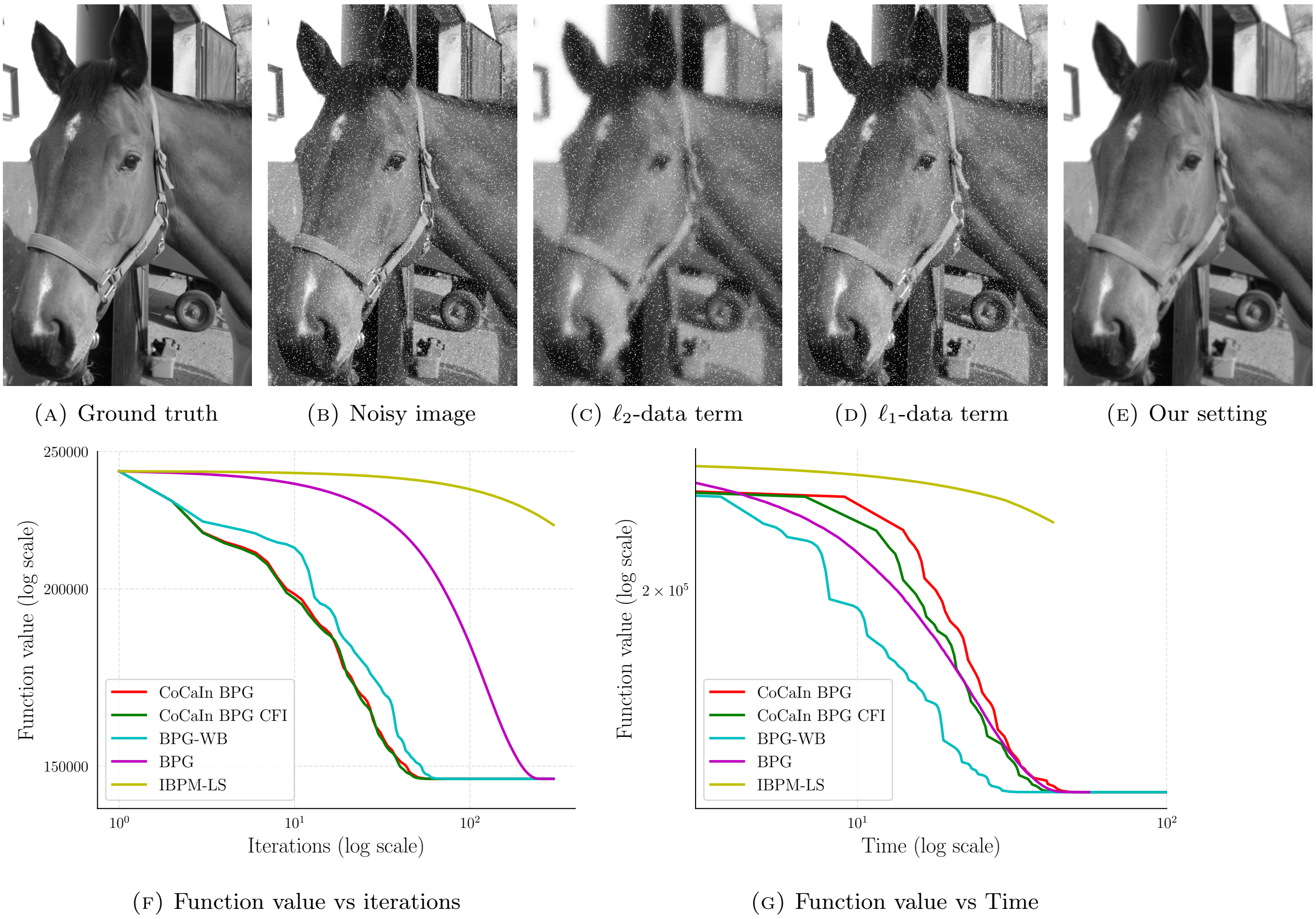
We consider the problem of image denoising of a given image $b \in \mathbb{R}^{M\times N}$, where $M, N \in \mathbb{N}$. The goal is to obtain the true image, denoted by $x \in \mathbb{R}^{M\times N}$. However, in real world applications, it is possible that the measurements are noisy with outliers. The standard routine to deal with outliers is to use robust loss function. The basic idea is to heavily penalize small errors and reasonably penalize large errors. This is done to ensure that the predicted data x, is not influenced significantly by outliers. We consider a fully non-convex formulation of the problem, which includes a non-convex loss function along with a non-convex regularization.

We need the following technical details to provide the full problem statement. The spatial finite difference operator is given by $(\mathcal{D}x)_{i,j} := \left((\mathcal{D}x)^1_{i,j}, (\mathcal{D}x)^2_{i,j}\right)$ where $i \in [M]$ and $j \in [N]$. The horizontal spatial finite differences are given by $(\mathcal{D}x)^1_{i,j} := x_{i+1,j} - x_{i,j}$ for all $i < M$ and 0 otherwise. The vertical spatial finite differences are given by $(\mathcal{D}x)^2_{i,j} := x_{i,j+1} - x_{i,j}$ for all $j < N$ and 0 otherwise. The problem involves the following functions

$$f_0\left(x\right) := \sum_{i=1}^{M}\sum_{j=1}^{N} \log\left(1 + |x_{i,j} - b_{i,j}|\right) , \tag{5.7.9}$$

$$f_1(x) := \lambda \sum_{i=1}^{M}\sum_{j=1}^{N} \log\left(1 + \rho \left\|(\mathcal{D}x)_{i,j}\right\|_2^2\right) , \tag{5.7.10}$$

where $\lambda, \rho > 0$. The function f_0 is non-smooth non-convex and f_1 is smooth non-convex. The function f_1 is a non-convex variant of the popular Total Variation (TV) regularizer, which is used to prefer smooth signals while preserving sharp changes in the signal (such as edges of images). For an overview on non-convex regularizations we refer the reader to [131, 167]. Consider $h\left(x\right) = (1/2)\left\|x\right\|_F^2$. It is easy to prove the convexity of $f_0\left(x\right) - (\alpha/2)\left\|x\right\|_F^2$, by checking that its right derivative is monotonically increasing [86, Theorem 6.4], for all $\alpha \le -1$. The function $Lh - f_1$ is convex for $L \ge 16\lambda\rho$. Due to separability of the function f_0, we can split the computation of the corresponding Bregman Proximal Gradient mapping, into the following separable



(A) Ground truth (B) Noisy image (C) ℓ_2-data term (D) ℓ_1-data term (E) Our setting

(F) Function value vs iterations (G) Function value vs Time

FIGURE 5.8: CoCaIn BPG for robust denoising. We denote ℓ_2-data term for the setting considered with f_0 set to squared ℓ_2-norm based loss and f_1 set to (5.7.10). We denote ℓ_1-data term for the setting with f_0 set to ℓ_1-norm loss and f_1 as in (5.7.10). By our setting, we consider (5.7.9) and (5.7.10). The plots illustrate that BPG methods are competitive for the non-convex robust image denoising problems. IBPM-LS from [139] is barely having any progress, due to flat surfaces. However, BPG methods do not have this issue. The plots illustrate that CoCaIn BPG performance is superior. Also, the reconstructed image obtained by applying CoCaIn BPG to our setting gives a robust reconstruction compared to other reconstructed images.

subproblems

$$x_{i,j}^{k+1} \in \operatorname{argmin}_{x_{i,j} \in \mathbb{R}} \left\{ \log\left(1 + |x_{i,j} - b_{i,j}|\right) + \left\langle x_{i,j} - y_{i,j}^k, \nabla f_1(y^k)_{i,j} \right\rangle + \frac{1}{2\tau_k} \left(x_{i,j} - y_{i,j}^k \right)^2 \right\},$$

which as discussed in Section 5.7.2, can be reduced to the computation of the proximal mapping of the function $\log\left(1 + |x - b|\right)$.

We consider two additional experimental settings apart from our main setting given by (5.7.9) and (5.7.10). Firstly, we use the ℓ_2-norm based data term with the same regularization as in (5.7.10). Secondly, we use the squared ℓ_1-norm based data term with regularization as in (5.7.10). We use the good image given in Figure 5.8a and add severe noise randomly of 10^5 magnitude. We illustrate the robustness of the model given by (5.7.9) and (5.7.10) to such outliers. The reconstructed image from ℓ_2-norm based data penalty term is given in Figure 5.8c and the reconstructed image from ℓ_1-norm based data penalty term is given in Figure 5.8d, after applying CoCaIn BPG. Clearly the ℓ_1-norm based data penalty is better than ℓ_2-norm based data penalty term, which is due to the robustness properties of ℓ_1-norm. However, even using ℓ_1-norm is not enough in the presence of severe outliers, the robustness properties are not so significant. This is mitigated by our setting, where the reconstructed image is given in Figure 5.8e. In our setting, the data term in (5.7.9) is very robust to outliers. In all the settings, we used $\lambda = 10$ and $\rho = 1$. The convergence plots for the experiments with (5.7.9) and (5.7.10) are given in Figure 5.8f and 5.8g. Note that CoCaIn BPG CFI uses the closed form inertia with Euclidean distance. BPG-WB and BPG are same as in earlier experiments. IBPM-LS is a general purpose line-search algorithm for non-convex non-smooth problems proposed in [139]. Even though, IBPM-LS is general, BPG based methods are much faster. The comparisons also illustrate that CoCaIn BPG is better in terms of convergence with respect to iterations and competitive with respect to time. CoCaIn BPG CFI performs very similar to CoCaIn BPG and as anticipated the time plots illustrate that CoCaIn BPG CFI is slightly faster than CoCaIn BPG.

5.8 Chapter conclusion

In this chapter, we proposed an inertial variant of the Bregman Proximal Gradient algorithm, namely, CoCaIn BPG. It relies on double backtracking strategy, which combines both the upper and lower bounds that arises in the L-smad property. In particular, the lower bound governs the inertia and the upper bound governs the step-size. We also proved the global convergence for the sequence generated by CoCaIn BPG. We supplement the theory with several practical applications and some illustrations. An implication of this chapter's work is that once a suitable Bregman distance is developed, the CoCaIn BPG algorithm is readily applicable. We leverage this in the subsequent chapters where we apply CoCaIn BPG algorithm to optimize the objectives that arise in matrix factorization, deep matrix factorization, deep neural networks settings based on Bregman distances developed in Chapter 4.

Chapter 6

Matrix factorization

6.1 Abstract

Matrix factorization is a popular non-convex optimization problem, for which alternating minimization schemes are mostly used. They usually suffer from the major drawback that the solution is biased towards one of the optimization variables. A remedy is non-alternating schemes based on the L-smad property. We already exploited this theory to propose suitable Bregman distances for matrix factorization problems in Chapter 4. In this chapter, we make use of the developed Bregman distances to make BPG and CoCaIn BPG applicable for matrix factorization. The global convergence guarantees are readily valid as a simple consequence. A major challenge in the usage of BPG methods is the efficiency of the update step, for which we develop here closed form solutions which helps improve practical efficiency. In several experiments, we observe a superior performance of our non-alternating schemes (BPG based methods) in terms of speed and objective value at the limit point.

6.2 Introduction

We recall the matrix factorization setting described in Section 4.5. Given a matrix $A \in \mathbb{R}^{M\times N}$, in matrix factorization setting one is interested in the factors $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$ such that $A \approx UZ$ holds. This is usually cast into the following non-convex optimization problem

$$\min_{U\in\mathcal{U}, Z\in\mathcal{Z}} \left\{ f(U,Z) \equiv \frac{1}{2}\left\|A - UZ\right\|_F^2 + \mathcal{R}_1(U) + \mathcal{R}_2(Z) \right\}, \tag{6.2.1}$$

where $\mathcal{R}_1, \mathcal{R}_2$ are regularization terms, $\frac{1}{2}\left\|A - UZ\right\|_F^2$ is the data-fitting term, and $\mathcal{U}, \mathcal{Z}$ are the constraint sets for U and Z respectively. Here, $\mathcal{R}_1(U)$ and $\mathcal{R}_2(Z)$ can be potentially non-convex extended real valued functions and possibly non-smooth. We denote $f_0(U,Z) := \mathcal{R}_1(U) + \mathcal{R}_2(Z)$ and $f_1(U,Z) := \frac{1}{2}\left\|A - UZ\right\|_F^2$. Many practical matrix factorization problems can be cast into the form of (6.2.1). The choice of f_0 and f_1 is dependent on the problem, for which we provide some examples in Section 6.5. Moreover by definition, f_0 is separable in U and Z, which we assume only for practical reasons.

The most frequently used techniques for solving matrix factorization problems involve alternating updates (Gauss–Seidel type methods [75]) like PALM [26], iPALM [144], BCD [171], BC-VMFB [50], HALS [51] and many others. A common disadvantage of these schemes is their bias towards one of the optimization variables. Such alternating schemes involve fixing a subset of variables to do the updates. In order to guarantee convergence to a stationary point, alternating schemes require the first term in (6.2.1) to have a Lipschitz continuous gradient only with respect to each subset of variables. However, in general Lipschitz continuity of the gradient fails to hold for all variables. In order to use non-alternating schemes for (6.2.1), one possible way is to generalize the gradient Lipschitz continuity and for this purpose, we use the L-smad property.

In this regard, we already provided Bregman distances in Chapter 4. Here, we ask the question: "Can we apply BPG and CoCaIn BPG efficiently for matrix factorization problems?". This question is significant, since convergence of the Bregman proximal minimization variants BPG and CoCaIn BPG relies on the L-smad property. A crucial issue is the efficient computability of the algorithm's update steps, which is particularly hard due to the coupling between two subsets of variables. We successfully solve this challenge.

6.2.1 Contributions

In particular, we list our contributions below.

- We make recently introduced powerful Bregman proximal minimization based algorithms BPG [28] and CoCaIn BPG (see Chapter 5) and the corresponding convergence results are applicable to the matrix factorization problems when L2 or L1-regularization is incorporated (see Section 7.3.1).
- We compute the analytic solution for subproblems of the proposed variants of BPG, for which the usual analytic solutions based on Euclidean distances cannot be used.
- Experiments show a significant advantage of BPG and CoCaIn BPG which are non-alternating by construction, compared to popular alternating minimization schemes such as PALM [26] and iPALM [144].

6.2.2 Related work

Alternating minimization is the go-to strategy for matrix factorization problems due to coupling between two subsets of variables [1, 73, 172]. In the context of non-convex and non-smooth optimization, recently PALM [26] was proposed and convergence to stationary point was proved. An inertial variant, iPALM was proposed in [144]. However, such methods require a subset of variables to be fixed. We remove such a restriction here and take the contrary view by proposing non-alternating schemes based on a powerful Bregman proximal minimization framework, where we use BPG and CoCaIn BPG algorithms based on the Bregman distances from Section 4.5.

Recently, the symmetric non-negative matrix factorization problem was solved with a non-alternating Bregman proximal minimization scheme [58] with the following kernel generating distance

$$h(U) = \frac{\|U\|_F^4}{4} + \frac{\|U\|_F^2}{2} .$$

However for the generic matrix factorization problem, such a h is not suitable, unlike our Bregman distance from Section 4.5.

Non-negative matrix factorization (NMF) is a variant of the matrix factorization problem which requires the factors to have non-negative entries [74, 101]. Some applications are hyperspectral unmixing, clustering and others [67, 73]. The non-negativity constraints pose new challenges [101] and only convergence to a stationary point [73, 87] is guaranteed, as NMF is NP-hard in general. Under certain restrictions, NMF can be solved exactly [4, 113] however such methods are computationally infeasible. We give efficient algorithms for NMF and show the superior performance empirically.

Matrix completion is another variant of matrix factorization arising in recommender systems [95] and bio-informatics [107, 163], which is an active research topic due to the hard non-convex optimization problem [41, 68]. The state-of-the-art methods were proposed in [89, 173] and other recent methods include [174]. Here, our algorithms are either faster or competitive.

Our algorithms are also applicable to graph regularized NMF (GNMF) [38], sparse NMF [26], nuclear norm regularized problems [39, 88], symmetric NMF via non-symmetric extension [177].

6.3 Closed form update steps for BPG-MF and CoCaIn BPG-MF

In this section, our focus is to use the Bregman distances proposed for matrix factorization problems in Section 4.5 to make BPG and CoCaIn BPG applicable. Also, we are interested in the transfer of theoretical convergence guarantees. A major challenge in the application of BPG based algorithms is that the update step is not available in closed form. Thus, the rest of the section is focussed on developing closed form update steps for BPG algorithms applied for matrix factorization problems.

We denote the BPG algorithm for matrix factorization as BPG-MF and the CoCaIn BPG algorithm as CoCaIn BPG-MF. BPG algorithm involves the following update step at each $k = 1, 2, \ldots$:

$$\begin{aligned} P^k = \lambda \nabla_U f_1 \left(U^k, Z^k\right) - \nabla_U h(U^k, Z^k) , \quad Q^k = \lambda \nabla_Z f_1 \left(U^k, Z^k\right) - \nabla_Z h(U^k, Z^k) , \\ (U^{k+1}, Z^{k+1}) \in \underset{(U,Z)\in\overline{C}}{\operatorname{argmin}} \left\{ \lambda f_0(U, Z) + \left\langle P^k, U \right\rangle + \left\langle Q^k, Z \right\rangle + h(U, Z) \right\} . \end{aligned} \tag{6.3.1}$$

CoCaIn BPG algorithm involves a similar update step at an extrapolated point and also involves the double backtracking step as described in Chapter 5. To make BPG-MF and CoCaIn BPG-MF an efficient choice for solving matrix factorization, namely closed form expressions for the main update steps (6.3.1) (similarly for CoCaIn BPG-MF) need to be developed.

Note that as BPG-MF and CoCaIn BPG-MF essentially involve the same update step, thus we focus on closed form updates for BPG-MF and the extensions to CoCaIn BPG-MF are straightforward.

For the derivation, we refer to the appendix (see Chapter C), here we just state our results. For the L2-regularized problem

$$f_1(U,Z) = \frac{1}{2}\left\|A - UZ\right\|_F^2, \quad f_0(U,Z) = \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right), \quad h = h_a$$

with $c_1 = 3, c_2 = \|A\|_F$ and $0 < \lambda < 1$ the BPG-MF updates are:

$$\boxed{U^{k+1} = -rP^k\,,\ Z^{k+1} = -rQ^k \text{ with } r \geq 0\,,\ c_1\big(\left\|-P^k\right\|_F^2 + \left\|-Q^k\right\|_F^2\big)r^3 + (c_2 + \lambda_0)r - 1 = 0\,.}$$

The following BPG-MF are equivalent to the above given closed form updates, however, the following updates have better numerical stability.

$$\boxed{\begin{aligned} &U^{k+1} = -r\frac{\sqrt{2}P^k}{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}\,,\ Z^{k+1} = -r\frac{\sqrt{2}Q^k}{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}} \text{ with } r \geq 0\,,\\ &2c_1r^3 + (c_2 + \lambda_0)r - \frac{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}{\sqrt{2}} = 0\,.\end{aligned}}$$

For NMF with additional non-negativity constraints, we replace $-P^k$ and $-Q^k$ by $\Pi_+(-P^k)$ and $\Pi_+(-Q^k)$ respectively where $\Pi_+(.) = \max\{0, .\}$ and max is applied element wise.

Now consider the following L1-regularized problem

$$f_1(U,Z) = \frac{1}{2}\left\|A - UZ\right\|_F^2, \quad f_0(U,Z) = \lambda_1\left(\|U\|_1 + \|Z\|_1\right), \quad h = h_a\,. \tag{6.3.2}$$

The soft-thresholding operator is defined for any $y \in \mathbb{R}^N$ by $\mathcal{S}_\theta(y) = \max\{|y| - \theta, 0\}\operatorname{sgn}(y)$ where $\theta > 0$. Set $c_1 = 3, c_2 = \|A\|_F$ and $0 < \lambda < 1$ the BPG-MF updates with the above given f_1, f_0, h are:

$$\boxed{\begin{aligned} &U^{k+1} = r\mathcal{S}_{\lambda_1\lambda}(-P^k),\ Z^{k+1} = r\mathcal{S}_{\lambda_1\lambda}(-Q^k) \text{ with } r \geq 0 \text{ and}\\ &\qquad c_1\left(\left\|\mathcal{S}_{\lambda_1\lambda}(-P^k)\right\|_F^2 + \left\|\mathcal{S}_{\lambda_1\lambda}(-Q^k)\right\|_F^2\right)r^3 + c_2r - 1 = 0\,.\end{aligned}}$$

The above given closed form updates can also be equivalently stated as following:

$$\boxed{\begin{aligned} &U^{k+1} = r\frac{\sqrt{2}\mathcal{S}_{\lambda_1\lambda}(-P^k)}{\sqrt{\|\mathcal{S}_{\lambda_1\lambda}(-P^k)\|_F^2 + \|\mathcal{S}_{\lambda_1\lambda}(-Q^k)\|_F^2}}\,,\ Z^{k+1} = r\frac{\sqrt{2}\mathcal{S}_{\lambda_1\lambda}(-Q^k)}{\sqrt{\|\mathcal{S}_{\lambda_1\lambda}(-P^k)\|_F^2 + \|\mathcal{S}_{\lambda_1\lambda}(-Q^k)\|_F^2}} \text{ with } r \geq 0\,,\\ &2c_1r^3 + c_2r - \frac{\sqrt{\|\mathcal{S}_{\lambda_1\lambda}(-P^k)\|_F^2 + \|\mathcal{S}_{\lambda_1\lambda}(-Q^k)\|_F^2}}{\sqrt{2}} = 0\,.\end{aligned}}$$

We denote a vector of ones as $e_D \in \mathbb{R}^N$. For additional non-negativity constraints we need to replace $\mathcal{S}_{\lambda_1\lambda}(-P^k)$ with $\Pi_+(-\left(P^k + \lambda_1\lambda e_M e_K^T\right))$ and $\mathcal{S}_{\lambda_1\lambda}\left(-Q^k\right)$ to $\Pi_+(-\left(Q^k + \lambda_1\lambda e_K e_N^T\right))$. Excluding the gradient computation, the computational complexity of our updates is $O(MK + NK)$ only, thanks to linear operations. PALM and iPALM additionally involve calculating Lipschitz constants with at most $O(K^2\max\{M,N\}^2)$ computations. Examples like graph regularized NMF (GNMF) [38], sparse NMF [26], matrix completion [95], nuclear norm regularization [39, 88], symmetric NMF [177] and proofs are given in the appendix.

Note that when $f_0 := 0$, we cannot conclude the global convergence result of BPG and CoCaIn BPG as f_1 is not coercive. However, under the L1-regularization and L2-regularization settings the objective is coercive, thus the global convergence results of BPG algorithms are applicable.

6.4 Discussion

We briefly remark some properties of the update steps of BPG-methods. Note that the updates are independent for U and Z in (6.3.1), where updates can be done in parallel blockwise (communication is only required to solve the 1D cubic equation). This can be potentially used to increase the speedup in practice, in particular for large matrices. Some terms in gradients overlap, so using temporary variables in implementation can possibly increase the speedup.

We now provide insights on why BPG-methods are a better choice over other methods, with focus on alternating methods.

- PALM-methods estimate a Lipschitz constant with respect to a block of coordinates in each iteration, which is expensive for large block matrices. BPG-methods use a global L-smad constant, which is computed only once.
- PALM-methods cannot be parallelized block wise, for example, in the two block case, the computation of the Lipschitz constant of the second block must wait for the first block to be updated, hence it is inherently serial.
- Alternating minimization methods do not converge for non-smooth regularization terms and can be inefficient (for, e.g., ALS) for some matrix factorization problems (see, for example, [94, 147]). BPG-methods and PALM-methods converge (due to linearization).
- PALM is not applicable to the 2D function $f_1(x, y) = (x^3 + y^3)^2$, because the block-wise Lipschitz continuity of the gradients fails to hold even after fixing one variable. BPG-methods are applicable here.
- PALM is not applicable to, for example, symmetric matrix factorization as also pointed in [58] or the following penalty method based (relaxed) orthogonal NMF problem (see (6.2.1))

$$\min_{U\in\mathcal{U},Z\in\mathcal{Z}} \left\{ f \equiv \frac{1}{2}\left\|A - UZ\right\|_F^2 + \frac{\rho}{2}\left\|U^TU - I\right\|_F^2 + I_{U\geq 0} + I_{Z\geq 0} + \mathcal{R}_1(U) + \mathcal{R}_2(Z) \right\},$$

  where second term does not have a block-wise Lipschitz continuous gradient for any $\rho > 0$. Here BPG-methods are applicable (similarly also for Projective NMF) with minor changes to the Bregman distance. For symmetric matrix factorization, we recover the kernel generating distances proposed in [58].
- BPG-methods are very general so the choice of applications will increase substantially and this will potentially open doors to design new losses and regularizers, without restricting to Lipschitz continuous gradients.

State of the art models. The state-of-the-art matrix factorization models in [89] go beyond two factors and new factorization models are introduced. BPG algorithms are not valid in their setting, and requires potentially developing new Bregman distances. Also, BPG based methods are not applicable for big data setting, where stochasticity plays a major role. The stochastic version of BPG was recently proposed in [55]. The empirical comparisons to [89] is still open. Moreover, designing the appropriate kernels in the context of new factorization models can possibly require substantially technical proofs.

Extensions. Our algorithms can potentially extended to several applications, for example, multi-task learning, general matrix sensing, weighted PCA with various applications including cluster analysis, phase retrieval, power system state estimation. Even though CoCaIn BPG-MF appears to perform best, the performance of BPG-MF which forms the basis for CoCaIn BPG-MF, is worst as illustrated in 6.5. This possibly implies that the kernel choice or the coefficients involved in the kernels are not optimal. Such optimal choice of kernel generating distances were partially explored in the context of symmetric matrix factorization setting in [58], where new Bregman distances based on Gram kernels were introduced with state of the art performance in applicable settings.

6.5 Experiments

In this section, we show experiments for the optimization problem in (6.2.1). Denote the regularization settings, **R1:** with $\mathcal{R}_1 \equiv \mathcal{R}_2 \equiv 0$, **R2:** with L2 regularization $\mathcal{R}_1(U) = \frac{\lambda_0}{2}\left\|U\right\|_F^2$ and $\mathcal{R}_2(Z) = \frac{\lambda_0}{2}\left\|Z\right\|_F^2$ for some $\lambda_0 > 0$, **R3:** with L1 Regularization $\mathcal{R}_1(U) = \lambda_0\left\|U\right\|_1$ and $\mathcal{R}_2(Z) = \lambda_0\left\|Z\right\|_1$ for some $\lambda_0 > 0$.

Algorithms. We compare our first order optimization algorithms, BPG-MF and CoCaIn BPG-MF, and recent state-of-the-art optimization methods iPALM [144] and PALM [26]. We focus on algorithms that guarantee convergence to a stationary point. We also use BPG-MF-WB, where WB stands for "with backtracking", which is equivalent to CoCaIn BPG-MF with $\gamma_k \equiv 0$. We use two settings for iPALM, where all the extrapolation parameters are set to a single value β set to 0.2 and 0.4. PALM is equivalent to iPALM if $\beta = 0$. We use the same initialization for all methods.

Simple matrix factorization. We set $\mathcal{U} = \mathbb{R}^{M\times K}$ and $\mathcal{Z} = \mathbb{R}^{K\times N}$. We use a randomly generated synthetic data matrix with $A \in \mathbb{R}^{200\times 200}$ and report performance in terms of function value for three regularization settings, **R1**, **R2** and **R3** with $K = 5$. Note that this enforces a factorization into at most rank 5 matrizes U and Z, which yields an additional implicit regularization. For **R2** and **R3** we use $\lambda_0 = 0.1$. CoCaIn BPG-MF is superior[1] as shown in Figure 6.1 .

Statistical evaluation. We also provide the statistical evaluation of all the algorithms in Figure 6.2, for the above problem. The optimization variables are sampled from [0,0.1] and 50 random seeds are considered. CoCaIn BPG outperforms other methods, however PALM methods are also very competitive. In L1 regularization setting, the performance of CoCaIn BPG is the best. In all settings, BPG-MF performance is worst due to a constant step size, which might change in settings where local adapation with backtracking line search is computationally not feasible.

Matrix completion. In recommender systems [95] given a matrix A with entries at few index pairs in set Ω, the goal is to obtain factors U and Z that generalize via following optimization problem

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\left\|P_\Omega\left(A - UZ\right)\right\|_F^2 + \frac{\lambda_0}{2}\left(\left\|U\right\|_F^2 + \left\|Z\right\|_F^2\right)\right\}, \tag{6.5.1}$$

where P_Ω preserves the given matrix entries and sets others to zero. We use 80% data of MovieLens-100K, MovieLens-1M and MovieLens-10M [83] datasets and use other 20% to test. CoCaIn BPG-MF is faster than all methods as given in Figure 6.3. The MovieLens datasets are essentially a matrix $A \in \mathbb{R}^{M\times N}$, where M denotes the number of users and N denotes the number of movies. Only a few non-zero entries are given and the entries denote the ratings which the user has provided for a particular movie. The ratings can take the

[1] Note that in the y-axis label $v(\mathcal{P})$ is the least objective value attained by any of the methods.

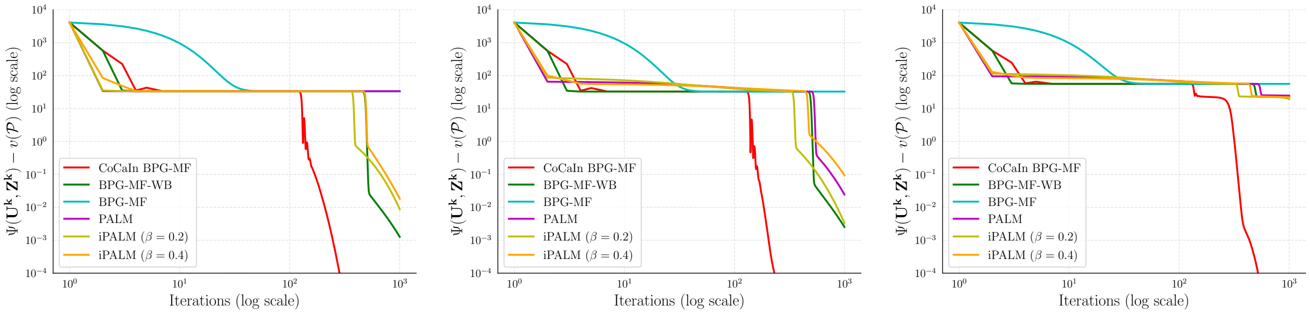
value between 1 and 5, which we refer to as scale. The exact statistics of all the MovieLens datasets are given below.

Dataset	Users	Movies	Non-zero entries	Scale
MovieLens100K	943	1682	100000	1-5
MovieLens1M	6040	3952	1000209	1-5
MovieLens10M	71567	10681	10000054	1-5

The plots provided for the matrix completion problem uses only 80% of the data and we use the remaining 20% as test data in order to obtain the generalization performance to unseen matrix entries with the resulting factors $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$ where we use $K = 5$. The predicted rating to a particular $i \in \{1, 2, \ldots, M\}$ and $j \in \{1, 2, \ldots, N\}$ is given by $(UZ)_{ij}$. The test data is comprised of matrix indices with unseen entries and we denote this set of indices as Ω_T. A popular measure for the test data is the Test RMSE, which is given by the following entity:

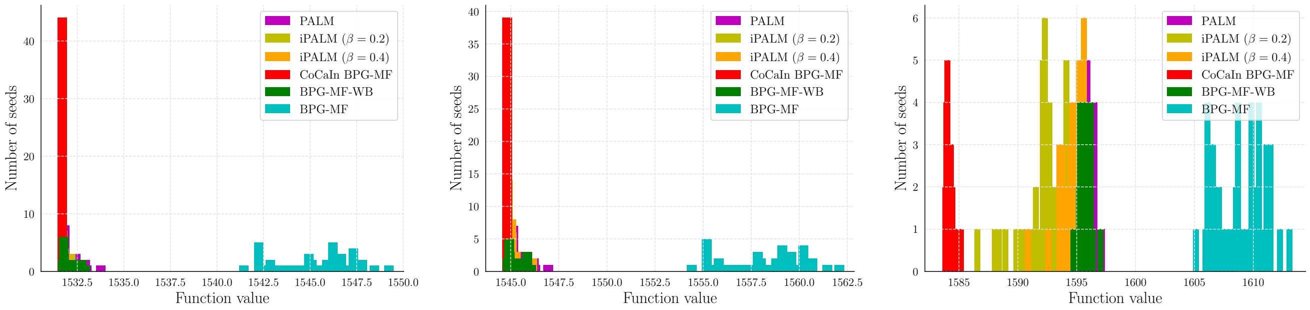
$$\text{Test RMSE} = \sqrt{\frac{1}{|\Omega_T|}\sum_{i=1}^{M}\sum_{j=1}^{N} I_{(i,j)\in\Omega_T}\left(A_{ij} - (UZ)_{ij}\right)^2}\,,$$

where $|\Omega_T|$ denotes the cardinality of the set Ω_T and $I_{(i,j)\in\Omega_T} = 1$ if the index pair (i, j) lies in the set Ω_T else it is zero. The Test RMSE comparisons for the MovieLens Dataset are given below in Figure 6.4.



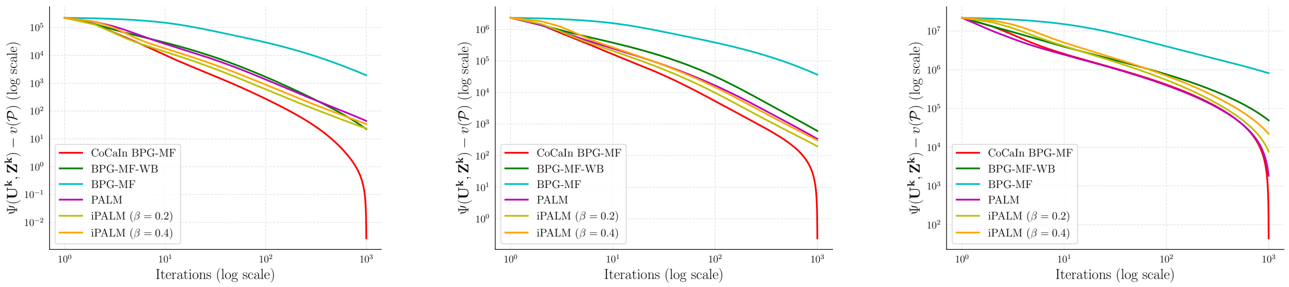
(A) No regularization (B) L2-regularization (C) L1-regularization

FIGURE 6.1: Simple matrix factorization on synthetic dataset.



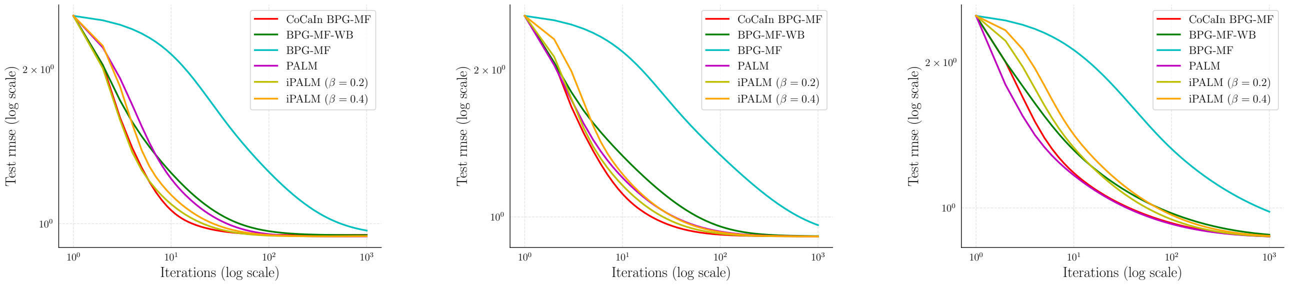
(A) No regularization (B) L2-regularization (C) L1-regularization

FIGURE 6.2: Statistical evaluation on simple matrix factorization.



(A) MovieLens-100K (B) MovieLens-1M (C) MovieLens-10M

Figure 6.3: Matrix completion on Movielens datasets [83].



(A) MovieLens-100K (B) MovieLens-1M (C) MovieLens-10M

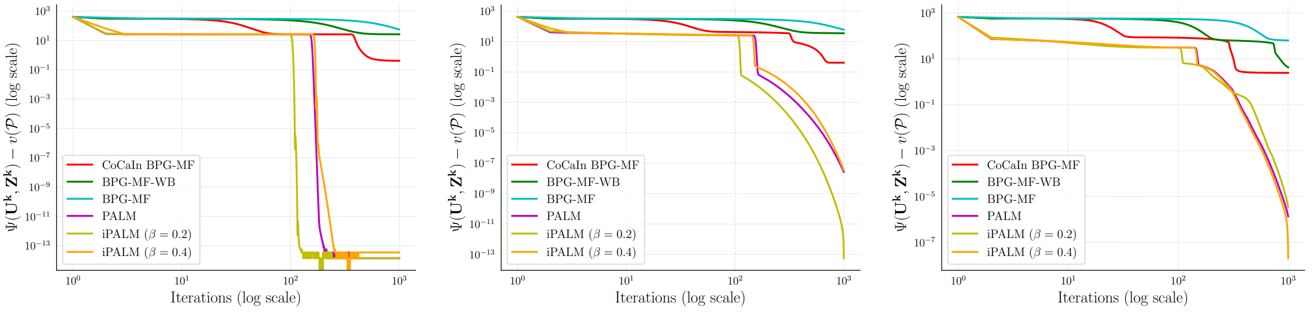
Figure 6.4: Test RMSE plots on MovieLens datasets [83].

As evident from Figures 6.1, 6.5, 6.3, CoCaIn BPG-MF, BPG-MF-WB can result in better performance than well known alternating methods. BPG-MF is not better than PALM and iPALM because of prohibitively small step-sizes (due to $\|A\|_F$ in (4.6.2)), which is resolved by CoCaIn BPG-MF and BPG-MF-WB using backtracking. Time comparisons are also provided, where we show that our methods are competitive. The plots in Figure 6.4 show that the proposed methods BPG-MF-WB and CoCaIn BPG-MF are competitive to PALM and iPALM. BPG-MF is slow in the beginning, however it is competitive to other methods towards the end.

Non-negative matrix factorization. We consider the same setting as the simple matrix factorization problem considered in 6.5, however we set $\mathcal{U} = \mathbb{R}_+^{M\times K}$ and $\mathcal{Z} = \mathbb{R}_+^{K\times N}$. We consider Medulloblastoma dataset [35] dataset with matrix $A \in \mathbb{R}^{5893\times 34}$. As evident from Figure 6.5, PALM based methods outpeform BPG methods here. This raises new open questions and hints at potential variants of BPG which are better suited for constrained problems.

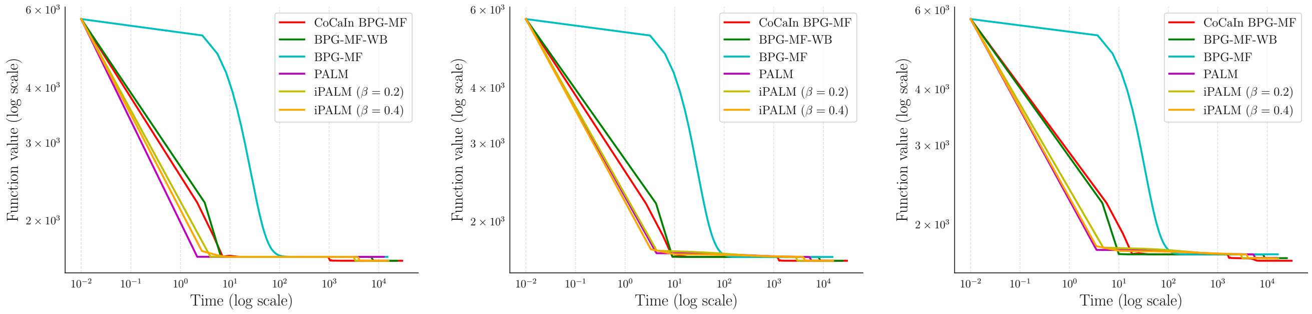
Time comparisons. We provide time comparisons in Figures 6.6, 6.7, 6.8 for all the experimental settings mentioned in Section 6.5, where we mention the dataset in the caption. Since, we used logarithmic scaling, we used an offset of 10^{-2} for all algorithms for better visualization.

As evident from the plots, the proposed variants BPG-MF-WB and CoCaIn BPG-MF are competitive that PALM and iPALM. BPG-MF is mostly slow, due to constant step-size, which can be potentially helpful when backtracking is computationally expensive.



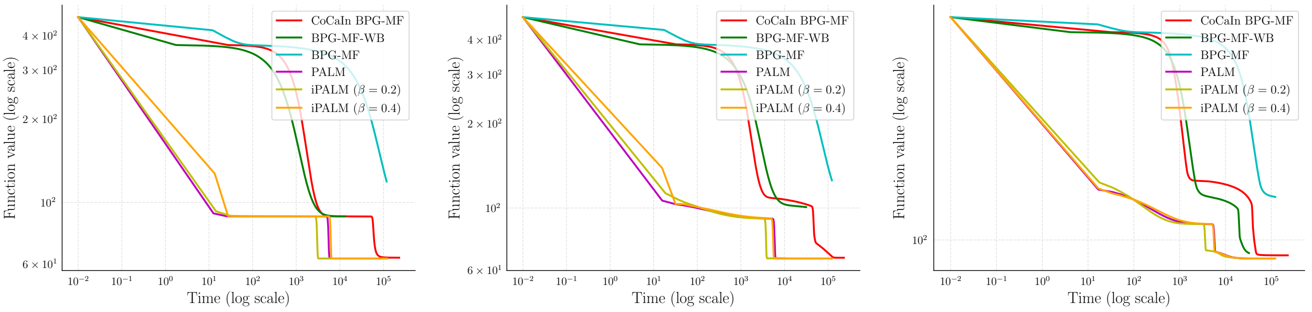
(A) No-regularization (B) L2-regularization (C) L1-regularization

FIGURE 6.5: Non-negative matrix factorization on Medulloblastoma dataset [35].



(A) No regularization (B) L2-regularization (C) L1-regularization

FIGURE 6.6: Time plots for simple matrix factorization on synthetic dataset.

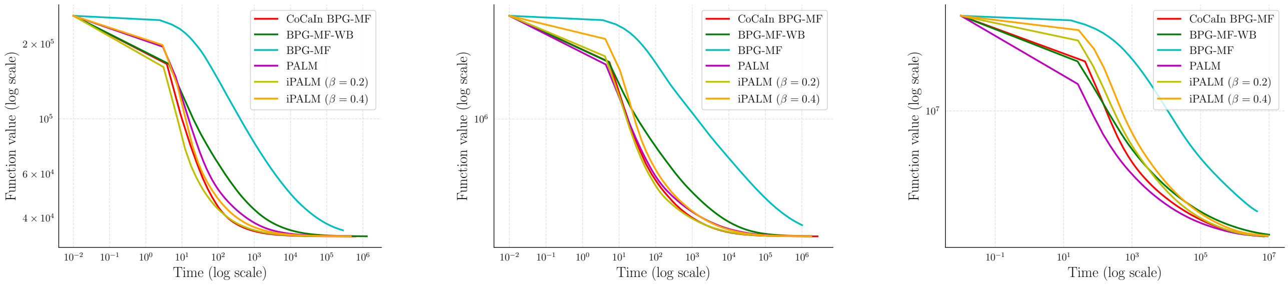


(A) No-Regularization (B) L2-regularization (C) L1-regularization

FIGURE 6.7: Time plots for non-negative matrix factorization on Medulloblastoma dataset [35].

6.6 Chapter conclusion

We proposed non-alternating algorithms to solve matrix factorization problems, contrary to the typical alternating strategies. We achieve this using the Bregman proximal algorithms, BPG [28] and an inertial variant CoCaIn BPG (Chapter 5) for matrix factorization problems. For this purpose, we use the Bregman distances from Section 4.5, which allow for applicability and also enable the transfer of convergence results when L1 or L2-regularization is used. Moreover, we also provide non-trivial efficient closed form update steps for many matrix factorization problems. This line of thinking raises new open questions, such as extensions to tensor factorization [94], to robust matrix factorization [173], stochastic variants [55, 78, 119, 128] and



(A) MovieLens-100K (B) MovieLens-1M (C) MovieLens-10M

FIGURE 6.8: Time plots for matrix completion on MovieLens datasets [83].

state-of-the-art matrix factorization model [89]. We consider similar extensions to deep matrix factorization and deep non-linear neural network settings in the subsequent chapters.

Chapter 7

Deep matrix factorization

7.1 Abstract

In this chapter, we use the Bregman distances for deep matrix factorization problems from Section 4.6, which yields BPG algorithms with theoretical convergence guarantees. In fact, these are the first non-alternating algorithms for such problems allowing for a constant step size strategy. Moreover, we demonstrate the numerical competitiveness of the proposed methods compared to existing state of the art schemes. The main implications of our results are strong convergence guarantees for BPG algorithms. We also propose strategies for their efficient implementation. For example, closed form updates and a closed form expression for the inertial parameter of CoCaIn BPG. Moreover, the BPG method requires neither diminishing step sizes nor line search, unlike its corresponding Euclidean version.

7.2 Introduction

In this chapter, we revisit the deep matrix factorization problem mentioned in Section 4.6, using the same notation. Recall that the optimization problem involved in the deep matrix factorization problem or equivalently training a so-called deep linear neural network (DLNN) model, is the following:

$$\min_{W_i \in \mathcal{W}_i ,\forall i \in \{1,\ldots,N\}} f_1(W) + f_0(W)\,,$$

where

$$f_1(W) := \frac{1}{2}\left\|W_1 W_2 \cdots W_N X - Y\right\|_F^2\,, \tag{7.2.1}$$

and f_0 is the regularization term.

We verified the L-smad property for f_1 by proposing appropriate Bregman distances in Section 4.6. In this chapter, we make use of such Bregman distances such that BPG and CoCaIn BPG algorithms are applicable. However, the standard implementation of BPG based methods is not efficient, in general. There are certain technical issues that require resolution. In this chapter, we tackle all such issue and provide practical solutions.

7.2.1 Contributions

In particular, our contributions are the following.

- We enable an efficient implementation of the update steps involved in BPG based methods via closed form analytic expressions for various practical settings.
- Finding appropriate inertia (or momentum) for CoCaIn BPG can be computationally expensive. In order to mitigate this issue, we propose a novel variant of CoCaIn BPG, called CoCaIn BPG CFI that improves the efficiency for large scale problems.
- The developed Bregman distances in Chapter 4 yield a base algorithm (BPG) that allows for modifications in analogy to the development of alternating, stochastic or inertial variants of the base Proximal Gradient (PG) method, for which we provide a comprehensive discussion.
- Additionally, we empirically illustrate that BPG based algorithms are usually competitive and are often superior to PG variants, whenever both are applicable.

7.2.2 Related work

In [58] a low-rank semidefinite program is reformulated in terms of a symmetric matrix factorization problem which is solved with BPG. To this end the authors prove that the corresponding objective is L-smad relative to a quartic kernel. In Section 4.5, this idea has been extended to a more general regularized matrix factorization problem, for which Bregman distance is proposed to guarantee the L-smad property of the corresponding objective. However, in the deep matrix factorization setting, such a Bregman distance is not valid. Extending on the matrix factorization setting, we already proposed suitable Bregman distances for the deep matrix factorization problem in Section 4.6. Hence, our main focus is to use those Bregman distances to make BPG and CoCaIn BPG algorithms applicable to solve deep matrix factorization problems. Various related state of the art results for deep matrix factorization models were also explored in [17, 115, 165]. A variant of the deep matrix factorization model can also be used for matrix completion [3, 94], which we explore towards the end of this chapter. As mentioned in Section 4.6, it is well justified to study the deep matrix factorization problems [49, 77, 92, 169, 175] before embarking on the deep non-linear neural network setting. Here, we mainly focus on the efficient application of BPG algorithms on deep matrix factorization problems.

Deep linear neural networks are not popular compared to the deep non-linear neural networks (see Chapter 8), as they capture more complex geometries [77]. Even though deep linear neural networks essentially describe a linear model, Mirror Descent (BPG with $f_0 := 0$) eventually inherits the so-called implicit regularization bias observed for Gradient Descent optimization [3, 72, 80]. The implicit regularization bias is helpful to incorporate the information beyond what is specified in the objective [79, 127]. As a future work, the exact quantification of the implicit regularization of BPG based methods could be explored.

7.3 BPG for deep matrix factorization

7.3.1 Closed form updates for BPG

In practice, in order to make use of kernel generating distances proposed in Proposition 4.6.0.1, 4.6.0.2 with BPG, we require efficient update steps. It is in general difficult to compute the Bregman proximal mapping (T_λ in (5.3.1)) in closed form, even for common f_0. Typically this involves the computation of the convex conjugate function of the problem-dependent h which can be hard to derive. In our case we show in Proposition 7.3.1.1, that the computation of the BPG map (5.3.1) can be reduced to a simple projection problem and a simple one-dimensional nonlinear equation, more precisely a polynomial equation with a unique real root. Such a closed form solution is also valid for any other Bregman proximal algorithm including, stochastic BPG [55]. Using f_1 from (7.2.1) and $f_0 := 0$ and we set h as in Proposition 4.6.0.1, 4.6.0.2.

Proposition 7.3.1.1. *In BPG, with above defined f_1, f_0, h, denoting $P_i^k := \lambda \nabla_{W_i} f_1\left(W^k\right) - \nabla_{W_i} h(W^k)\,,$ the update steps in each iteration are given by*

$$W_i^{k+1} = -r\,\frac{\sqrt{N}\,P_i^k}{\|P\|_F}\,, \qquad \textit{for all } i \in \{1,\ldots,N\},$$

with $\|P\|_F^2 = \sum_{i=1}^N \left\|P_i^k\right\|_F^2$. For $N = 2$, $r \geq 0$ satisfies

$$2c_1(2)r^3 + c_2(2)r - \frac{\|P\|_F}{\sqrt{2}} = 0\,,$$

if $N > 2$ and even, $r \geq 0$ satisfies

$$2c_1(N)r^{2N-1} + c_2(N)r^{N-1} + \frac{2\rho}{N}r - \frac{\|P\|_F}{\sqrt{N}} = 0\,,$$

and, if $N > 2$ and odd, $r \geq 0$ satisfies

$$2c_1(N)r^{2N-1} + c_3(N)\left(\frac{Nr^2+1}{N+1}\right)^{\frac{N-1}{2}} r + \frac{2\rho}{N}r - \frac{\|P\|_F}{\sqrt{N}} = 0\,. \tag{7.3.1}$$

The proof is given in Section D.2.1 in the appendix. With a slight abuse of denotation, we are referring that Proposition 7.3.1.1 provides closed form solution, even though r must be found by solving one-dimensional nonlinear equation. We now consider non-zero f_0.

L2-regularization. The squared L2-regularizer is given by

$$f_0(W) := \frac{\lambda_0}{2}\sum_{i=1}^N \|W_i\|_F^2\,, \qquad \text{with } \lambda_0 > 0\,. \tag{7.3.2}$$

To obtain closed forms replace $\frac{2\rho}{N}$ with $\left(\frac{2\rho}{N} + \lambda\lambda_0\right)$ in Proposition 7.3.1.1.

L1-regularization. The L1-regularizer is given by

$$f_0(W) := \sum_{i=1}^{N} \mu_i \left\| W_i \right\|_1 \,, \tag{7.3.3}$$

with $\mu_i > 0$ for all $i \in \{1, \ldots, N\}$. Using the element wise soft-thresholding operator $\mathcal{S}_\theta(x) = \max\{|x| - \theta, 0\}\mathrm{sgn}(x)$, the closed form updates are obtained by replacing $-P_i^k$ with $\mathcal{S}_{\lambda\mu_i}(-P_i^k)$ in Proposition 7.3.1.1. Proof is given in Section D.2.3 in the appendix.

7.3.2 Global convergence of BPG for regularized DLNN

We prove the global convergence of BPG applied to minimize $f := f_1 + f_0$ by invoking [28, Theorem 4.1], where f_0 being either L2-regularizer or L1-regularizer, and f_1 be as defined in (4.6.1).

Theorem 7.3.2.1 (Global convergence of BPG for regularized DLNN)**.** *Let f_1 be defined as in (4.6.1) with $N > 1$, and f_0 be either L2-regularization as in (7.3.2) or L1-regularization as in (7.3.3). If $N = 2$, choose the kernel generating distance function $h = H_a$ as in (4.6.2). If $N > 2$ and even, then choose h as in (4.6.5), otherwise, if $N > 2$ and odd, then choose h as in (4.6.6). Then, f has the KL-property and f_1 is L-smad w.r.t. h. Moreover, the sequence $\{x^k\}_{k\in\mathbb{N}}$ generated by BPG is bounded, has finite length, and converges to a critical point of f.*

The proof is provided in Section D.1 in the appendix. We remark that our theory does not provide global convergence guarantees for no regularization ($f_0 := 0$).

7.4 CoCaIn BPG for deep matrix factorization

7.4.1 Closed form inertia

Now, we present one of our main contribution for efficiently using CoCaIn BPG. The maximal extrapolation is restricted by (5.4.3), where for a constant $\kappa > 0$, the following holds:

$$D_h(x^k, y^k) \le \kappa D_h(x^{k-1}, x^k)\,. \tag{7.4.1}$$

For large scale problems, checking the condition (7.4.1) in a backtracking loop may be expensive. For this purpose, we propose a crucial closed form solution for the extrapolation parameter, which is efficient to implement in practice. For Euclidean distances, $0 < \gamma_k \le \sqrt{\kappa}$ satisfies (7.4.1). Such a closed form interval is non-trivial to obtain in general. However, the structure of the proposed Bregman distances allows for a closed form inertial parameter, as shown in Proposition 7.4.1.1.

Proposition 7.4.1.1. *Denote $x^k = (W_1^k, \ldots, W_N^k)$. For $\kappa > 0$, $y^k := x^k + \gamma_k(x^k - x^{k-1})$ and $x^k \ne x^{k-1}$, the parameter γ_k given by*

$$0 < \gamma_k \le \sqrt{\frac{\kappa D_h(x^{k-1}, x^k)}{\chi(N)}} \le 1\,,$$

satisfies the condition (7.4.1)*, where* $\chi(N)$ *is given by*

$$\begin{cases} c_1(N)\mathcal{B}_k + c_2(N)\mathcal{C}_k\,, & \textit{for } N = 2\,,\\ c_1(N)\mathcal{B}_k + c_2(N)\mathcal{C}_k + \rho\left\|\Delta_k\right\|^2, & \textit{for even } N > 2\,,\\ c_1(N)\mathcal{B}_k + c_3(N)\mathcal{D}_k + \rho\left\|\Delta_k\right\|^2, & \textit{for odd } N > 2\,, \end{cases}$$

with $\Delta_k = x^k - x^{k-1}$, $\Omega_k = 2\left\|x^k\right\|^2 + 2\left\|\Delta_k\right\|^2$,

$$\mathcal{B}_k = \frac{(2N-1)\left\|\Delta_k\right\|^2(\Omega_k)^{(N-1)}}{N^{N-1}}\,,\quad \mathcal{C}_k = \frac{\left((N-1)\left\|\Delta_k\right\|^2(\Omega_k)^{\frac{N-2}{2}}\right)}{(N^{\frac{N}{2}-1})}\,,\quad \mathcal{D}_k = \frac{N\left\|\Delta_k\right\|^2(\Omega_k+1)^{\frac{N-1}{2}}}{(N+1)^{\frac{N-1}{2}}}\,.$$

The proof of Proposition 7.4.1.1 is given in Section D.3.1. For $N = 2$ (matrix factorization) we provide tighter bounds in Lemma D.3.2.2 in the appendix. We denote the variant of CoCaIn BPG with closed form inertia (Proposition 7.4.1.1) as CoCaIn BPG CFI.

7.4.2 Global convergence of CoCaIn BPG for regularized DLNN

We focus on the specialized global convergence result of CoCaIn BPG for regularized DLNN problems, which relies on the L-smad property for DLNN provided in Proposition 4.6.0.1 and Proposition 4.6.0.2.

Theorem 7.4.2.1 (Global convergence of CoCaIn BPG for regularized DLNN)**.** *Let* f_1 *be defined as in* (4.6.1) *with* $N > 1$*, and* f_0 *be either L2-regularization given in* (7.3.2) *or L1-regularization given in* (7.3.3)*. If* $N = 2$*, choose the kernel generating distance function* $h = H_a$ *as in* (4.6.2)*. If* $N > 2$ *and even, then choose* h *as in* (4.6.5)*, otherwise, if* $N > 2$ *and odd, then choose* h *as in* (4.6.6)*. Then,* f *has the KL-property and* f_1 *is* L*-smad w.r.t.* h*. Moreover, the sequence* $\{x^k\}_{k\in\mathbb{N}}$ *generated by CoCaIn BPG is bounded, has finite length, and converges to a critical point of* f*.*

The proof is identical to Theorem 7.3.2.1, except Theorem 5.6.2.2 from Chapter 5 is used instead of [28, Theorem 4.1]. Note that CoCaIn BPG CFI is a special case of CoCaIn BPG. Thus, Theorem 7.4.2.1 also holds true for CoCaIn BPG CFI.

7.5 Discussion of BPG variants

We discuss the applicability and performance of BPG based algorithms for DLNN compared to several existing optimization schemes.

The base algorithm BPG. The key advantage of BPG for DLNN compared to its Euclidean variant, the Proximal gradient (PG) method, is the guaranteed convergence when a constant step size rule is used. This is enabled by validity of L-smad property (Proposition 4.6.0.1 and 4.6.0.2). On the contrary, PG, which requires a classical L-smoothness can only be used by the following trick. Under a coercivity assumption, all iterates generated by PG lie in a compact set, on which a global Lipschitz constant for the objective's gradient can be found. However, the compact set is usually unknown (and cannot be determined before running the algorithm), and can potentially be extremely large which makes the practical computation of such a global Lipschitz constant difficult or computationally intractable. A good heuristic guess may result in PG being more efficient than BPG. Therefore, BPG and CoCaIn BPG (with $\bar{L} = \underline{L}$ in Algorithm 5) render promising alternatives to PG when line search must be avoided due to a prohibitively expensive function evaluation.

BPG with backtracking. If backtracking line search variants are affordable for solving the given optimization problem, then BPG, CoCaIn BPG and their Euclidean variants PG and iPiano provide the same convergence guarantees. Intuitively, from a global perspective, the adapted upper and lower bounds given by the Bregman distance for BPG should tightly approximate to the objective function than quadratic functions as required for L-smoothness. This situation can change when backtracking line search is used and only locally tight approximations are sought. We cannot claim that any of the two strategies has a clear and consistent advantage. The performance can depend significantly on the starting point and the initialization of the parameters and needs problem dependent exploration.

BPG vs PALM. Proximal Alternating Linearized Minimization (PALM) [26] has a clear bias towards the first block of coordinates, if the update direction points into a narrow valley. This effect may be compensated by its inertial variant iPALM. For DLNN with identical regularizers, this effect cannot be observed due to the symmetry of the objective function with respect to the blocks of coordinates, resulting in an oftentimes favorable performance. We leave the exploration of alternating variants of BPG as future work. Related works include [85, 103, 162].

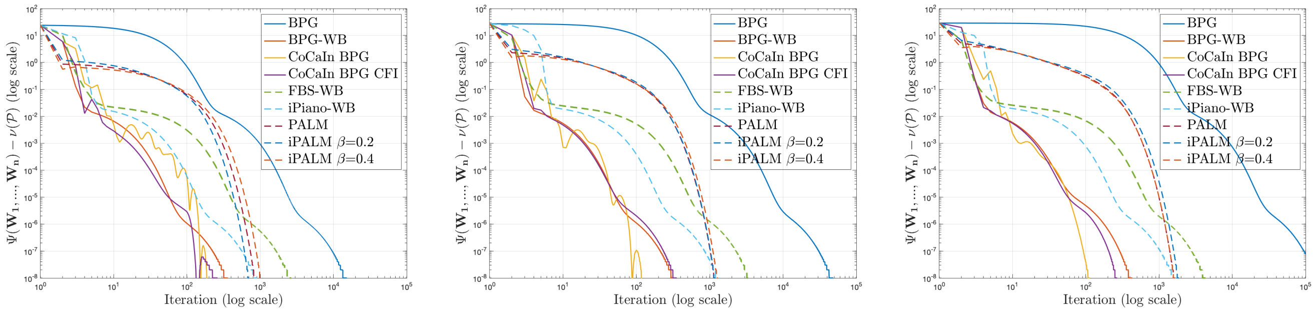
Alternating vs non-alternating strategies. We would like to stress two important advantages of non-alternating schemes such as BPG over alternating minimization strategies like PALM or iPALM. Firstly, BPG allows for block-wise parallelization, and, secondly, there are interesting settings for which alternating minimization is not applicable. The obvious example is symmetric matrix factorization, for which BPG is studied in [58]. In the context of DLNN ($N > 2$ in (4.6.1)) requiring $W_1 = W_2 = \ldots = W_N$ (upto a transpose) can be considered as a prototype for an unrolled recurrent neural network architecture, where weights are shared across layers. Here, there is no natural way to apply alternating minimization schemes and the objective is not classically L-smooth.

Stochastic setting extensions. A stochastic version of BPG was developed recently in [55], for which our Bregman distances are valid to train DLNN. Several popular stochastic variants such as Adam [93], Adagrad [63], SC-Adagrad [119] can potentially be extended with a Bregman proximal framework.

7.6 Experiments

We provide experiments for deep matrix factorization with squared L2-regularizer, L1-regularizer and a non-regularized setting (4.6.1). For regularized objectives, Theorem 7.3.2.1, 7.4.2.1 provide global convergence guarantees for BPG and CoCaIn BPG, respectively. Here, we provide three experiments, two with randomly generated dataset and the other with real world data.

Algorithms. In the experiments, we compare BPG and CoCaIn BPG (Algorithm 5) with many existing optimization methods. We consider alternating strategies such as PALM [26] and iPALM [144]. As non-alternating algorithms, we use Forward–Backward Splitting with backtracking (FBS-WB) and iPiano with backtracking (iPiano-WB) [137]. We also inspect CoCaIn BPG CFI and BPG-WB, which is CoCaIn BPG with $\gamma_k \equiv 0$.



(c) L1-regularization $(N = 3)$ (a) L1-regularization $(N = 4)$ (b) L1-regularization $(N = 5)$

Figure 7.1: Convergence plots illustrate the competitive performance of CoCaIn BPG variants for DLNN.

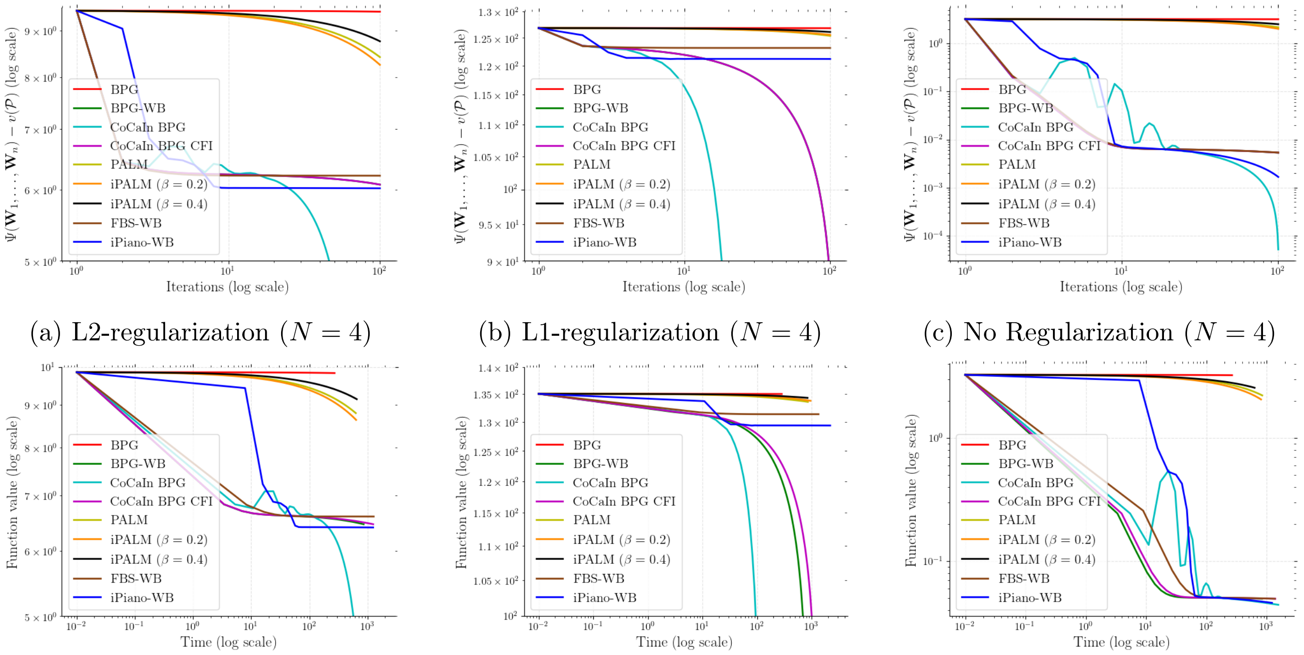
Experiment 1. We set $W_i \in \mathbb{R}^{5\times 5}$, $\forall i = 1, ..., N$, where all weights are initialized with 0.1. Our dataset contains 50 data points with input $X \in \mathbb{R}^{5\times 50}$ and the output $Y \in \mathbb{R}^{5\times 50}$ randomly generated in the interval $[0, 1]$. In this experiment, we work with a network consisting of three, four and five layers $(N = 3, 4, 5)$ and L1 regularization is used. The convergence plots are given in Figure 7.1, where the y-axis measures difference between the absolute objective and the least objective value attained by any of the methods. The additional experiments within the setting of Experiment 1, however, with squared L2 regularization and no regularization are provided in Figure 7.3. In these experiments, we set the regularization parameter $\lambda_0 = 0.1$, the step size λ of BPG to 0.99 and $\rho = 1$. For iPALM we use two settings $\beta = 0.2$ and $\beta = 0.4$. In most of the settings the convergence speed of CoCaIn BPG is similar to iPiano-WB. The alternating schemes PALM and iPALM do not include a time consuming backtracking mechanism. In terms of speed, this results in a better performance for the non-regularized DLNN problem. However, in the regularized setting BPG based methods with a possibly more effective update step remain superior together with iPiano-WB. In this experiment, there is no clear speed advantage of CoCaIn BPG over CoCaIn BPG CFI. The size of the used data is small yet and the strength of the closed form inertial BPG might lie in large scale datasets.

Experiment 2. We consider the matrix completion problem [95] that essentially uses

$$f_1(W) := \frac{1}{2}\left\|P_\Omega(W_1 W_2 \cdots W_N X - Y)\right\|_F^2$$

instead of f_1 in (4.6.1) and P_Ω is a masking operator over a given set of indices Ω, which sets the elements at indices that are not in Ω to zero, while retaining other elements. The changes incurred are the replacement of $\|Y\|_F$ by $\|P_\Omega(Y)\|_F$ in Proposition 4.6.0.1 and 4.6.0.2, and the replacement of the term $(W_1 W_2 \ldots W_N X - Y)$ to $P_\Omega\left(W_1 W_2 \ldots W_N X - Y\right)$ in the gradient expression given in Proposition A.3.0.1 in the appendix. We use MovieLens-100K data [83] with $N = 4$ and X is a scalar with $X = 1$. We use 80% of the data here and later we use 20% of the data to test the performance of the model. The weights $W_1 \in \mathbb{R}^{943\times 5}, W_2 \in \mathbb{R}^{5\times 5}, W_3 \in \mathbb{R}^{5\times 5}, W_4 \in \mathbb{R}^{5\times 1682}$ are initialized with 0.01. The convergence plots are given in Figure 7.2. For Experiment 2, we use $\rho = 0.001$. From the plots in Figure 7.6, it is clear that CoCaIn BPG is not the best in terms of test RMSE, even though it is the best performing algorithm in terms of achieving lower objective value (Figure 7.2). The theoretical justification is still requires further exploration.

Experiment 3. In this experiment we use the same hyperparameters, weight initialization and input $X \in \mathbb{R}^{7\times 50}$ as in Experiment 1. While we used independently generated input and output data in Experiment 1, the output data is now generated with $Y = AX + 0.0001N$, where A is a randomly generated matrix in



(a) L2-regularization ($N = 4$) (b) L1-regularization ($N = 4$) (c) No Regularization ($N = 4$)

(a) L2-regularization ($N = 4$) (b) L1-regularization ($N = 4$) (c) No Regularization ($N = 4$)

FIGURE 7.2: Convergence plots illustrate the competitive performance of CoCaIn BPG for matrix completion task.

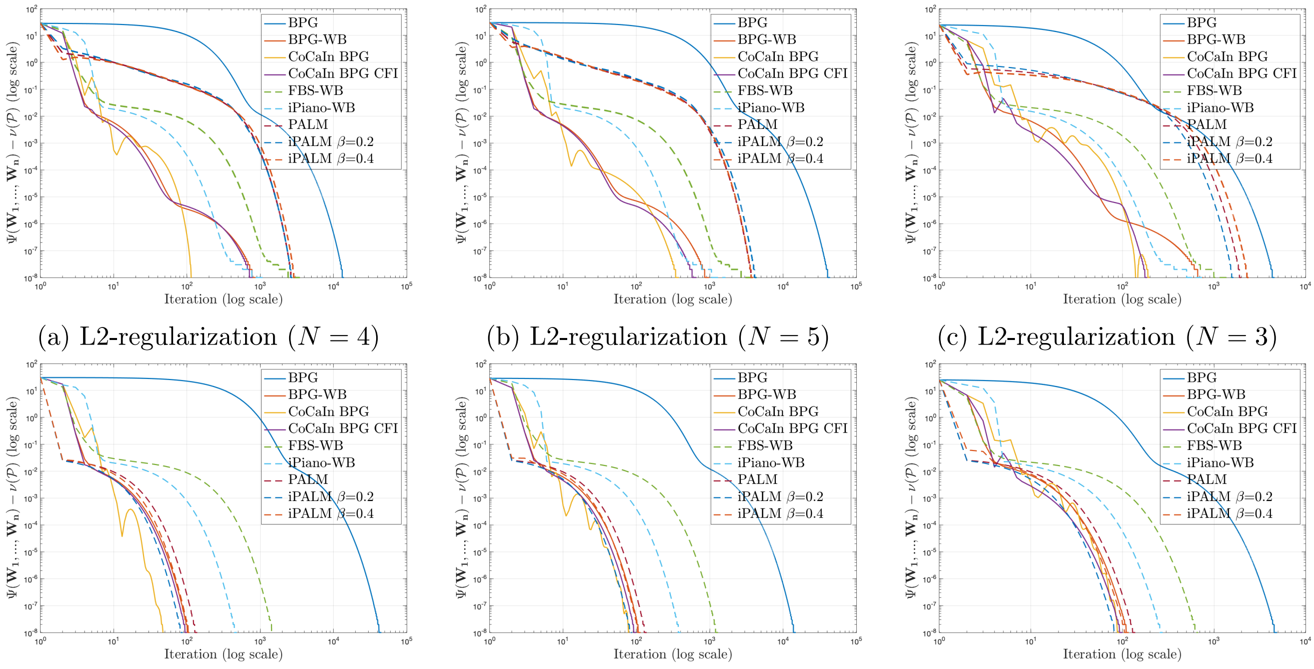
$[0, 0.1]^{2\times7}$ and $N \sim \mathcal{N}(0, 1)$. Additionally, the weights are not squared matrices, i.e $W_1 \in \mathbb{R}^{2\times3}$. The results are provided in Figure 7.5. While BPG-WB and CoCaIn BPG CFI achieve the best performance in a setting with L2-regualrizer or no regularizer, both algorithms can not compete with the alternating algorithms PALM and iPALM as well as iPiano-WB in case of L1-regularizers. Here, CoCaIn BPG is strong with a convergence better than iPiano-WB.

Analysis. The performance of CoCaIn BPG, CoCaIn BPG CFI and BPG-WB is mostly better than other methods. The next competitive algorithms include FBS-WB and iPiano-WB, followed by PALM and iPALM. The performance of the alternating algorithms strongly depend on the usage of a regularizer, whereas BPG-WB is competitive in both settings. At first glance, the performance of BPG appears to be weaker compared to CoCaIn BPG, BPG-WB, FBS-WB, iPiano-WB and other methods. However, line search techniques may not be always desirable in practical scenarios, because line search requires multiple objective evaluations, which can be computationally expensive (see Section 7.5). Moreover, PALM and iPALM need block-wise Lipschitz constant computations in each iteration, which can be expensive. PALM based methods do not perform well on matrix completion task. In the setting of Experiment 2, we provide additional plots in Figure 7.6, where we plot Test RMSE vs iterations and Test RMSE is given by

$$\text{Test RMSE} = \sqrt{\frac{1}{|\Omega_T|}\left\|P_{\Omega_T}(W_1W_2\cdots W_NX - Y)\right\|_F^2}$$

where Ω_T is the index set of test data, which is 20% of the full MovieLens-100K data. The results for time comparison are given in Figure 7.4. For better visualization an offset of 10^{-2} is used in the time plots.

Finally, note that the proposed Bregman distances involve the norms of the weights, which can be very large for large N and might result in numerically instability. An important open research problem, is to develop numerically stable Bregman distances.



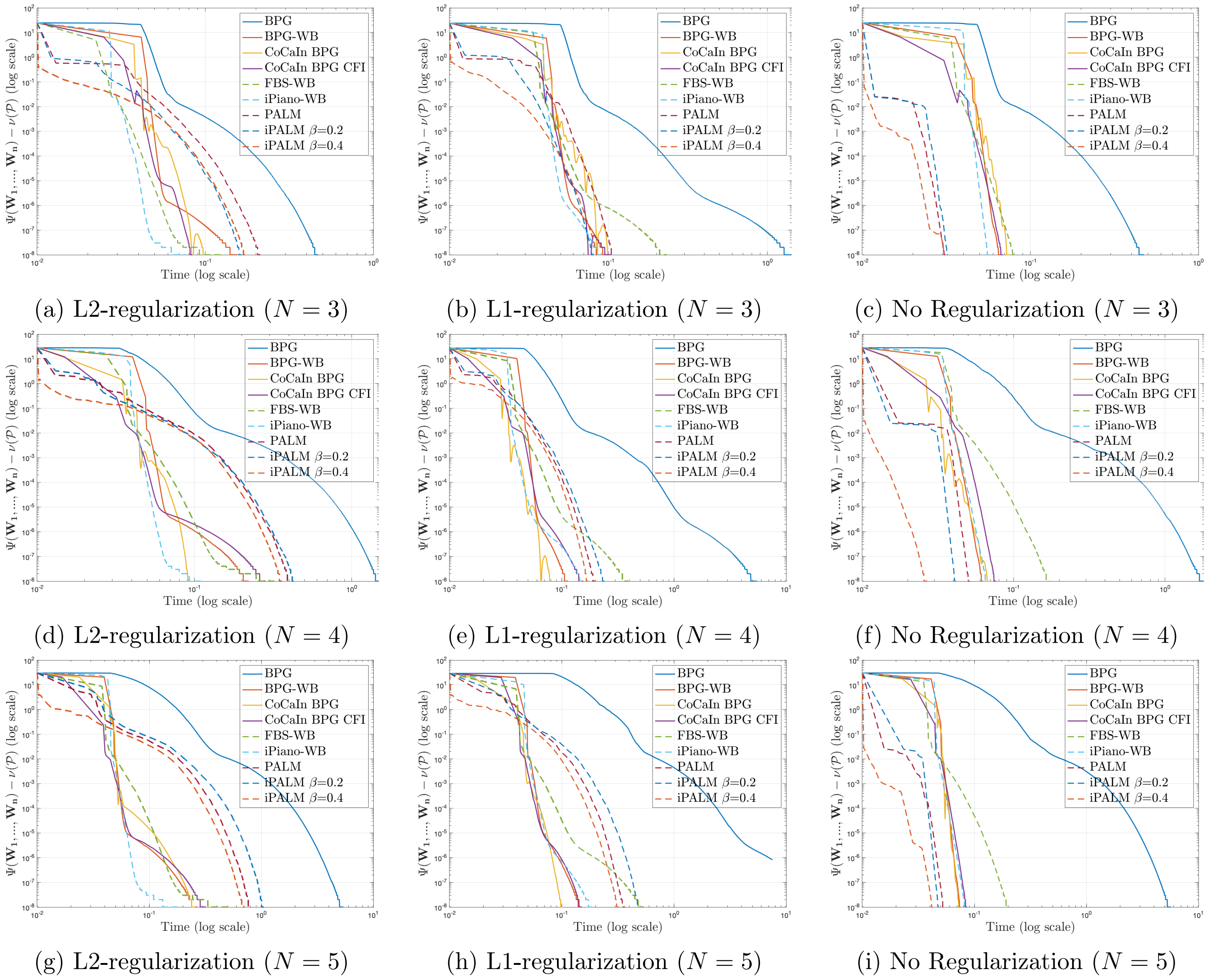
(a) L2-regularization ($N = 4$) (b) L2-regularization ($N = 5$) (c) L2-regularization ($N = 3$)

(d) No Regularization ($N = 5$) (e) No Regularization ($N = 4$) (f) No Regularization ($N = 3$)

FIGURE 7.3: Plots illustrate the competitive performance of CoCaIn BPG variants for DLNN in Experiment 1.

7.7 Chapter conclusion

We considered the optimization problem involved in deep matrix factorization with a quadratic loss, or equivalently, training a deep linear neural network. Our main contribution is to make BPG and its inertial variant CoCaIn BPG applicable and enable the transfer of their convergence results to such problems. We provide various crucial pointers for efficient implementation of BPG based algorithms. In particular, we develop the update formulas, which are crucial for efficient large scale optimization. Also, the validity of inertial (or momentum) parameter in CoCaIn BPG requires to be checked via backtracking line search. To avoid expensive backtracking operation, we derive an analytic expression. This work paves the way for our next chapter, where we embark on the challenging deep non-linear neural network based on similar ideas developed in this chapter.



(a) L2-regularization ($N = 3$)

(b) L1-regularization ($N = 3$)

(c) No Regularization ($N = 3$)

(d) L2-regularization ($N = 4$)

(e) L1-regularization ($N = 4$)

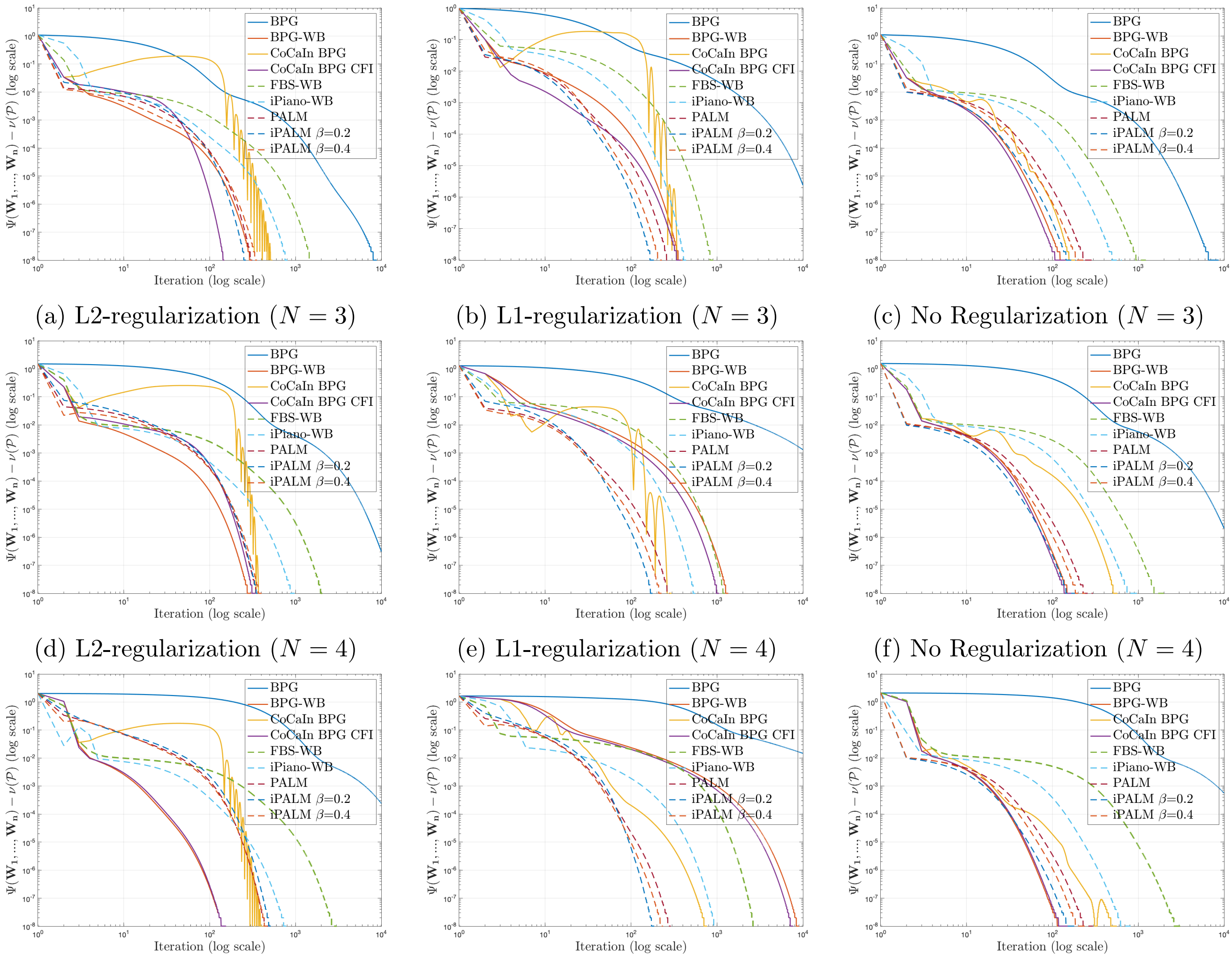
(f) No Regularization ($N = 4$)

(g) L2-regularization ($N = 5$)

(h) L1-regularization ($N = 5$)

(i) No Regularization ($N = 5$)

Figure 7.4: Time plots illustrate the competitive performance of BPG methods, PALM methods in Experiment 1.



(a) L2-regularization ($N = 3$)

(b) L1-regularization ($N = 3$)

(c) No Regularization ($N = 3$)

(d) L2-regularization ($N = 4$)

(e) L1-regularization ($N = 4$)

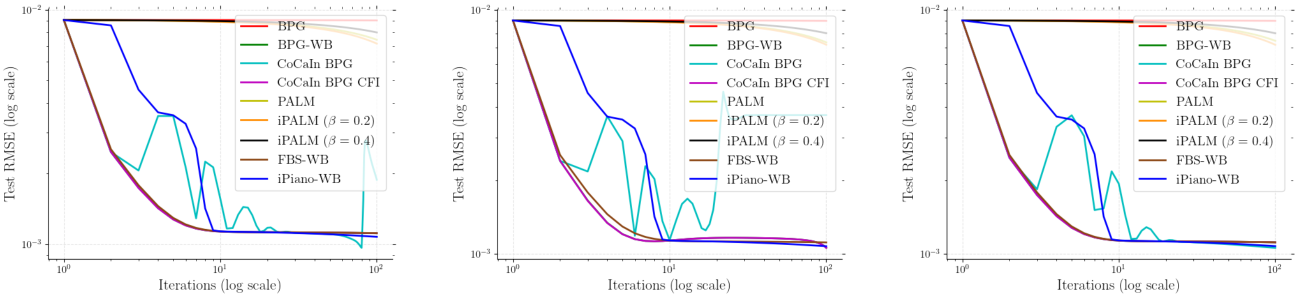
(f) No Regularization ($N = 4$)

(g) L2-regularization ($N = 5$)

(h) L1-regularization ($N = 5$)

(i) No Regularization ($N = 5$)

FIGURE 7.5: Convergence plots for Experiment 3 where BPG based methods and PALM based methods are competitive.



(a) L2-regularization ($N = 4$)

(b) L1-regularization ($N = 4$)

(c) No Regularization ($N = 4$)

FIGURE 7.6: Test RMSE plots for Experiment 2 illustrating the competitive performance of CoCaIn BPG CFI.

Chapter 8

Deep neural networks

8.1 Abstract

In this chapter, we consider the objectives arising in deep non-linear neural settings. In particular, we consider both the regression setting and the classification setting. Based on the Bregman distances proposed in Chapter 4, we make BPG based algorithms applicable. We provide results for both the regression setting and the classification setting, while extending to deep non-linear neural networks. For efficient practical application of BPG methods, we develop closed form update steps for various practical settings. We also develop closed form inertial solutions for efficient implementation of CoCaIn BPG. We provide various empirical evaluations using real world datasets to supplement our claims.

8.2 Introduction

Deep learning is a popular technique to achieve the state of the art performance on many Machine Learning problems that arise in Computer Vision, Natural Language Processing and many other research areas [77, 96, 105, 154]. In the previous chapters, we considered matrix factorization (Chapter 6) and deep matrix factorization (Chapter 7) which essentially falls under the category of deep linear neural network setting. However, in practice deep non-linear neural networks are preferred. We are mainly interested in the regression setting (see Section 4.7) and the classification setting (see Section 4.8), which we recall below. We recall the regression setting from Section 4.7, which involves the following optimization problem:

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in [N]} \left\{ f_0(W) + f_1(W) \right\} , \tag{8.2.1}$$

where

$$f_1(W) := \frac{1}{2} \left\| \sigma_N(W_N \ldots \sigma_1(W_1 X)) - Y \right\|_F^2 ,$$

and f_0 is the regularization term. The classification setting as described in Section 4.8 involves the same problem as (8.2.1), however, f_1 is set to the following:

$$f_1(W) := \sum_{j=1}^{M} \left(-\log \left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^{K} e^{z_{j,k}}} \right) \right) ,$$

where with deep linear neural networks we set $z_j = W_1 \ldots W_N x_j$, and with generic deep non-linear neural network we set $z_j := \sigma_N(W_N \ldots \sigma_1(W_1 x_j))$.

Much of the effort has gone into the understanding of the optimization of objectives that arise in deep neural networks. However, there is no constant step-size algorithm with global convergence guarantees that is suitable for deep non-linear neural networks, as far as we know. We tackle this open problem via the so-called L-smad property, which we introduced in Chapter 4. The L-smad property played a crucial role in the development of BPG-based methods (see [28] and Chapter 5). However, proving such a property is non-trivial and was tackled in Section 4.7, 4.8 for objectives in deep neural network settings. In the same spirit as matrix factorization and deep matrix factorization, there will be certain technical issues that need to be resolved in order to apply BPG-based methods based on the L-smad property. In this chapter, we will successfully tackle the issues. The techniques used are essentially the same as that of deep matrix factorization setting and thus we do not go into detail regarding the proofs. A notable distinction between the deep matrix factorization case and the deep non-linear neural networks case is that in the later case we do not have a distinction between odd and even layers, whereas in the former case we required such a distinction.

8.2.1 Contributions

We use the Bregman distances proposed in Section 4.7, 4.8 for the regression and the classification settings that arise in the context of deep non-linear neural networks to enable the applicability of BPG-based methods, in particular BPG and CoCaIn BPG. To this end, we briefly list our contributions below.

- A major challenge that arises in the application of BPG based methods is the efficient implementation of the update steps. In this regard, we provide closed form solutions to the subproblems that arise in BPG based methods.

- Based on the above-mentioned Bregman distances, in order to enhance the efficiency of the implementation of CoCaIn BPG method, we provide results pertaining to closed form inertia that result in efficient extrapolation steps.

- Finally, we supplement our theory with various empirical comparisons on real world datasets for both the regression and the classification settings. We observe that BPG methods are competitive compared to forward–backward splitting method. However, we found that CoCaIn BPG suffers from severe numerical issues, which we leave it as an open research question.

8.2.2 Related work

Optimization of deep neural networks is a hard research problem [31, 77]. Various state of the art results are achieved via efficient optimization of deep neural networks [96, 105, 154]. Typically, stochastic gradient based algorithms are used in deep neural network training [30, 32]. In the full gradient setting, algorithms such as the Gradient Descent variants (with and without momentum) are applicable [151], and adaptive algorithms like Adam [93], Adagrad [63], SC-Adagrad [119] are applicable. Inspite of their efficiency, they require heavy tuning of the step-size and other hyperparameters while having limited theoretical convergence guarantees. Here, algorithms with constant step-size with global convergence guarantees are relatively unknown. To train deep neural networks, one possible class of algorithms that has global convergence guarantees is the class of alternating optimization methods. As mentioned in Chapter 6, popular alternating methods [75] include PALM [26], iPALM [144], BCD [171], BC-VMFB [50], HALS [51]. However, alternating methods have a bias towards one of the weights, and also the computation of the Lipschitz constant of a block-wise gradient can be expensive. Inorder to mitigate this we leverage the Bregman distances proposed in Section 4.7, 4.8 to make non-alternating Bregman proximal minimization algorithms applicable along with their convergence guarantees.

8.3 Closed form updates

We already proposed suitable Bregman distances for deep non-linear neural networks in Section 4.7, 4.8 for both the regression and the classification settings. These Bregman distances can essentially be seen as a special case of the following kernel generating distance:

$$\sum_{u=1}^{2N} \mathcal{C}_u \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{N} \right)^u , \tag{8.3.1}$$

where the constants $\mathcal{C}_u$ are non-negative for $u \in \{1, \ldots, 2N\}$. The choices for constants $\mathcal{C}_u$ vary according to the setting. The subproblems that arise in the update steps of BPG and CoCaIn BPG are similar. Thus, we focus on the closed form update steps for BPG with the following result.

Proposition 8.3.0.1. *Let f_1 be any of the before-mentioned objectives stated in this chapter and h be its corresponding kernel generating distance that takes the form (8.3.1) with appropriately chosen coefficients, such that (f_1, h) satisfies L-smad property. In BPG, denoting $P_i^k := \lambda \nabla_{W_i} f_1 \left(W^k\right) - \nabla_{W_i} h(W^k)$, the update step in each iteration are given by*

$$W_i^{k+1} = -r \, \frac{\sqrt{N}\, P_i^k}{\|P\|_F} \,, \qquad \textit{for all } i \in \{1, \ldots, N\},$$

with $\|P\|_F^2 = \sum_{i=1}^N \left\|P_i^k\right\|_F^2$. Then, quantity $r \geq 0$ satisfies

$$\sum_{u=1}^{2N} 2\mathcal{C}_u \left(\frac{u}{N}\right) r^{2u-1} - \frac{\sqrt{\sum_{i=1}^N \left\|P_i^k\right\|_F^2}}{\sqrt{N}} = 0. \tag{8.3.2}$$

The proof is provided in Section E.1 in the appendix. In order to obtain an even more general result, one can replace $2N$ with any positive integer greater than one.

8.3.1 Regularization

Recall that we are interested in the following problem:

$$\inf \left\{ f(x) := f_0(x) + f_1(x) : \; x \in \mathbb{R}^N \right\} . \tag{8.3.3}$$

We now consider the closed form update of BPG-based methods when regularization term f_0 is used in conjunction with f_1, where f_1 is any of the before-mentioned objectives in Sections 4.7, 4.8.

L2-regularization. Recall that the squared L2-regularizer is given by

$$f_0(W) := \frac{\lambda_0}{2} \sum_{i=1}^{N} \|W_i\|_F^2 \; , \qquad \text{with } \lambda_0 > 0 \, . \tag{8.3.4}$$

To obtain the closed form solutions replace $\frac{2\mathcal{C}_1}{N}$ with $\left(\frac{2\mathcal{C}_1}{N} + \lambda\lambda_0\right)$ in Proposition 8.3.0.1.

L1-regularization. Recall that the L1-regularizer is given by

$$f_0(W) := \sum_{i=1}^{N} \mu_i \|W_i\|_1 \; , \tag{8.3.5}$$

with $\mu_i > 0$ for all $i \in \{1, \ldots, N\}$. Using the element wise soft-thresholding operator $\mathcal{S}_\theta(x) = \max\{|x| - \theta, 0\}\mathrm{sgn}(x)$, the closed form updates are obtained by replacing $-P_i^k$ with $\mathcal{S}_{\lambda\mu_i}(-P_i^k)$ in Proposition 8.3.0.1.

8.4 Closed form inertia

In CoCaIn BPG, the linear extrapolation parameter γ_k is found such that the following condition holds true:

$$D_h(x^k, y^k) \leq \kappa D_h(x^{k-1}, x^k) \, .$$

In this section, we focus on obtaining closed form solutions for γ_k based on the Bregman distances considered for regression and classification problems arising in deep neural networks.

8.4.1 Closed form inertia - Regression setting

Consider the setting from Section 4.7. As a consequence of Lemma A.3.0.3 we obtain the following result.

Lemma 8.4.1.1. *Let h be as in (4.7.2). Denote for any $k \geq 1$, $x^k = (W_1^k, \ldots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{E}_k := \sum_{u=1}^{2N} \left(\Gamma_u \frac{(2u-1)}{u^{u-1}} \right) \|\Delta_k\|^2 \left(2 \left\| x^k \right\|^2 + 2 \|\Delta_k\|^2 \right)^{(u-1)} .$$

The following upper bound holds true

$$D_h(x^k, y^k) \leq \gamma_k^2 \mathcal{E}_k \, .$$

Furthermore, as a simple consequence of Lemma 8.4.1.1 we obtain the following closed form inertia, which can be used in CoCaIn BPG.

Proposition 8.4.1.1. *Let h be as in (4.7.2). Denote $x^k = (W_1^k, \dots, W_N^k)$. For $\kappa > 0$, $y^k := x^k + \gamma_k(x^k - x^{k-1})$ and $x^k \neq x^{k-1}$, the parameter γ_k given by*

$$0 < \gamma_k \leq \sqrt{\frac{\kappa D_h(x^{k-1}, x^k)}{\chi(N)}} \leq 1\,,$$

satisfies the condition

$$D_h(x^k, y^k) \leq \kappa D_h(x^{k-1}, x^k)\,,$$

where $\chi(N) = \mathcal{E}_k$.

8.4.2 Closed form inertia - DLNN - Classification setting

We continue the setting of Section 4.8.1. In the same spirit as Section 8.4.1, we consider the issue of obtaining a closed form inertial solution for efficient application of CoCaIn BPG. As a consequence of Lemma A.3.0.3 we obtain the following result.

Lemma 8.4.2.1. *Let h be as in (4.8.10). Denote for any $k \geq 1$, $x^k = (W_1^k, \dots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{F}_k := \widehat{\Gamma}_N \left(\frac{(2N-1)}{N^{N-1}}\right) \|\Delta_k\|^2 \left(2\left\|x^k\right\|^2 + 2\|\Delta_k\|^2\right)^{(N-1)} + \frac{\rho}{2}\|\Delta_k\|^2\,.$$

The following upper bound holds true

$$D_h(x^k, y^k) \leq \gamma_k^2 \mathcal{F}_k\,.$$

Furthermore, as a simple consequence of Lemma 8.4.2.1 we obtain the following closed form inertia, which can be used in CoCaIn BPG.

Proposition 8.4.2.1. *Let h be as in (4.8.10). Denote $x^k = (W_1^k, \dots, W_N^k)$. For $\kappa > 0$, $y^k := x^k + \gamma_k(x^k - x^{k-1})$ and $x^k \neq x^{k-1}$, the parameter γ_k given by*

$$0 < \gamma_k \leq \sqrt{\frac{\kappa D_h(x^{k-1}, x^k)}{\chi(N)}} \leq 1\,,$$

satisfies the condition

$$D_h(x^k, y^k) \leq \kappa D_h(x^{k-1}, x^k)\,,$$

where $\chi(N) = \mathcal{F}_k$.

8.4.3 Closed form inertia - DNN - Classification setting

We consider the setting from Section 4.8.2. As a consequence of Lemma A.3.0.3 we obtain the following result.

Lemma 8.4.3.1. *Let h be as in (4.8.12). Denote for any $k \geq 1$, $x^k = (W_1^k, \dots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{G}_k := \sum_{u=1}^{N} \left(\tilde{\Gamma}_u \frac{(2u-1)}{u^{u-1}}\right) \|\Delta_k\|^2 \left(2\left\|x^k\right\|^2 + 2\|\Delta_k\|^2\right)^{(u-1)}\,.$$

The following upper bound holds true

$$D_h(x^k, y^k) \leq \gamma_k^2 \mathcal{G}_k \,.$$

Furthermore, as a simple consequence of Lemma 8.4.3.1 we obtain the following closed form inertia, which can be used in CoCaIn BPG.

Proposition 8.4.3.1. *Let h be as in (4.8.12). Denote $x^k = (W_1^k, \ldots, W_N^k)$. For $\kappa > 0$, $y^k := x^k + \gamma_k(x^k - x^{k-1})$ and $x^k \neq x^{k-1}$, the parameter γ_k given by*

$$0 < \gamma_k \leq \sqrt{\frac{\kappa D_h(x^{k-1}, x^k)}{\chi(N)}} \leq 1 \,,$$

satisfies the condition

$$D_h(x^k, y^k) \leq \kappa D_h(x^{k-1}, x^k) \,,$$

where $\chi(N) = \mathcal{G}_k$.

8.5 Experiments

In this section, we consider the numerical performance of BPG methods on the objectives arising in deep non-linear neural networks. Basically, we consider the optimization of the following problem:

$$\min_{W_i \in \mathcal{W}_i, \forall i \in \{1, \ldots, N\}} f_0(W) + f_1(W) \,, \tag{8.5.1}$$

where f_0 is either the squared L2 regularization or L1 regularization or no regularization ($f_0 := 0$), and the choice of f_1 depends on the setting we use. We consider the following settings:

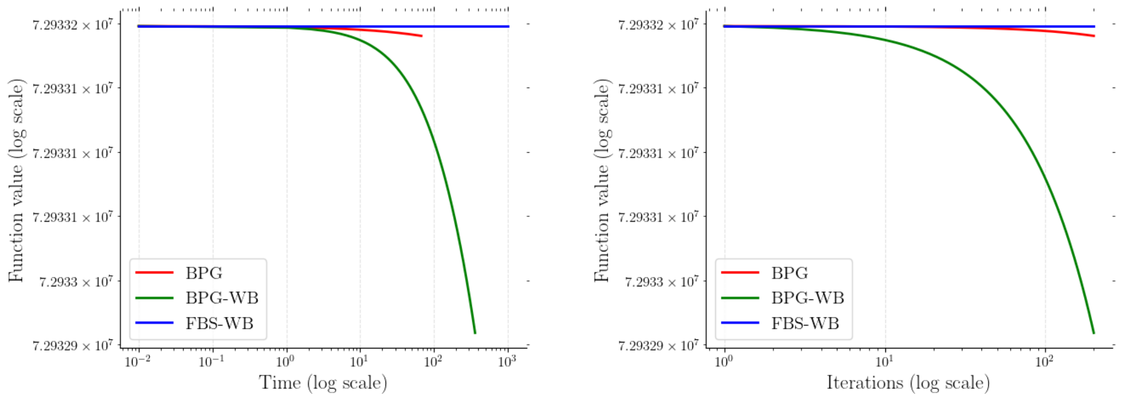
Regression setting with deep non-linear neural nets - Experiment A. Here, considering the setting as in Section 4.7, where we use the following choice of f_1:

$$f_1(W) := \frac{1}{2} \left\| \sigma_N(W_N \ldots \sigma_1(W_1 X)) - Y \right\|_F^2 \,. \tag{8.5.2}$$

We use sigmoid function as activation functions, that is $\sigma_i(x) = \frac{1}{1+e^{-x}}$ for $i = 1, \ldots, N$. We redo the calculation as in Lemma 4.7.2.3 to obtain that the following Legendre function and the objective in (8.5.2) satisfy the L-smad property:

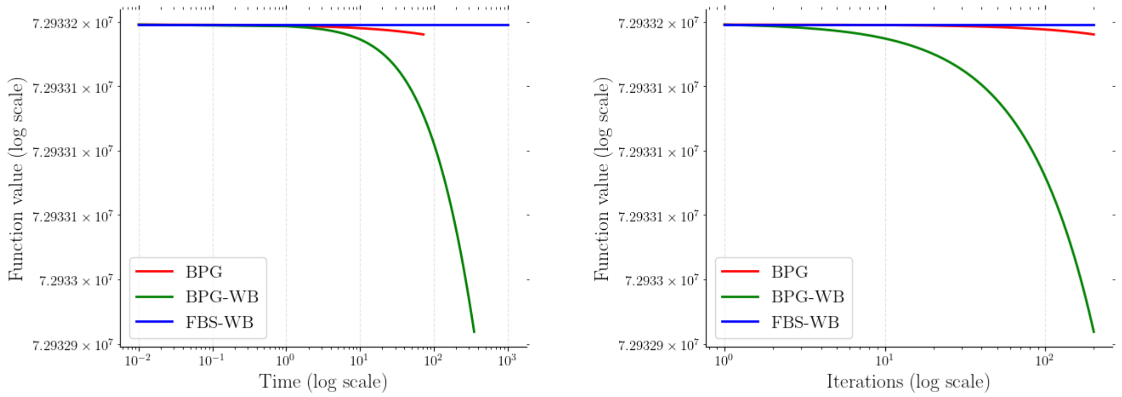
$$h(W) = c_1 \left(\frac{\|W\|_F^2}{N} \right) + c_2 \left(\frac{\|W\|_F^2}{N} \right)^{N+1} \,,$$

where $c_1 = \frac{N\tilde{c}_1}{2}$, $c_2 = \tilde{c}_2 (N+1)^N \left(\frac{N}{N+1} \right)^{N+1}$ with $\tilde{c}_1 = \frac{N}{2}\tilde{\Theta} + \frac{1}{4}\left(\|Y\|_F + \sqrt{\tilde{\Theta}} \right) 2\tilde{\Theta} + \frac{N-1}{4}(1 + \tilde{\Theta})$ and $\tilde{c}_2 = \frac{N}{2}\tilde{\Theta} + \frac{1}{4}\left(\|Y\|_F + \sqrt{\tilde{\Theta}} \right) \tilde{\Theta} + \frac{N-1}{4}$, $\tilde{\Theta} = \max\left(\left(\max_{i=1,\ldots,N} d_i d_0 \right), \|X\|_F^2 \right)$. For the regression setting, we use the Boston house pricing dataset [90] available at `https://raw.githubusercontent.com/jbrownlee/Datasets/master/housing.data` containing 506 samples with 13 features for each sample. The description regarding the Boston house pricing dataset can be found at `https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html`.



(a) Regression, L2-regularization (b) Regression, L2-regularization

FIGURE 8.1: Comparison on BPG, BPG-WB, FBS-WB on deep neural network with L2-regularization in regression setting. Here, BPG-WB outperforms other methods in terms of function values versus iterations or time.



(a) Regression, L1-regularization (b) Regression, L1-regularization

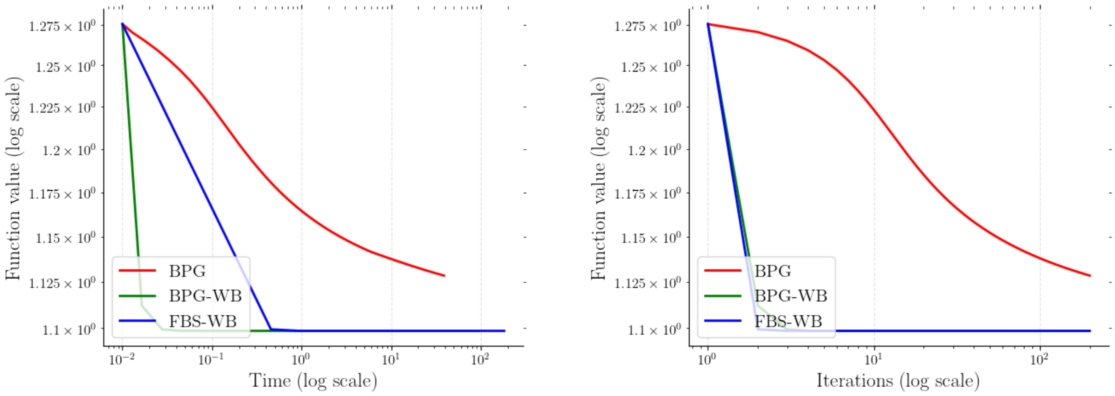
FIGURE 8.2: Comparison on BPG, BPG-WB, FBS-WB on deep neural network with L1-regularization in regression setting. Here, BPG-WB outperforms other methods in terms of function values versus iterations or time.

Classification setting with deep non-linear neural nets - Experiment B. Here, based on Section 4.8 we use the following choice of f_1:

$$f_1(W) := \sum_{j=1}^{M} \left(-\log \left(\frac{e^{z_{j,j_k}}}{\sum_{k=1}^{K} e^{z_{j,k}}} \right) \right) , \tag{8.5.3}$$

where $z_j := \sigma_N(W_N \ldots \sigma_1(W_1 x_j))$ for $j \in \{1, \ldots, M\}$. We use sigmoid function as activation functions, that is $\sigma_i(x) = \frac{1}{1+e^{-x}}$ for $i = 1, \ldots, 4$. We redo the calculation as in Lemma 4.8.1.1 to obtain that the following Legendre function and the objective in (8.5.2) satisfy the L-smad property:

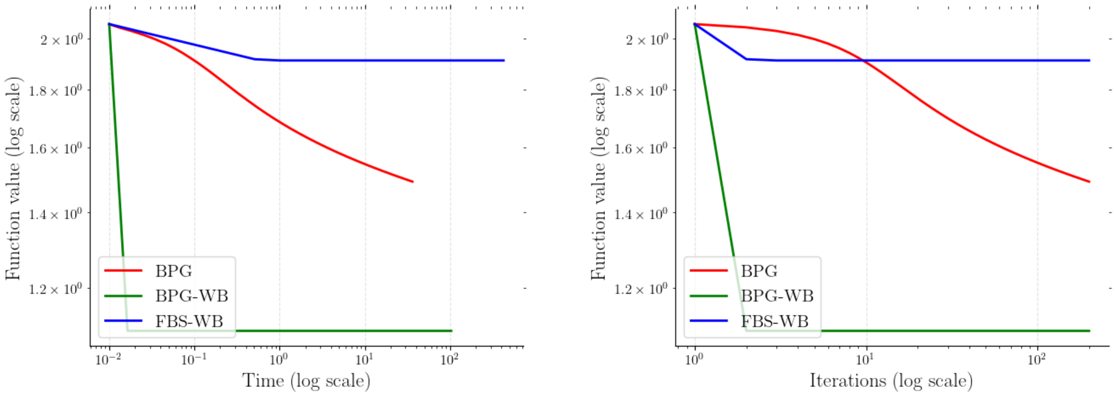
$$h(W) = \widehat{c}_1 \left(\frac{\|W\|_F^2}{N} \right) + \widehat{c}_2 \left(\frac{\|W\|_F^2}{N} \right)^{N+1} ,$$



(a) Classification, L2-regularization

(b) Classification, L2-regularization

FIGURE 8.3: Comparison on BPG, BPG-WB, FBS-WB on deep neural network with L2-regularization in classification setting. Here, BPG-WB outperforms other methods in terms of function values versus time and is competitive to FBS-WB in terms of function values versus iterations.



(a) Classification, L1-regularization

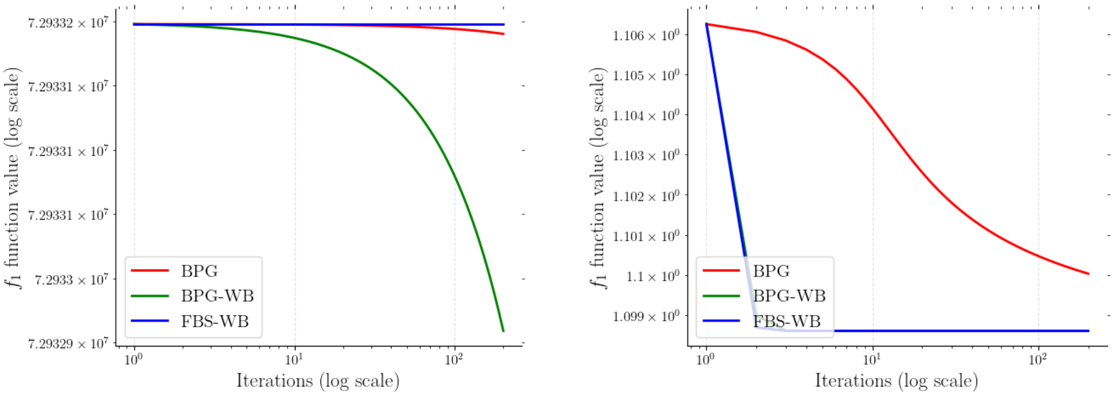
(b) Classification, L1-regularization

FIGURE 8.4: Comparison on BPG, BPG-WB, FBS-WB on deep neural network with L1-regularization in classification setting. Here, BPG-WB outperforms other methods in terms of function values versus iterations or time.

where $\widehat{c}_1 = 2\sqrt{K}N^2\tilde{\theta}$ and $\widehat{c}_2 = 2\sqrt{K}\tilde{\theta}\frac{N^{N+1}}{(N+1)}$. For the classification setting, we use the Iris dataset from `https://archive.ics.uci.edu/ml/datasets/iris` containing 150 samples with 4 features for each sample. In the Iris dataset, there are three class labels and for each label there are 50 samples.

In both the settings, we fix $N = 4$. We set $\mu_i = 0.1$ for all $i = 1, \ldots, N$ for the L1 regularization setting and $\lambda_0 = 0.1$ for squared L2 regularization. For the purpose of empirical comparisons, we use Bregman Proximal Gradient (BPG) algorithm, BPG with backtracking (BPG-WB), Forward–Backward Splitting with backtracking (FBS-WB) algorithms. For the L2-regularization setting, the results of regression and classification problems setting are given in Figures 8.1, 8.3. In Figures 8.2, 8.4, we illustrate the regression and classification setting under L1-regularization. We used the same initialization for all the algorithms.

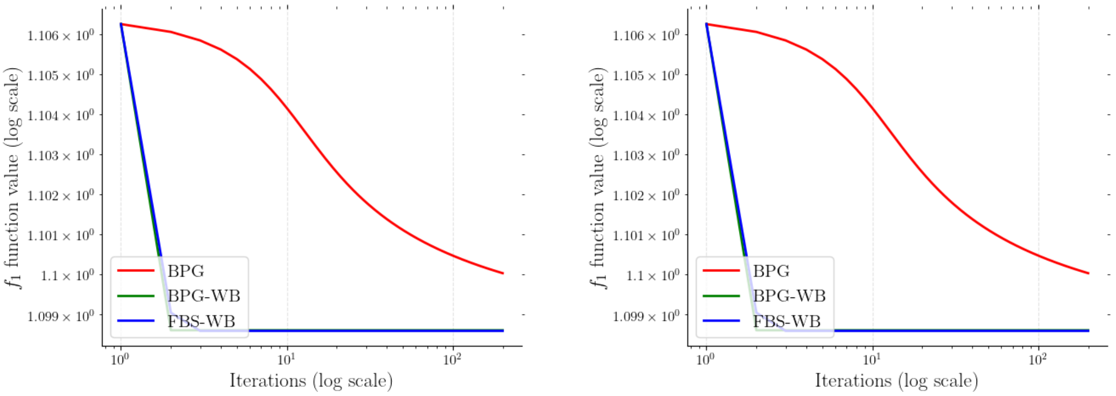
In the regression setting, we set the $d_4 = 1$, $d_3 = 5$, $d_2 = 5$ and $d_1 = 13$. In the classification setting, we set the $d_4 = 3$, $d_3 = 3$, $d_2 = 3$ and $d_1 = 4$. The choices of parameters for the backtracking step is the same for FBS-WB and BPG-WB. In all the plots, we observe that BPG-WB is competitive to FBS-WB. In terms of function value vs time, BPG-WB is faster compared to FBS-WB. We note that the Bregman distances used



(a) Regression, L2-regularization

(b) Classification, L2-regularization

FIGURE 8.5: We consider the plots of f_1 function values versus iterations in the context of regression and classification setting with L2-regularization. We compared BPG, BPG-WB and FBS-WB. Here, either BPG-WB is competitive to or outperforms other algorithms.



(a) Regression, L1-regularization

(b) Classification, L1-regularization

FIGURE 8.6: We consider the plots of f_1 function values versus iterations in the context of regression and classification setting with L1-regularization. We compared BPG, BPG-WB and FBS-WB. Here, either BPG-WB is competitive or outperforms other algorithms.

in BPG methods involve higher order terms which make the BPG methods unstable to initialization with large values. We leave the comprehensive exploration of the algorithms for future work. Also, the double backtracking step involved in CoCaIn BPG has resulted in severe numerical issues. This results in exploding function values or infinite loop while backtracking and this needs to be further explored. In Figures 8.5 and 8.6, we plot the f_1 function value versus iterations and see that BPG-WB is either outperforms other methods or is competitive to other methods.

8.6 Chapter conclusion

In this chapter, a constant step-size based algorithm with global convergence guarantees was proposed to train deep non-linear neural networks. For this purpose, we use suitable Bregman distances proposed in Section 4.7 to make the Bregman proximal minimization methods and their guarantees applicable. All the technical issues such as closed form updates and closed form inertia of CoCaIn BPG are resolved. Our empirical comparisons illustrate that BPG-WB is competitive to FBS-WB. However, in our preliminary observations CoCaIn BPG appears to face severe numerical issues, which needs to be resolved in future. Another open problem that still persists is the applicability of stochastic BPG in [55] to optimize the objectives mentioned in this chapter. Our work in this chapter can pave for a new a class of algorithms that have global convergence guarantees, suitable for various other deep neural network classes, such as residual deep neural networks.

Chapter 9

Model BPG

9.1 Abstract

The L-smad property cannot handle non-smooth functions, for example, simple non-smooth functions like $\left|x^4 - 1\right|$ and also many practical composite problems are out of scope. We fix this issue by proposing the MAP property, which generalizes the L-smad property and is also valid for a large class of non-convex non-smooth composite problems. Based on the proposed MAP property, we propose a globally convergent algorithm called Model BPG, that unifies several existing algorithms. The convergence analysis is based on a new Lyapunov function. We also numerically illustrate the superior performance of Model BPG on standard phase retrieval problems, robust phase retrieval problems, and Poisson linear inverse problems, when compared to a state of the art optimization method that is valid for generic non-convex non-smooth optimization problems.

9.2 Introduction

In the earlier chapters, we focussed on the additive composite problems. However, in this chapter, we focus on generic composite problems. In particular, we are interested in solving the following non-convex optimization problem:

$$(\mathcal{P}_M) \qquad \inf_{x\in\mathbb{R}^N} f(x),$$

where $f : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper lower semicontinuous function that is lower bounded. Special instances of the above mentioned problem include two broad classes of problems, namely, additive composite problems (Section 9.6.1) and composite problems (Section 9.6.2).

In this chapter, we design an abstract framework for provably globally convergent algorithms based on suitable approximations of the objective, where the convergence analysis is moreover driven by a requirement on the approximation quality. A classical special case is that of a continuously differentiable $f : \mathbb{R}^N \to \mathbb{R}$, whose gradient mapping is Lipschitz continuous over $\mathbb{R}^N$. For such a function, the following Descent Lemma (cf. Lemma 1.2.3 of [124])

$$-\frac{\underline{L}}{2}\|x-\bar{x}\|^2 \le f(x) - f(\bar{x}) - \langle \nabla f(\bar{x}), x-\bar{x}\rangle \le \frac{\bar{L}}{2}\|x-\bar{x}\|^2 , \quad \text{for all } x,\bar{x}\in\mathbb{R}^N , \tag{9.2.1}$$

which describes the approximation quality of the objective f by its linearization $f(\bar{x}) + \langle \nabla f(\bar{x}), x-\bar{x}\rangle$ in terms of a quadratic error estimate with certain $\underline{L}, \bar{L} > 0$. Such inequalities play a crucial role in designing algorithms that are used to minimize f. Gradient Descent is one such algorithm, which we focus here. We illustrate Gradient Descent in terms of sequential minimization of suitable approximations to the objective, based on the first order Taylor expansion – the linearization of f around the current iterate $x_k \in \mathbb{R}^N$. Consider the following *model function* at the iterate $x_k \in \mathbb{R}^N$:

$$f(x;x_k) := f(x_k) + \langle \nabla f(x_k), x - x_k\rangle \ , \tag{9.2.2}$$

where $\langle \,\cdot\,,\,\cdot\,\rangle$ denotes the standard inner product in the Euclidean vector space $\mathbb{R}^N$ of dimension N and $f(\,\cdot\,;x_k)$ is the linearization of f around x_k. Set $\tau > 0$. Now, the Gradient Descent update can be written equivalently as follows:

$$x_{k+1} = \mathrm{argmin}_{x\in\mathbb{R}^N}\left\{ f(x;x_k) + \frac{1}{2\tau}\|x-x_k\|^2 \right\} \quad \Leftrightarrow \quad x_{k+1} = x_k - \tau\nabla f(x_k)\,. \tag{9.2.3}$$

Its convergence analysis is essentially based on the Descent Lemma (9.2.1), which we reinterpret as a bound on the linearization error (model approximation error) of f. However, obviously (9.2.1) imposes a quadratic error bound, which cannot be satisfied in general.

We discussed earlier in Chapters 4, 5 that the L-smad property fixes this issue. We briefly recall the L-smad property. A continuously differentiable function $f : \mathbb{R}^N \to \mathbb{R}$ is L-smad with respect to a Legendre function $h : \mathbb{R}^N \to \mathbb{R}$ over $\mathbb{R}^N$ with $\bar{L}, \underline{L} > 0$, if the following condition holds true:

$$-\,\underline{L}D_h(x,\bar{x}) \le f(x) - f(\bar{x}) - \langle \nabla f(\bar{x}), x-\bar{x}\rangle \le \bar{L}D_h(x,\bar{x})\,, \quad \text{for any } x,\bar{x}\in\mathbb{R}^N . \tag{9.2.4}$$

We interpret these inequalities as a generalized distance measure for the linearization error of f. Similar to the Gradient Descent setting, minimization of $f(\bar{x}) + \langle \nabla f(\bar{x}), x-\bar{x}\rangle + \frac{1}{\tau}D_h(x,\bar{x})$ essentially results in the Bregman Proximal Gradient (BPG) algorithm's update step [28].

However, the L-smad property relies on the continuous differentiability of the function f, thus non-smooth functions as simple as $\left|x^4-1\right|$ or $\left|1-(xy)^2\right|$ or $\log(1+\left|1-(xy)^2\right|)$ cannot be captured under the L-smad property. Numerous difficult non-smooth optimization problems cannot be captured either. This motivates a more general notion than the L-smad property.

This lead us to the development of the MAP property (Definition 9.3.0.3), where MAP abbreviates Model Approximation Property. Consider a function $f:\mathbb{R}^N \to \mathbb{R}$ that is proper lower semicontinuous, and a Legendre function $h:\mathbb{R}^N \to \mathbb{R}$ with $\operatorname{dom} h = \mathbb{R}^N$. For certain $\bar{x} \in \mathbb{R}^N$, we consider generic model function $f(x;\bar{x})$ that is proper lsc and approximates the function around the model center $\bar{x}$, while preserving the local first order information (Definition 9.3.0.2). The MAP property is satisfied with the constants $\bar{L} > 0$ and $\underline{L} \in \mathbb{R}$ if for any $\bar{x} \in \mathbb{R}^N$ the following holds:

$$-\underline{L}D_h(x,\bar{x}) \leq f(x) - f(x;\bar{x}) \leq \bar{L}D_h(x,\bar{x})\,, \quad \forall x \in \mathbb{R}^N\,. \tag{9.2.5}$$

Note that we do not require the continuous differentiability of the function f. Our MAP property is inspired from [55], however, their work considers only the lower bound, and also they rely on decomposition of function into two components.

We illustrate the MAP property with a simple example. Consider a composite problem $f(x) = g(F(x)) := \left|x^4-1\right|$, where $F(x) := x^4-1$ is a continuously differentiable function over $\mathbb{R}$, and $g(x) := |x|$ is a Lipschitz continuous function over $\mathbb{R}$. Note that neither the Lipschitz continuity of the gradient nor the L-smad property is valid for this problem. However, the MAP property is valid here. At certain $\bar{x} \in \mathbb{R}$, we consider the model function that is given by $f(x;\bar{x}) := g(F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x}))$, where $\nabla F(\bar{x})$ is the Jacobian of F at $\bar{x}$. Then, with $\bar{L} = \underline{L} = 4$, the MAP property is satisfied:

$$-\underline{L}D_h(x,\bar{x}) \leq g(F(x)) - g(F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x})) \leq \bar{L}D_h(x,\bar{x})\,, \text{ for all } x,\bar{x} \in \mathbb{R}\,, \tag{9.2.6}$$

where $h(x) = \frac{1}{4}x^4$ and the generated Bregman distance is $D_h(x,\bar{x}) = \frac{1}{4}x^4 - \frac{1}{4}\bar{x}^4 - \bar{x}^3(x-\bar{x})$. We provide further details in Example 9.3.0.1 and in Example 9.3.0.2.

We considered the above given composite problem for illustration purposes, and we emphasize that our framework is applicable for large classes of non-convex problems (see Section 9.6). Similar to the BPG setting, minimization of $f(x;\bar{x}) + \frac{1}{\tau}D_h(x,\bar{x})$ essentially results in Model BPG algorithm's update step. The precise definition of the model function is provided in Definition 9.3.0.2, the MAP property in full generality is provided in Definition 9.3.0.3, and the Model BPG algorithm is provided in Algorithm 7.

We now discuss our main contributions and the related work.

9.2.1 Contributions

Our main contributions are the following.

- We introduce the MAP property, which generalizes the Lipschitz continuity assumption of the gradient mapping and the L-smad property [10, 28]. Earlier proposed notions were restricted to additive composite problems. The MAP property is essentially an extended Descent Lemma that is valid for generic composite problems (see Section 9.6), based on Bregman distances. Our theory is applicable to generic non-convex non-smooth objectives, and is not restricted to composite objectives. MAP like property was also partially considered in [55], however with focus on stochastic optimization. The MAP property relies on the notion of model function, that serves as a function approximation, and preserves the local first order information of the function. Our work extends the foundations laid by [55, 60] that consider generic model functions (potentially non-convex), and [139] which considers convex model functions.

- Based on the MAP property, Model based Bregman Proximal Gradient (Model BPG) algorithm (Algorithm 7) is proposed. Several existing algorithms such as Proximal Gradient Method [52], Bregman Proximal Gradient method [28] (or Mirror Descent [14]), Prox-Linear algorithm [62], and many other algorithms can be seen as a special case. Moreover, novel algorithms arise depending on the definition of the model function. We emphasize that Model BPG is practical, simple to implement and also does not require special knowledge about the problem such as the so-called information zone [27]. Close variants of Model BPG already exist in the literature, such as line search based Bregman Proximal Gradient method [139], and Mirror Descent variant [55], however, the convergence of the full sequence of iterates was not known.

- The standard global convergence analysis, in the sense that the full sequence of iterates converges to a single point, relies on descent properties of function values evaluated at the iterates of an algorithm. However, using function values can be restrictive, and alternatives are sought [142]. To fix this issue, we introduce a new Lyapunov function, through which we prove the global convergence of the full sequence of iterates generated by Model BPG. We eventually show that the sequence generated by Model BPG converges to a critical point of the objective function, which is potentially non-convex and non-smooth. Notably, the usage of a Lyapunov function is popular for inertial algorithms [137] (also see Chapter 5) and through our work we aim to popularize Lyapunov functions also for noninertial algorithms.

- The global convergence analysis of Bregman Proximal Gradient (BPG) [28] relies on the full domain of the Bregman distance. However, there are many Bregman distances for which the domain is restricted. We show in this chapter, that under certain assumptions that are typically satisfied in practice, the global convergence of the full sequence of iterates generated by Model BPG using generic Bregman distances can indeed be obtained (Theorem 9.5.5.2, 9.5.6.3). In general, this requires the limit points of the sequence to lie in the interior of domain of the employed Legendre function. While this is certainly a restriction, nevertheless, the considered setting is highly nontrivial and novel in the general context of non-convex non-smooth optimization. Moreover, it allows us to avoid the common restriction of requiring (global) strong convexity of the Legendre function, which is a severe drawback that rules out many interesting applications in related approaches (e.g., see Section 9.7.3).

- We provide a comprehensive numerical section showing the superior performance of Model BPG compared to a state of the art optimization algorithm, namely, inexact Bregman proximal minimization line search (IBPM-LS) [138], on standard phase retrieval problems, robust phase retrieval problems and Poisson linear inverse problems.

9.2.2 Related work

Our work is fundamentally based on three pillars, namely, Bregman distances, model functions, and Kurdyka–Łojasiewicz (KL) inequality. Bregman distances are certain generalized proximity measures, which generalize Euclidean distances (see Chapter 4). Model functions serve as function approximations which preserve local first order information about the function. The KL inequality is a certain regularity property of the function crucial for the global convergence analysis of Model BPG, and is typically satisfied by objectives that arise in practice (see Chapter 3). Here, we briefly review the related work on model functions. The rest of the related work regarding KL property and the Bregman proximal minimization is already considered in the earlier chapters.

The MAP property relies on the concept of the model function, which is essentially a function approximation that preserves the local first order information. In smooth optimization, it is common to use the Taylor approximation of a certain order as model function. In non-smooth optimization, we can only speak of "Taylor-like" models [60, 132, 133, 139], which is a (nonunique) approximation that satisfies certain error bound or a growth function [60, 139]. The class of model functions used in [132, 133] only satisfy a lower bound, and bundle methods are developed, which is a different class of algorithms that we do not discuss here. The growth functions in [60, 139] that measure the approximation quality of the model function, which is also used in this chapter, can be interpreted as a generalized first-order oracle. It has been shown in [139] that the concept of model functions unifies several algorithms for smooth and non-smooth optimization, for example, Gradient Descent, Proximal Gradient Descent, Levenberg Marquardt's method, ProxDescent, certain variable metric versions of these algorithms and some related majorization–minimization based algorithms. More recently, model functions were considered in the context of the Conditional Gradient Method in [140]. A particularly interesting class of model functions is the one for which the approximation quality measure is formed by Bregman distances [10, 28, 139], which is our main focus in this chapter.

9.3 Problem setting and Model BPG algorithm

We solve possibly non-smooth and non-convex optimization problems of the form

$$(\mathcal{P}_M) \qquad \inf_{x\in\mathbb{R}^N} f(x)\,, \tag{9.3.1}$$

that satisfy the following assumption, which we impose henceforth.

Assumption F. The objective function $f\colon \mathbb{R}^N \to \overline{\mathbb{R}}$ is proper, lower semi-continuous (possibly non-convex non-smooth) and a coercive function, i.e., as $\|x\| \to \infty$ we have $f(x) \to \infty$.

Due to [150, Theorem 1.9], the function f satisfying Assumption F is bounded from below, and $\operatorname{Argmin}_{x\in\mathbb{R}^N} f(x)$ is nonempty and compact. We denote the following:

$$v(\mathcal{P}_M) := \inf_{x\in\mathbb{R}^N} f(x) > -\infty\,.$$

We denote the set of critical points with respect to the limiting subdifferential as $\operatorname{crit} f$. We require the following technical definitions.

Definition 9.3.0.1 (Growth function [60, 139])**.** A differentiable univariate function $\varsigma\colon \mathbb{R}_+ \to \mathbb{R}_+$ is called *growth function* if it satisfies $\varsigma(0) = \varsigma'_+(0) = 0$, where ς'_+ denotes the one sided (right) derivative of ς. If, in addition, $\varsigma'_+(t) > 0$ for $t > 0$ and equalities $\lim_{t\searrow 0}\varsigma'_+(t) = \lim_{t\searrow 0}\varsigma(t)/\varsigma'_+(t) = 0$ hold, we say that ς is a *proper growth function.*

Example of a proper growth function is $\varsigma(t) = \frac{\eta}{r}t^r$ for $\eta, r > 0$. Lipschitz continuity and Hölder continuity can be interpreted with growth functions or, more generally, with uniform continuity [139]. We use the notion of a growth function to quantify the difference between a model function (defined below) and the objective function.

Definition 9.3.0.2 (Model function)**.** Let f be a proper lower semi-continuous (lsc) function. A function $f(\,\cdot\,,\bar{x})\colon \mathbb{R}^N \to \overline{\mathbb{R}}$ with $\operatorname{dom} f(\,\cdot\,,\bar{x}) = \operatorname{dom} f$ is called *model function* for f around the *model center* $\bar{x} \in \operatorname{dom} f$, if there exists a growth function $\varsigma_{\bar{x}}$ such that the following is satisfied:

$$|f(x) - f(x;\bar{x})| \le \varsigma_{\bar{x}}(\|x - \bar{x}\|)\,, \quad \forall\, x \in \operatorname{dom} f. \tag{9.3.2}$$

Model function is essentially a first-order approximation to a function f (see Lemma F.2.0.1), which explains the naming as "Taylor-like model" by [60]. The qualitative approximation property is represented by the growth function. We refer to (9.3.2) as a bound on the model error, and the symbol $\varsigma_{\bar{x}}$ denotes the dependency of the growth function on the model center $\bar{x}$.

Few remarks are in order, which we provide below:

- Informally, the model function approximates the function well near the model center. Convex model functions are explored in [139, 140], however in our setting, the model functions can be non-convex.
- Nonconvex model functions were considered in [60], however only subsequential convergence was shown. Their work is focussed on the termination criterion of the algorithms, however, they do not present an implementable algorithm.

If the growth function constants are independent of $\bar{x}$, this results in a uniform approximation. However, typically the growth function depends on the model center, as we illustrate below.

Example 9.3.0.1 (Running example)**.** Let $f(x) = |g(x)|$ with $g(x) = \|x\|^4 - 1$. With $\bar{x} \in \mathbb{R}^N$ as the model center, we consider the following model function:

$$f(x;\bar{x}) := |g(\bar{x}) + \langle \nabla g(\bar{x}), x - \bar{x}\rangle|\ .$$

As per the proof provided in Section F.1 in the appendix, the model error is given by

$$|f(x) - f(x;\bar{x})| \le 24\|\bar{x}\|^2\|x - \bar{x}\|^2 + 8\|x - \bar{x}\|^4\,,$$

where the growth function is $\varsigma_{\bar{x}}(t) = 24\|\bar{x}\|^2 t^2 + 8t^4$.

The above example illustrates that a constant in the growth function $\varsigma_{\bar{x}}(t)$ is dependent on the model center. It is often of interest to obtain a uniform approximation for the model error $|f(x) - f(x;\bar{x})|$, where the growth function is not dependent on the model center. In general, obtaining such a uniform approximation is not trivial, and may even be impossible. Moreover, typically finding an appropriate growth function is not trivial.

For this purpose, it is preferable to have a global bound on the model error, for which such a bound can be easily verified, the dependency on the model center is more structured, and the constants arising do not have any dependency on the model center. In the context of additive composite problems, previous works such as [10, 28, 109] relied on Bregman distances to upper bound the model error and verified the model error property with a simple convexity test based on second order information (c.f. [10, Proposition 1]). Based on this idea, we propose the following MAP property, which is valid for a huge class of generic non-convex problems and also generalizes the previous works. We emphasize that the MAP property is valid for a large class of non-smooth functions. MAP like property that is valid for composite problems was also explored in [55]. We provide the precise connections to previous works and examples in Section 9.6.

Definition 9.3.0.3 (MAP: Model approximation property)**.** Let h be a Legendre function that is continuously differentiable over $\operatorname{int}\operatorname{dom} h$. A proper lsc function f with $\operatorname{dom} f \subset \operatorname{cl}\operatorname{dom} h$ and $\operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h \neq \emptyset$, and model function $f(\,\cdot\,,\bar{x})$ for f around $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$ satisfies the *Model Approximation Property (MAP) at* $\bar{x}$, with the constants $\bar{L} > 0, \underline{L} \in \mathbb{R}$, if for any $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$ the following holds:

$$-\underline{L}D_h(x,\bar{x}) \leq f(x) - f(x;\bar{x}) \leq \bar{L}D_h(x,\bar{x})\,, \quad \forall x \in \operatorname{dom} f \cap \operatorname{dom} h\,. \tag{9.3.3}$$

Remark 9.3.0.1. We provide the following remarks.

- The design of a model function is independent of an algorithm. However, algorithms can be governed by the model function, for example, Model BPG in Algorithm 7. The property of a model function is rather an analogue to differentiability or a (uniform) first-order approximation. Note that for $\bar{x} \in \operatorname{int}\operatorname{dom} h$, the Bregman distance $D_h(x,\bar{x})$ is bounded by $o(\|x-\bar{x}\|)$, which is a growth function. Therefore, the MAP property requires additional algorithm specific properties of the model function. In particular, we require the constants $\bar{L}$ and $\underline{L}$ to be independent of $\bar{x}$, which provides a global consistency between the model function approximations.

- The condition $\operatorname{dom} f \subset \operatorname{cl}\operatorname{dom} h$ is a minor regularity condition. For example, if $\operatorname{dom} f = [0,\infty)$ and $\operatorname{dom} h = (0,\infty)$ (e.g., for h in Burg's entropy), such a function h can still be used in MAP property. However, the L-smad property [28] would require $x, \bar{x}$ in (9.3.3) to lie in $\operatorname{int}\operatorname{dom} h$ (see also Section 9.6.1).

- Note that the choice of $\underline{L}$ is unrestricted in MAP property. For non-convex f, $\underline{L}$ is typically a positive real number. For convex f typically the condition $\underline{L} \geq 0$ holds true. However, note that the values of $\underline{L}, \bar{L}$ are governed by the model function. In the context of convex additive composite problems, $\underline{L} < 0$ can hold true for relatively strongly convex functions [109].

- A closely related work in [55] considers only the lower bound of the MAP property and their algorithm terminates by choosing an iterate based on certain probability distribution. In stark contrast, Model BPG relies on the upper bound of the MAP property and there is no need to invoke any probabilistic argument to choose the final iterate. Also, [55] considers weakly convex model functions whereas we do not have such a restriction.

- For the global convergence analysis of Model BPG sequences, in addition to the condition $\tau_k \in [\underline{\tau}, \bar{\tau}]$ on step-size, the condition that $\tau_k \to \tau$, as $k \to \infty$ for certain $\tau > 0$ is required (see Theorem 9.5.5.2 , 9.5.6.3).

Example 9.3.0.2 (Running example – contd)**.** We continue Example 9.3.0.1 to illustrate the MAP property. Let $h(x) = \frac{1}{4}\|x\|^4$, we clearly have

$$g(x) - g(\bar{x}) - \langle \nabla g(\bar{x}), x - \bar{x}\rangle \leq 4D_h(x,\bar{x})\,, \quad \forall\, x \in \mathbb{R}^N\,,$$

which in turn results in the following upper bound for the model error

$$|f(x) - f(x;\bar{x})| \leq |g(x) - g(\bar{x}) - \langle \nabla g(\bar{x}), x - \bar{x}\rangle| \leq 4D_h(x,\bar{x})\,.$$

The upper bound is obtained in terms of a Bregman distance. Clearly, the constants arising do not have any dependency on the model center.

We now present Model BPG that we analyze for the setting of Assumption G.

Algorithm 7: Model BPG: Model based Bregman Proximal Gradient

- **Initialization:** Select $x_0 = x_1 \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$. Choose $\underline{\tau}, \bar{\tau}$ such that $0 < \underline{\tau} < \bar{\tau} < (1/\bar{L})$.
- **For each** $k \geq 1$**:** Choose $\tau_k \in [\underline{\tau}, \bar{\tau}]$ and compute

$$x_{k+1} \in \operatorname*{Argmin}_{x \in \mathbb{R}^N} \left\{ f(x; x_k) + \frac{1}{\tau_k} D_h(x, x_k) \right\} . \tag{9.3.4}$$

Assumption G. Let h be a Legendre function that is $\mathcal{C}^2$ over $\operatorname{int} \operatorname{dom} h$. Moreover, the conditions $\operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h \neq \emptyset$ and $\operatorname{crit} f \cap \operatorname{int} \operatorname{dom} h \neq \emptyset$ hold true.

- The exist $\bar{L} > 0, \underline{L} \in \mathbb{R}$ such that for any $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, the function f with $\operatorname{dom} f \subset \operatorname{cl} \operatorname{dom} h$, and model function $f(\,\cdot\,, \bar{x})$ for f around the model center $\bar{x}$ satisfies the *MAP property at* $\bar{x}$ with the constants $\bar{L}, \underline{L}$.
- For any $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, the following qualification condition holds true:

$$\partial_x^\infty f(x; \bar{x}) \cap (-N_{\operatorname{dom} h}(x)) = \{0\}\,, \quad \forall\, x \in \operatorname{dom} f \cap \operatorname{dom} h\,. \tag{9.3.5}$$

- For all $x, y \in \operatorname{dom} f$, the condition

$$(0, v) \in \partial^\infty f(x; y) \quad \text{implies} \quad v = 0\,, \quad \text{and} \quad (v, 0) \in \partial^\infty f(x; y) \quad \text{implies} \quad v = 0$$

  hold true. Moreover, $f(x; y)$ is regular [150, Definition 7.25] at any $(x, y) \in \operatorname{dom} f \times \operatorname{dom} f$.
- The function $f(x; \bar{x})$ is a proper, lsc function and is continuous over $(x, \bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$.

By $\partial_x f(x; \bar{x})$ we mean the limiting subdifferential of the model function $x \mapsto f(x; \bar{x})$ with $\bar{x}$ fixed and $\partial f(x; y)$ denotes the limiting subdifferential w.r.t (x, y); dito for the horizon subdifferential.

Discussion on Assumption G. The qualification condition in (9.3.5) is required for the applicability of the subdifferential summation rule (see [150, Corollary 10.9]). Assumption G(iii) and [150, Corollary 10.11] ensures that for all $x, y \in \operatorname{dom} f$, the following holds true:

$$\partial f(x; y) = \partial_x f(x; y) \times \partial_y f(x; y)\,, \; \partial^\infty f(x; y) = \partial_x^\infty f(x; y) \times \partial_y^\infty f(x; y)\,. \tag{Assumption G(iii)'}$$

Our analysis relies on (Assumption G(iii)'). However, note that Assumption G(iii) is a sufficient condition for (Assumption G(iii)') to hold. Certain classes of functions mentioned in Section 9.6 satisfy (Assumption G(iii)') directly, instead of Assumption G(iii). Assumption G(iv) is typically satisfied in practice and plays a key role in Lemma 9.5.6.2. Based on Assumption G(iii), for any fixed $\bar{x} \in \operatorname{dom} f$, the model function $f(x; \bar{x})$ is regular at any $x \in \operatorname{dom} f$. Using this fact, we deduce that the model function preserves the first order information of the function, in the sense that for $x \in \operatorname{dom} f$ the condition $\partial_y f(y; x)|_{y=x} = \widehat{\partial} f(x)$ holds true, which we prove in Lemma F.2.0.1 in the appendix. Many popular algorithms such as Gradient Descent, Proximal gradient method, Bregman Proximal Gradient method, Prox-Linear method are special cases of Model BPG depending on the choice of the model function and the choice of Bregman distance, thus making it a unified algorithm (also c.f. [139]). Examples of model functions are provided in Section 9.6, for which we verify all the assumptions. Other related model functions can also be found in [139, Section 5].

Let $\tau > 0$, $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, the update mapping from (9.3.4) of Model BPG is defined by

$$T_\tau(\bar{x}) := \underset{x \in \mathbb{R}^N}{\operatorname{Argmin}}\ f(x; \bar{x}) + \frac{1}{\tau} D_h(x, \bar{x}) \,. \tag{9.3.6}$$

Denote $\varepsilon_k := \left(\frac{1}{\tau_k} - \bar{L}\right) > 0$ and clearly $\underline{\varepsilon} \leq \varepsilon_k \leq \bar{\varepsilon}$, where $\bar{\varepsilon} := \frac{1}{\underline{\tau}} - \bar{L}$ and $\underline{\varepsilon} := \frac{1}{\bar{\tau}} - \bar{L}$.

Well-posedness of the update step (9.3.4) is given by the following result.

Lemma 9.3.0.1. *Let Assumption F, G hold true and let $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$. Then, for all $0 < \tau < \frac{1}{L}$ the set $T_\tau(\bar{x})$ is a nonempty compact subset of $\operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$.*

Proof. Firstly, note that as a consequence of MAP property due to Assumption G and nonnegativity of Bregman distances, the following condition is satisfied

$$f(x) \leq f(x; \bar{x}) + \frac{1}{\tau} D_h(x, \bar{x}) \,, \quad \forall\, x \in \operatorname{dom} f \cap \operatorname{dom} h \,. \tag{9.3.7}$$

If the set $\operatorname{dom} f \cap \operatorname{dom} h$ is bounded, the objective $f(\,\cdot\,; \bar{x}) + \frac{1}{\tau} D_h(\,\cdot\,, \bar{x})$ is coercive. Otherwise, the coercivity of f implies that the objective $f(\,\cdot\,; \bar{x}) + \frac{1}{\tau} D_h(\,\cdot\,, \bar{x})$ is coercive, due to (9.3.7). Then, the result follows from a simple application of [98, Lemma 3.6] and [150, Theorem 1.9]. □

The conclusion of the lemma remains true under other sufficient conditions. For instance, if the model has an affine minorant and h is supercoercive (for example, see [28, Section 3.1]). We now show that Model BPG results in monotonically nonincreasing function values.

Proposition 9.3.0.1 (Sufficient descent property in function values)**.** *Let Assumptions F, G hold. Also, let $(x_k)_{k \in \mathbb{N}}$ be a sequence generated by Model BPG, then the following holds for $k \geq 1$*

$$f(x_{k+1}) \leq f(x_k) - \varepsilon_k D_h(x_{k+1}, x_k) \,. \tag{9.3.8}$$

We provide the proof of Proposition 9.3.0.1 in Section F.3 in the appendix.

Remark 9.3.0.2. Under Assumptions F, G, the coercivity of f along with Proposition 9.3.0.1 implies that the iterates of Model BPG lie in the compact set $\{x : f(x) \leq f(x_0)\}$, thus bounded.

9.4 Gradient-like Descent sequence

We briefly review the concept of Gradient-like Descent sequence from [136]. For ease of global convergence analysis of Model BPG we use following results from [136]. Let $\mathcal{F} \colon \mathbb{R}^N \times \mathbb{R}^P \to \overline{\mathbb{R}}$ be a proper, lower semi-continuous function that is bounded from below, then assume the following assumption from [136] holds.

Assumption H (Gradient-like Descent sequence [136])**.** Let $(u_n)_{n \in \mathbb{N}}$ be a sequence of parameters in $\mathbb{R}^P$ and let $(\varepsilon_n)_{n \in \mathbb{N}}$ be an ℓ_1-summable sequence of non-negative real numbers. Moreover, we assume there are sequences $(a_n)_{n \in \mathbb{N}}$, $(b_n)_{n \in \mathbb{N}}$, and $(d_n)_{n \in \mathbb{N}}$ of non-negative real numbers, a non-empty finite index set $I \subset \mathbb{Z}$ and $\theta_i \geq 0$, $i \in I$, with $\sum_{i \in I} \theta_i = 1$ such that the following holds:

(i) (Sufficient decrease condition) For each $n \in \mathbb{N}$, it holds that

$$\mathcal{F}(x_{n+1}, u_{n+1}) + a_n d_n^2 \leq \mathcal{F}(x_n, u_n)\,.$$

(ii) (Relative error condition) For each $n \in \mathbb{N}$, the following holds: (set $d_j = 0$ for $j \leq 0$)

$$b_{n+1} \|\partial \mathcal{F}(x_{n+1}, u_{n+1})\|_- \leq b \sum_{i \in I} \theta_i d_{n+1-i} + \varepsilon_{n+1}\,.$$

(iii) (Continuity condition) There exists a subsequence $((x_{n_j}, u_{n_j}))_{j \in \mathbb{N}}$ and $(\tilde{x}, \tilde{u}) \in \mathbb{R}^N \times \mathbb{R}^P$ such that

$$(x_{n_j}, u_{n_j}) \xrightarrow{\mathcal{F}} (\tilde{x}, \tilde{u}) \quad \text{as} \quad j \to \infty\,.$$

(iv) (Distance condition) It holds that

$$\begin{aligned} &d_n \to 0 \Longrightarrow \|x_{n+1} - x_n\|_2 \to 0 \qquad \text{and} \\ &\exists n' \in \mathbb{N} \colon \forall n \geq n' \colon d_n = 0 \Longrightarrow \exists n'' \in \mathbb{N} \colon \forall n \geq n'' \colon x_{n+1} = x_n \end{aligned}$$

(v) (Parameter condition) It holds that

$$(b_n)_{n \in \mathbb{N}} \notin \ell_1\,, \quad \sup_{n \in \mathbb{N}} \frac{1}{b_n a_n} < \infty\,, \quad \inf_n a_n =: \underline{a} > 0\,.$$

Such an assumption is crucial in order to obtain global convergence of the sequences generated by Model BPG. Assumption H is more general compared to the conditions that arise in standard Gradient-like Descent sequence [28], which is basically based on the first three conditions.

We now provide the global convergence statement from [136], based on Assumption H. Firstly, denote the following. The set of limit points of a bounded sequence $((x_n, u_n))_{n \in \mathbb{N}}$ is given by $\omega(x_0, u_0) := \limsup_{n \to \infty} \{(x_n, u_n)\}$, and the subset of $\mathcal{F}$-attentive limit points is denoted by

$$\omega_{\mathcal{F}}(x_0, u_0) := \left\{ (\bar{x}, \bar{u}) \in \omega(x_0, u_0) \,|\, (x_{n_j}, u_{n_j}) \xrightarrow{\mathcal{F}} (\bar{x}, \bar{u}) \text{ for } j \to \infty \right\}.$$

Theorem 9.4.0.1 (Global convergence [136, Theorem 10])**.** *Suppose $\mathcal{F}$ is a proper lower semi-continuous Kurdyka–Łojasiewicz function that is bounded from below. Let $(x_n)_{n \in \mathbb{N}}$ be a bounded sequence generated by an abstract algorithm parametrized by a bounded sequence $(u_n)_{n \in \mathbb{N}}$ that satisfies Assumption H. Assume that $\mathcal{F}$-attentive convergence holds along converging subsequences of $((x_n, u_n))_{n \in \mathbb{N}}$, i.e. $\omega(x_0, u_0) = \omega_{\mathcal{F}}(x_0, u_0)$. Then, the following holds:*

(i) The sequence $(d_n)_{n \in \mathbb{N}}$ satisfies $\sum_{k=0}^{\infty} d_k < +\infty$, i.e., the trajectory of the sequence $(x_n)_{n \in \mathbb{N}}$ has finite length with respect to the abstract distance measures $(d_n)_{n \in \mathbb{N}}$.

(ii) Suppose d_k satisfies $\|x_{k+1} - x_k\|_2 \leq \bar{c} d_{k+k'}$ for some $k' \in \mathbb{Z}$ and $\bar{c} \in \mathbb{R}$, then $\sum_{k=0}^{\infty} \|x_{k+1} - x_k\|_2 < +\infty$, and the trajectory of the sequence $(x_n)_{n \in \mathbb{N}}$ has a finite Euclidean length, and thus $(x_n)_{n \in \mathbb{N}}$ converges to $\tilde{x}$ from (iii).

(iii) Moreover, if $(u_n)_{n \in \mathbb{N}}$ is a converging sequence, then each limit point of $((x_n, u_n))_{n \in \mathbb{N}}$ is a critical point, which in the situation of (ii) is the unique point $(\tilde{x}, \tilde{u})$ from (iii).

9.5 Global convergence analysis of Model BPG algorithm

The convergence analysis of most algorithms in non-convex optimization is based on a descent property. Usually, the objective value is shown to decrease, for example, as in Proposition 9.3.0.1 and in the analysis of additive composite problems [28, Lemma 4.1]. However, function values proved to be restrictive, primarily because the same techniques as additive composite problems do not work anymore for general composite problems, and alternatives like [142] are sought after.

9.5.1 New Lyapunov function

Here, we discuss one of our main contribution. We propose a Lyapunov function as our measure of progress. The Lyapunov function $F_{\bar{L}}^h$ is given by

$$F_{\bar{L}}^h \colon \mathbb{R}^N \times \mathbb{R}^N \to \overline{\mathbb{R}}\,, \quad (x, \bar{x}) \mapsto f(x;\bar{x}) + \bar{L}D_h(x,\bar{x})\,, \tag{9.5.1}$$

and $\operatorname{dom} F_{\bar{L}}^h = (\operatorname{dom} f)^2 \cap (\operatorname{dom} h \times \operatorname{int}\operatorname{dom} h)$. The set of critical points of the above given Lyapunov function is given by

$$\operatorname{crit} F_{\bar{L}}^h := \left\{(x,\bar{x}) \in \mathbb{R}^N \times \mathbb{R}^N :\ (0,0) \in \partial F_{\bar{L}}^h(x,\bar{x})\right\}\,. \tag{9.5.2}$$

Usage of Lyapunov functions is a popular strategy in the analysis of inertial methods [137] (Chapter 5). Even though our algorithm is non-inertial in nature, we show that the above defined Lyapunov function is suitable for the global convergence analysis. Certain previous works such as [114] considered a Lyapunov function based analysis for (non-inertial) Forward–Douglas–Rachford splitting method. Also, Lyapunov function based analysis is popular in the context of dynamical systems [81].

The motivation for using the Lyapunov function $F_{\bar{L}}^h$ instead of the function f is the following. In each iteration of Model BPG, we optimize the model function with a proximity measure, and the analysis with our proposed Lyapunov function reflects this explicitly, unlike the function value. The proposed Lyapunov function is related to the Bregman-Moreau envelope [98] of the model function $f(\,\cdot\,;\bar{x})$ where $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$. Under certain special case of the model function (Section 9.6.1), such a Bregman-Moreau envelope is related to the Bregman forward-backward envelope [2]. In the context where the Bregman distance is set to the Euclidean distance, the related works which consider value function based analysis is provided [25, 142, 158].

We now look at some properties of $F_{\bar{L}}^h$.

Proposition 9.5.1.1. *The Lyapunov function defined in (9.5.1) satisfies the following properties:*

(i) *For all $x \in \operatorname{dom} f \cap \operatorname{dom} h$ and $y \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, we have $f(x) \le F_{\bar{L}}^h(x,y)$.*

(ii) *For all $x \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, we have $F_{\bar{L}}^h(x,x) = f(x)$.*

(iii) *Moreover, we have*

$$\inf_{(x,y)\,\in\,\mathbb{R}^N\times\mathbb{R}^N} F_{\bar{L}}^h(x,y) \ge v(\mathcal{P}_M) > -\infty\,. \tag{9.5.3}$$

Proof. (i) This follows from MAP property and the definition of $F_{\bar{L}}^h$.

(ii) Substituting $y = x$ in (9.5.1) gives the result.

(iii) By MAP property, for all $(x,y) \in \operatorname{dom} F_{\bar{L}}^h$ we have the following:

$$v(\mathcal{P}_M) \le f(x) \le f(x;y) + \bar{L}D_h(x,y)\,.$$

Furthermore, we obtain the following:

$$\inf_{x\in \operatorname{dom} f \cap \operatorname{dom} h} f(x) \le \inf_{(x,y)\in \operatorname{dom} F_{\bar{L}}^h} \left(f(x;y) + \bar{L} D_h(x,y) \right) .$$

The statement follows using $\inf_{x\in\mathbb{R}^N} f(x) = v(\mathcal{P}_M) > -\infty$ due to Assumption F .

□

Equipped with the Lyapunov function $F_{\bar{L}}^h$, we focus now on the global convergence result of Model BPG. Our global convergence analysis is broadly divided into the following five parts.

- **Sufficient descent property.** In Section 9.5.2, we show that the sequence generated by Model BPG results in monotonically nonincreasing Lyapunov function values.
- **Relative error condition.** In Section 9.5.3, based on certain additional assumptions, we show that the infimal norm of the subdifferential of the Lyapunov function can be upper bounded by an entity that depends on the difference of successive iterates, and that entity tends towards zero asymptotically, implying stationarity in the limit.
- **Subsequential convergence.** In Section 9.5.4, we explore the behavior of limit points obtained from the sequence generated by Model BPG. We prove $F_{\bar{L}}^h$-attentive convergence along converging subsequences. Moreover, we prove that the set of $F_{\bar{L}}^h$-attentive limit points is compact, connected and $F_{\bar{L}}^h$ is constant on this set. When all limit points of the sequence generated by Model BPG lie in $\operatorname{int} \operatorname{dom} h$, we show that all the limit points are critical points of the Lyapunov function.
- **Global convergence to stationarity point of the Lyapunov function.** Under the condition that the Lyapunov function satisfies Kurdyka–Łojasiewicz property, we show in Section 9.5.5 that the full sequence generated by Model BPG converges to a point x such that (x, x) is a critical point of the Lyapunov function. However, the relation of x to the function f is not imminent here.
- **Global convergence to stationarity point of the function.** In Section 9.5.6, we prove that the update mapping is continuous and also show that fixed points of the update mapping are critical points of f. We exploit these properties to deduce that the full sequence of iterates generated by Model BPG converges to a critical point of f.

9.5.2 Sufficient descent property

We have already proved the sufficient descent property in terms of function values in Proposition 9.3.0.1. Here, we prove the sufficient descent property of the Lyapunov function.

Proposition 9.5.2.1 (Sufficient descent property)**.** *Let Assumptions F, G hold. Also, let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model BPG, then the following holds for $k \ge 1$*

$$F_{\bar{L}}^h(x_{k+1}, x_k) \le F_{\bar{L}}^h(x_k, x_{k-1}) - \varepsilon_k D_h(x_{k+1}, x_k) . \tag{9.5.4}$$

Proof. By global optimality of x_{k+1} as in (9.3.4), we have

$$f(x_{k+1}; x_k) + \frac{1}{\tau_k} D_h(x_{k+1}, x_k) \le f(x_k; x_k) = f(x_k) .$$

We have the following inequality from the MAP property

$$f(x_k; x_k) = f(x_k) \le f(x_k; x_{k-1}) + \bar{L} D_h(x_k, x_{k-1}) \,.$$

Thus, the result follows from the definition of $F^h_{\bar{L}}$ in (9.5.1). □

Proposition 9.5.2.2. *Let Assumptions F, G hold and let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model BPG. The following assertions hold:*

(i) *The sequence $\left\{F^h_{\bar{L}}(x_{k+1}, x_k)\right\}_{k\in\mathbb{N}}$ is nonincreasing and converges to a finite value.*

(ii) *$\sum_{k=1}^{\infty} D_h(x_{k+1}, x_k) < \infty$, and hence the sequence $\{D_h(x_{k+1}, x_k)\}_{k\in\mathbb{N}}$ converges to zero.*

(iii) *For any $n \in \mathbb{N}$, the condition*

$$\min_{1\le k\le n} D_h(x_{k+1}, x_k) \le \frac{F^h_{\bar{L}}(x_1, x_0) - v(\mathcal{P}_M)}{\underline{\varepsilon} n}$$

holds true.

Proof. (i) Nonincreasing property follows trivially from Proposition 9.5.2.1 and as $\varepsilon_k > 0$. We know from Proposition 9.5.1.1(iii) that the Lyapunov function is lower bounded, which implies convergence of $\left\{F^h_{\bar{L}}(x_{k+1}, x_k)\right\}_{k\in\mathbb{N}}$ to a finite value.

(ii) Let n be a positive integer. Summing (9.5.4) from $k = 1$ to n and using $\underline{\varepsilon} \le \varepsilon_k$ we get

$$\sum_{k=1}^{n} D_h(x_{k+1}, x_k) \le \frac{1}{\underline{\varepsilon}} \left(F^h_{\bar{L}}(x_1, x_0) - F^h_{\bar{L}}(x_{n+1}, x_n)\right) \le \frac{1}{\underline{\varepsilon}} \left(F^h_{\bar{L}}(x_1, x_0) - v(\mathcal{P}_M)\right) , \tag{9.5.5}$$

since $F^h_{\bar{L}}(x_{n+1}, x_n) \ge v(\mathcal{P}_M)$. Taking the limit as $n \to \infty$, we obtain the first assertion, from which we immediately deduce that $\{D_h(x_{k+1}, x_k)\}_{k\in\mathbb{N}}$ converges to zero.

(iii) From (B.4.1) we also obtain,

$$n \min_{1\le k\le n} \left(D_h(x_{k+1}, x_k)\right) \le \sum_{k=1}^{n} \left(D_h(x_{k+1}, x_k)\right) \le \frac{1}{\underline{\varepsilon}} \left(F^h_{\bar{L}}(x_1, x_0) - v(\mathcal{P}_M)\right) ,$$

which after division by n yields the result. □

9.5.3 Relative error condition

For the purposes of analysis, we require the following assumption.

Assumption I. We have the following conditions:

- Consider any bounded set $B \subset \operatorname{dom} f$. There exists $c > 0$ such that for any $x, y \in B$ we have

$$\|\partial_y f(x; y)\|_{-} \le c\|x - y\| \,.$$

- The function h has bounded second derivative on any compact subset $B \subset \operatorname{int}\operatorname{dom} h$.

- For bounded $(u_k)_{k\in\mathbb{N}}$, $(v_k)_{k\in\mathbb{N}}$ in $\operatorname{int}\operatorname{dom} h$, the following holds as $k \to \infty$:

$$D_h(u_k, v_k) \to 0 \quad \iff \quad \|u_k - v_k\| \to 0\,.$$

Through Example 9.5.3.1, we illustrate Assumption I(i), which governs the variation of the model function w.r.t. model center. Assumption I(ii) is a standard condition required for the analysis of Bregman proximal methods [28, 139](Chapter 5). Assumption I(iii) essentially states that the asymptotic behavior of vanishing Bregman distance is equivalent to that of vanishing Euclidean distance (cf. [139, Remark 18]). Such a condition is satisfied for many Bregman distances, such as those distances based on Boltzmann–Shannon entropy [139, Example 40] and Burg entropy [139, Example 41].

Example 9.5.3.1. We continue Example 9.3.0.1 to illustrate Assumption I(i). A quick calculation reveals that $\nabla^2 g(x)$ is bounded over bounded sets. Consider any bounded set $B \subset \mathbb{R}^N$. Define $c := \sup_{\bar{x}\in B} \|\nabla^2 g(\bar{x})\|$ and choose any $\bar{x} \in B$, then consider the model function given by :

$$f(x;\bar{x}) := |g(\bar{x}) + \langle \nabla g(\bar{x}), x - \bar{x}\rangle|\ .$$

The subdifferential of the model function is given by

$$\partial_{\bar{x}} f(x;\bar{x}) = u\nabla^2 g(\bar{x})(x-\bar{x})\,,$$

where $u \in \partial_{g(\bar{x})+\langle \nabla g(\bar{x}),x-\bar{x}\rangle} |g(\bar{x}) + \langle \nabla g(\bar{x}), x - \bar{x}\rangle|$. Considering the fact that $\|u\| \le 1$ and by the definition of c we have the following:

$$\inf_{v\in\partial_{\bar{x}} f(x;\bar{x})} \|v\| \le c\|x-\bar{x}\|\,,$$

which verifies Assumption I(i).

Now, we look at the relative error condition, which bounds the infimal norm of the subdifferential of the Lyapunov function, i.e., $\inf_{v\in\partial F^h_{\bar{L}}(x_{k+1},x_k)} \|v\|$, with the term $\|x_{k+1} - x_k\|$ upto a scaling factor. Such a bound is useful to achieve stationarity asymptotically, and plays a crucial role in proving global convergence. Note that with the descent property (Proposition 9.5.2.1) and Assumption I(iii), we have $\|x_{k+1} - x_k\| \to 0$.

Lemma 9.5.3.1 (Relative error). *Let Assumptions F, G, I hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by Model BPG lie in a compact set in* $\operatorname{int}\operatorname{dom} h$*, then there exists a constant $C > 0$ such that for certain $k \ge 0$, we have*

$$\|\partial F^h_{\bar{L}}(x_{k+1}, x_k)\|_- \le C\|x_{k+1} - x_k\|\,, \tag{9.5.6}$$

where $\|\partial F^h_{\bar{L}}(x_{k+1}, x_k)\|_- := \inf_{v\in\partial F^h_{\bar{L}}(x_{k+1},x_k)} \|v\|$.

Proof. As per [150, Exercise 8.8] or [116, Theorem 2.19], the subdifferential $\partial F^h_{\bar{L}}(x_{k+1}, x_k)$ is given by

$$\partial F^h_{\bar{L}}(x_{k+1}, x_k) = \partial f(x_{k+1}; x_k) + \bar{L}\nabla D_h(x_{k+1}, x_k)\,, \tag{9.5.7}$$

because the Bregman distance is continuously differentiable around $x_k \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$. Using [150, Corollary 10.11], Assumption G(iv), and using the fact that h is $\mathcal{C}^2$ over $\operatorname{int}\operatorname{dom} h$ (cf. Assumption G) we obtain

$$\partial F^h_{\bar L}(x_{k+1},x_k) = \Big(\partial_{x_{k+1}} f(x_{k+1};x_k) + \bar L\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big),\\ \partial_{x_k} f(x_{k+1};x_k) - \bar L\nabla^2 h(x_k)(x_{k+1}-x_k)\Big)\,. \tag{9.5.8}$$

Consider the following:

$$\begin{aligned}\inf_{\zeta\in\partial F(x_{k+1},x_k)} \|v\| &= \inf_{\xi\in\partial f(x_{k+1};x_k)} \|\xi + \bar L\nabla D_h(x_{k+1};x_k)\|\,,\\ &= \left(\inf_{(\xi_x,\xi_y)\in\partial f(x_{k+1};x_k)} \|(\xi_x,\xi_y) + \bar L\nabla D_h(x_{k+1},x_k)\|\right)\,,\\ &\le \left(\inf_{\xi_x\in\partial_{x_{k+1}} f(x_{k+1};x_k)} \|(\xi_x + \bar L\big(\nabla h(x_{k+1}) - \nabla h(x_k)))\|\right)\\ &+ \left(\inf_{\xi_y\in\partial_{x_k} f(x_{k+1};x_k)} \|(\xi_y + \bar L\nabla^2 h(x_k)(x_{k+1}-x_k))\|\right)\,,\end{aligned} \tag{9.5.9}$$

where in the first equality we use (9.5.7), in the second equality we use the result in (9.5.8) with $\xi := (\xi_x,\xi_y)$ such that $\xi_x \in \partial_{x_{k+1}} f(x_{k+1},x_k)$ and $\xi_y \in \partial_{x_k} f(x_{k+1},x_k)$, and in the last step we used

$$\nabla D_h(x_{k+1},x_k) = (\nabla h(x_{k+1}) - \nabla h(x_k), \nabla^2 h(x_k)(x_{k+1}-x_k))\,. \tag{9.5.10}$$

The optimality of x_{k+1} in (9.3.4) implies the existence of $\xi^{k+1}_{x_{k+1}} \in \partial_{x_{k+1}} f(x_{k+1};x_k)$ such that the following condition holds:

$$\xi^{k+1}_{x_{k+1}} + \frac{1}{\tau_k}(\nabla h(x_{k+1}) - \nabla h(x_k)) = 0\,. \tag{9.5.11}$$

Therefore, the first block coordinate in (9.5.8) satisfies

$$\xi^{k+1}_{x_{k+1}} + \bar L\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big) = \varepsilon_k\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big)\,. \tag{9.5.12}$$

Now consider the first term of the right hand side in (9.5.9). We have

$$\begin{aligned}\inf_{\xi_x\in\partial_{x_{k+1}} f(x_{k+1};x_k)} \|(\xi_x + \bar L\big(\nabla h(x_{k+1}) - \nabla h(x_k)))\| &\le \|\xi^{k+1}_{x_{k+1}} + \bar L\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big)\|\,,\\ &\le \varepsilon_k\|\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big)\|\,,\\ &\le \varepsilon_k\tilde L_h\|x_{k+1}-x_k\|\,,\end{aligned}$$

where in the second step we used (9.5.12) and in the last step we applied mean value theorem along with the fact that the entity $\|\nabla^2 h(x_{k+1} + s(x_{k+1}-x_k))\|$ is bounded by a constant $\tilde L_h > 0$ for certain $s\in[0,1]$, due to Assumption I(ii). Considering the second term of the right hand side in (9.5.9), we have

$$\begin{aligned}&\inf_{\xi_y\in\partial_{x_k} f(x_{k+1};x_k)} \|(\xi_y + \bar L\nabla^2 h(x_k)(x_{k+1}-x_k))\|\\ &\le \inf_{\xi_y\in\partial_{x_k} f(x_{k+1};x_k)} \|\xi_y\| + \|\bar L\nabla^2 h(x_k)(x_{k+1}-x_k)\|\,,\\ &\le c\|x_{k+1}-x_k\| + \bar L L_h\|(x_{k+1}-x_k)\|\,,\end{aligned}$$

where in the last step we used Assumption I(i) and the fact that $\|\nabla^2 h(x_k)\|$ is bounded by L_h. The result follows from combining the results obtained for (9.5.12). □

9.5.4 Subsequential convergence

We now consider results on generic limit points and show that stationarity can indeed be attained for iterates produced by Model BPG. The set of limit points of some sequence $(x_k)_{k\in\mathbb{N}}$ is denoted as follows

$$\omega(x_0) := \left\{ x \in \mathbb{R}^N \,|\, \exists K \subset \mathbb{N}\colon x_k \underset{k\in K}{\to} x \right\} ,$$

and its subset of f-attentive limit points

$$\omega_f(x_0) := \left\{ x \in \mathbb{R}^N \,|\, \exists K \subset \mathbb{N}\colon (x_k, f(x_k)) \underset{K}{\to} (x, f(x)) \right\} .$$

We explore below certain properties that are generic to any bounded sequence, and are later helpful to quantify properties of the sequence generated by Model BPG.

Proposition 9.5.4.1. *For a bounded sequence $(x_k)_{k\in\mathbb{N}}$ such that $\|x_{k+1} - x_k\| \to 0$ as $k \to \infty$, the following holds:*

(i) $\omega(x_0)$ *is connected and compact,*

(ii) $\lim_{k\to\infty} \operatorname{dist}(x_k, \omega(x_0)) = 0$.

The proof relies on the same technique as the proof of [26, Lemma 3.5] (also see [26, Remark 3.3]).

We now show that the sequence generated by Model BPG $(x_k)_{k\in\mathbb{N}}$ indeed attains $\|x_{k+1} - x_k\| \to 0$ as $k \to \infty$, which in turn enables the application of Proposition 9.5.4.1 to deduce the properties of the sequence generated by Model BPG, which later proves to be crucial for the proof of global convergence.

Proposition 9.5.4.2. *Let Assumption F, G, I hold. Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model BPG. Then, we have*

$$\underline{\varepsilon} D_h(x_{k+1}, x_k) \to 0\,, \qquad \text{as } k \to \infty\,. \tag{9.5.13}$$

The condition $\underline{\varepsilon} > 0$ implies that $x_{k+1} - x_k \to 0$ as $k \to \infty$.

Proof. Note that the sequence $(x_k)_{k\in\mathbb{N}}$ is a bounded sequence (see Remark 9.3.0.2). By the descent property (Proposition 9.5.2.1) and using $\varepsilon_k \geq \underline{\varepsilon}$ we have after rearranging

$$\underline{\varepsilon} D_h(x_{k+1}, x_k) \leq F^h_{\overline{L}}(x_k, x_{k-1}) - F^h_{\overline{L}}(x_{k+1}, x_k)\,.$$

Summing on both sides and due to the convergence of Lyapunov function, using Proposition 9.5.2.1, we obtain

$$\sum_{k=1}^{\infty} \Big(\underline{\varepsilon} D_h(x_{k+1}, x_k) \Big) \leq F^h_{\overline{L}}(x_0, x_{-1}) - \lim_{k\to\infty} F^h_{\overline{L}}(x_{k+1}, x_k) < \infty\,,$$

which implies (10.4.4). For $\underline{\varepsilon} > 0$, Assumption I(iii) together with (10.4.4) imply $x_{k+1} - x_k \to 0$ as $k \to \infty$. □

Analyzing the full set of limit points of the sequence generated by Model BPG is difficult, as illustrated in [139]. Obtaining the global convergence is still an open problem. Moreover, the work in [139] relies on convex model functions.

In order to simplify slightly the setting, we restrict the set of limit points to the set $\operatorname{int}\operatorname{dom} h$. Such a choice may appear to be restrictive, however, Model BPG when applied to many practical problems results in sequences that have this property as illustrated in Section 9.7.

To this regard, denote the following

$$\omega^{\operatorname{int}\operatorname{dom} h}(x_0) := \omega(x_0) \cap \operatorname{int}\operatorname{dom} h \quad \text{and} \quad \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0) := \omega_f(x_0) \cap \operatorname{int}\operatorname{dom} h\,.$$

The subset of $F_{\bar{L}}^h$-attentive (similar to f-attentive) limit points is

$$\omega_{F_{\bar{L}}^h}(x_0) := \left\{ (y,x) \in \mathbb{R}^N \times \mathbb{R}^N \mid \exists K \subset \mathbb{N} \colon (x_k, F_{\bar{L}}^h(x_k, x_{k-1})) \underset{K}{\to} (x, F_{\bar{L}}^h(y,x)) \right\}.$$

Also, we define $\omega_{F_{\bar{L}}^h}^{(\operatorname{int}\operatorname{dom} h)^2} := \omega_{F_{\bar{L}}^h} \cap (\operatorname{int}\operatorname{dom} h \times \operatorname{int}\operatorname{dom} h)$.

Proposition 9.5.4.3. *Let Assumptions F, G, I hold. Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model BPG. Then, the following holds:*

(*i*) $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$,

(*ii*) $x \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ *if and only if* $(x,x) \in \omega_{F_{\bar{L}}^h}^{(\operatorname{int}\operatorname{dom} h)^2}(x_0)$.

(*iii*) $F_{\bar{L}}^h$ *is constant and finite on* $\omega_{F_{\bar{L}}^h}^{(\operatorname{int}\operatorname{dom} h)^2}(x_0)$ *and* f *is constant and finite on* $\omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ *with same value.*

Proof. (*i*) We show the inclusion $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) \subset \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ and $\omega_f^{\operatorname{int}\operatorname{dom} h}(x_0) \subset \omega^{\operatorname{int}\operatorname{dom} h}(x_0)$ is clear by definition. Let $x^\star \in \omega^{\operatorname{int}\operatorname{dom} h}(x_0)$, then we obtain the following

$$\begin{aligned} f(x^\star) + \left(\underline{L} + \frac{1}{\tau_k}\right) D_h(x^\star, x_k) &\overset{(9.3.3)}{\geq} f(x^\star; x_k) + \frac{1}{\tau_k} D_h(x^\star, x_k) \overset{(9.3.4)}{\geq} f(x_{k+1}; x_k) + \frac{1}{\tau_k} D_h(x_{k+1}, x_k) \\ &\overset{(9.3.3)}{\geq} f(x_{k+1}) - \left(\bar{L} - \frac{1}{\tau_k}\right) D_h(x_{k+1}, x_k) \overset{\varepsilon_k > 0}{\geq} f(x_{k+1})\,. \end{aligned}$$

Obviously, by Assumption I(iii) combined with the fact that $x_k \underset{K}{\to} x^\star$, we have $D_h(x^\star, x_k) \to 0$ as $k \underset{K}{\to} \infty$, which, together with the lower semicontinuity of f, implies

$$f(x^\star) \geq \liminf_{k \underset{K}{\to} \infty} f(x_{k+1}) \geq f(x^\star)\,,$$

thus $x^\star \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$.

(*ii*) If $x \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$, then we have $x_k \underset{K}{\to} x$ for $K \subset \mathbb{N}$, and $f(x_k) \underset{K}{\to} f(x)$. As a consequence of Proposition 9.5.2.2 and Assumption I(iii), $D_h(x_{k+1}, x_k) \to 0$ as $k \to \infty$, which implies that $x_{k+1} \underset{K}{\to} x$. The first part of the proof implies $f(x_{k+1}) \underset{K}{\to} f(x)$. We also have $F_{\bar{L}}^h(x_{k+1}, x_k) \underset{K}{\to} f(x)$ which we prove below, which implies that $(x,x) \in \omega_{F_{\bar{L}}^h}^{\operatorname{int}\operatorname{dom} h}(x_0)$. Note that by definition of $F_{\bar{L}}^h$ we have the following

$$\begin{aligned} F_{\bar{L}}^h(x_{k+1}, x_k) &= f(x_{k+1}; x_k) + \bar{L} D_h(x_{k+1}, x_k)\,, \\ &= f(x_{k+1}) + (f(x_{k+1}; x_k) - f(x_{k+1})) + \bar{L} D_h(x_{k+1}, x_k)\,, \end{aligned}$$

and with the MAP property we have

$$f(x_{k+1}) \leq F^h_{\bar{L}}(x_{k+1}, x_k) \leq f(x_{k+1}) + (\bar{L} + \underline{L}) D_h(x_{k+1}, x_k)\,. \tag{9.5.14}$$

Thus, we have that $F^h_{\bar{L}}(x_{k+1}, x_k) \xrightarrow[K]{} f(x)$ as $D_h(x_{k+1}, x_k) \xrightarrow[K]{} 0$. Conversely, suppose $(x, x) \in \omega^{\operatorname{int}\operatorname{dom} h}_{F^h_{\bar{L}}}(x_0)$ and $x_k \xrightarrow[K]{} x$ for $K \subset \mathbb{N}$. This, together with $D_h(x_{k+1}, x_k) \to 0$ as $k \xrightarrow[K]{} \infty$, induces $F^h_{\bar{L}}(x_{k+1}, x_k) \xrightarrow[K]{} f(x)$, which further implies $f(x_{k+1}) \xrightarrow[K]{} f(x)$ due to the following. Note that we have

$$\begin{aligned} f(x_{k+1}) &= F^h_{\bar{L}}(x_{k+1}, x_k) + (f(x_{k+1}) - f(x_{k+1}; x_k)) + \bar{L} D_h(x_{k+1}, x_k) \\ &\geq F^h_{\bar{L}}(x_{k+1}, x_k) + (\bar{L} - \underline{L}) D_h(x_{k+1}, x_k)\,. \end{aligned}$$

Finally we have

$$F^h_{\bar{L}}(x_{k+1}, x_k) + (\bar{L} - \underline{L}) D_h(x_{k+1}, x_k) \leq f(x_{k+1}) \leq F^h_{\bar{L}}(x_{k+1}, x_k)\,.$$

Thus, with $D_h(x_{k+1}, x_k) \to 0$ as $k \xrightarrow[K]{} \infty$ and $F^h_{\bar{L}}(x_{k+1}, x_k) \xrightarrow[K]{} f(x)$, we deduce that $f(x_{k+1}) \xrightarrow[K]{} f(x)$. And therefore $x \in \omega^{\operatorname{int}\operatorname{dom} h}_f(x_0)$.

(iii) By Proposition 9.5.2.1, the sequence $(F^h_{\bar{L}}(x_{k+1}, x_k))_{k\in\mathbb{N}}$ converges to a finite value $\underline{F}$. Note that $D_h(x_{k+1}, x_k) \to 0$ as $k \xrightarrow[K]{} \infty$ due to Proposition 9.5.2.2 (ii), when combined with Assumption I(iii) implies that $\|x_{k+1} - x_k\| \to 0$. For $(x^\star, x^\star) \in \omega^{(\operatorname{int}\operatorname{dom} h)^2}_{F^h_{\bar{L}}}(x_0, x_0)$ there exists $K \subset \mathbb{N}$ such that $x_k \xrightarrow[K]{} x^\star$ and $F^h_{\bar{L}}(x_{k+1}, x_k) \xrightarrow[K]{} F^h_{\bar{L}}(x^\star, x^\star) = f(x^\star)$, i.e., the value of the limit point is independent of the choice of the subsequence. The result follows directly and by using (i). □

The following result summarizes that $F^h_{\bar{L}}$-attentive sequences converge to a stationary point.

Theorem 9.5.4.1 (Sub-sequential convergence to stationary points)**.** *Let Assumptions F, G, I hold. If the sequence $(x_k)_{k\in\mathbb{N}}$ is generated by Model BPG, then*

$$\omega^{(\operatorname{int}\operatorname{dom} h)^2}_{F^h_{\bar{L}}}(x_0) \subset \operatorname{crit}(F^h_{\bar{L}})\,. \tag{9.5.15}$$

Proof. From (9.5.6), we have $\|\partial F^h_{\bar{L}}(x_{k+1}, x_k)\|_- \leq C\|x_{k+1} - x_k\|$ for some constant $C > 0$. Using $\|x_{k+1} - x_k\| \to 0$, convergence of $(\tau_k)_{k\in\mathbb{N}}$, and Proposition 9.5.4.3(i) yields (10.4.5), by the closedness property of the limiting subdifferential. □

Discussion. Subsequential convergence to a stationary point was already considered in few works. In particular, the work in [60] already provides such a result, however, it relies on certain abstract assumptions. Even though such assumptions are valid for some practical algorithms, the authors do not consider a concrete algorithm. Moreover, their abstract update step depends on the minimization of the model function, which can require additional regularity conditions on the problem. For example, if the model function is linear, then the domain must be compact to guarantee the existence of a solution. A related line-search variant of Model BPG was considered in [139], for which subsequential convergence to a stationarity point was proven. The subsequential convergence results in [139] are more general than our work, as they analyse the behavior of limit points in $\operatorname{dom} h$, $\operatorname{cl}\operatorname{dom} h$, $\operatorname{int}\operatorname{dom} h$ (cf. [139, Theorem 22]). Our analysis is restricted to limit points in $\operatorname{int}\operatorname{dom} h$, as typically such an assumption holds in practice (see Section 9.7). Though subsequential convergence is satisfactory, proving global convergence is nontrivial, in general. It is not yet clear from our work, whether global convergence can be proven if the limit points lie on the boundary of $\operatorname{dom} h$. Both the above-mentioned works rely on function values to obtain a subsequential convergence result. We change this trend. In this chapter, we rely on Lyapunov function and obtain an even stronger result, that is global convergence of the sequence generated by Model BPG to a stationarity point.

9.5.5 Global convergence to a stationary point of the Lyapunov function

Assumption J. Let $\mathcal{O}$ be an o-minimal structure. The functions $\tilde{f} : \mathbb{R}^N \times \mathbb{R}^N \to \overline{\mathbb{R}}$, $(x, \bar{x}) \mapsto f(x; \bar{x})$ with $\operatorname{dom} \tilde{f} := \operatorname{dom} f \times \operatorname{dom} f$, and $\tilde{h} : \mathbb{R}^N \times \mathbb{R}^N \to \overline{\mathbb{R}}$, $(x, \bar{x}) \mapsto h(\bar{x}) + \langle \nabla h(\bar{x}), x - \bar{x} \rangle$ with $\operatorname{dom} \tilde{h} := \operatorname{dom} h \times \operatorname{int} \operatorname{dom} h$ are definable $\mathcal{O}$.

Lemma 9.5.5.1. *Let Assumptions F, G, I, J hold. Then, the Lyapunov function $F^h_{\overline{L}}$ is definable in $\mathcal{O}$, and satisfies KL property at any point of* $\operatorname{dom} \partial F^h_{\overline{L}}$.

Proof. As per the conditions of Lemma 3.7.0.2, we deduce that functions that are definable in an o-minimal structure are closed under addition and multiplication. With Assumption J, it is easy to deduce that the $F^h_{\overline{L}}$ is also definable in $\mathcal{O}$ using Lemma 3.7.0.2. Invoking Theorem 3.7.0.3, we deduce that $F^h_{\overline{L}}$ satisfies KL property at any point of $\operatorname{dom} \partial F^h_{\overline{L}}$. □

In the context of additive composite problems, the global convergence analysis of BPG based methods [28] (Chapter 5) relies on strong convexity of h. However, in our setting we relax such a requirement on h, via the following assumption. Note that imposing such an assumption (Assumption K) is weaker than imposing the strong convexity of h, as we only need the strong convexity property to hold over a compact convex set. Such a property can be satisfied even if h is not strongly convex, for example, Burg's entropy (see Section 9.7.3).

Assumption K. For any compact convex set $B \subset \operatorname{int} \operatorname{dom} h$, there exists $\sigma_B > 0$ such that h is σ_B-strongly convex over B, i.e., for any $x, y \in B$ the condition $D_h(x, y) \geq \frac{\sigma_B}{2} \|x - y\|^2$ holds.

Now, we present the global convergence result of the sequence generated by Model BPG.

Theorem 9.5.5.2 (Global convergence to a stationary point under KL property)**.** *Let Assumptions F, G, I, J, K hold. Let the sequence $(x_k)_{k \in \mathbb{N}}$ be generated by Model BPG (Algorithm 7) with $\tau_k \to \tau$ for certain $\tau > 0$ and the condition $\omega^{\operatorname{int} \operatorname{dom} h}(x_0) = \omega(x_0)$ holds true. Then, convergent subsequences are $F^h_{\overline{L}}$-attentive convergent, and*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty \qquad \textit{(finite length property)}.$$

Moreover, the sequence $(x_k)_{k \in \mathbb{N}}$ converges to x such that (x, x) is a critical point of $F^h_{\overline{L}}$.

Proof. Note that the sequence $(x_k)_{k \in \mathbb{N}}$ generated by Model BPG is a bounded sequence (see Remark 9.3.0.2). The proof relies on Theorem 9.4.0.1 provided in Section 9.4, for which we need to verify the conditions (i)–(v). Due to Lemma 9.5.5.1, $F^h_{\overline{L}}$ satisfies Kurdyka–Łojasiewicz property at each point of $\operatorname{dom} \partial F^h_{\overline{L}}$.

Note that as $\omega^{\operatorname{int} \operatorname{dom} h}(x_0) = \omega(x_0)$ holds true, there exists a sufficiently small $\varepsilon > 0$ such that $\tilde{B} := \{x : \operatorname{dist}(x, \omega(x_0)) \leq \varepsilon\} \subset \operatorname{int} \operatorname{dom} h$. As $\omega(x_0)$ is compact due to Proposition 9.5.4.1(i), the set $\tilde{B}$ is also compact. Moreover, the convex hull of the set $\tilde{B}$ denoted by $B := \operatorname{conv} \tilde{B}$ is also compact, as the convex hull of a compact set is also compact in finite dimensional setting. A simple calculation reveals that the set B lies in the set $\operatorname{int} \operatorname{dom} h$. Thus, due to Proposition 10.4.3.1 along with Proposition 9.5.4.1(ii), without loss of generality, we assume that the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model BPG lies in the set B. By definition of σ_B as per Assumption K we have

$$D_h(x_{k+1}, x_k) \geq \frac{\sigma_B}{2}\|x_{k+1} - x_k\|^2\,, \tag{9.5.16}$$

through which we obtain

$$F_{\bar{L}}^{h}(x_{k+1}, x_k) \leq F_{\bar{L}}^{h}(x_k, x_{k-1}) - \frac{\varepsilon_k \sigma_B}{2}\|x_{k+1} - x_k\|^2\,,$$

which is (i) with $d_k = \frac{\varepsilon_k\sigma_B}{2}\|x_{k+1} - x_k\|^2$ and $a_k = 1$. We also have existence of $w_{k+1} \in \partial F_{\bar{L}}^{h}(x_{k+1}, x_k)$ due to Lemma 9.5.3.1 such that for some $C > 0$ we have

$$\|\partial F_{\bar{L}}^{h}(x_{k+1}, x_k)\|_{-} \leq C\|x_{k+1} - x_k\|\,,$$

which is (ii) with $b = C$, since the coefficients for both Euclidean distances are bounded from above. The continuity condition (iii) is deduced from a converging subsequence, whose existence is guaranteed by boundedness of $(x_k)_{k\in\mathbb{N}}$, and Proposition 9.5.4.3 guarantees that such convergent subsequences are $F_{\bar{L}}^{h}$-attentive convergent. The distance condition (iv) holds trivially as $\varepsilon_k > 0$ and $\sigma_B > 0$. The parameter condition (v), holds because $b_n = 1$ in this setting, hence $(b_n)_{n\in\mathbb{N}} \notin \ell_1$ and also we have

$$\sup_{n\in\mathbb{N}} \frac{1}{b_n a_n} = 1 < \infty\,, \quad \inf_n a_n = 1 > 0\,.$$

Theorem 9.4.0.1 implies the finite length property from which we deduce that the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model BPG converges to a single point, which we denote by x. As $(x_{k+1})_{k\in\mathbb{N}}$ also converges to x, the sequence $((x_{k+1}, x_k))_{k\in\mathbb{N}}$ converges to (x, x), which is a critical point of $F_{\bar{L}}^{h}$ due to Theorem 9.5.4.1. □

9.5.6 Global convergence to a stationary point of the objective function

The global convergence result in Theorem 9.5.5.2 shows that Model BPG converges to a point, which in turn can be used to represent a critical point of the Lyapunov function. However, our goal is to find a critical point of the objective function f. We now establish the connection between a critical point of the Lyapunov function and a critical point of the objective function. Such a connection can later be exploited to conclude that the sequence generated by Model BPG converges to a critical point of f.

Firstly, we need the following result, which establishes the connection between fixed points of the update mapping and critical points of f.

Lemma 9.5.6.1. *Let Assumptions F, G hold. For any $0 < \tau < (1/\bar{L})$ and $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, the fixed points of the update mapping $T_\tau(\bar{x})$ are critical points of f.*

Proof. Let $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$ be a fixed point of T_τ, in the sense the condition $\bar{x} \in T_\tau(\bar{x})$ holds true. By definition of $T_\tau(\bar{x})$, the following condition holds true:

$$0 \in \partial f(x; \bar{x}) + \frac{1}{\tau}\left(\nabla h(x) - \nabla h(\bar{x})\right)$$

at $x = \bar{x}$, which implies that $0 \in \partial f(\bar{x}; \bar{x})$. As a consequence of Lemma F.2.0.1, we have $\partial f(\bar{x}; \bar{x}) \subset \partial f(\bar{x})$, thus $\bar{x}$ is a critical point of the function f. □

We also require the following technical result.

Lemma 9.5.6.2 (Continuity property)**.** *Let Assumptions F, G, I hold. Let the sequence* $(x_k)_{k\in\mathbb{N}}$ *be bounded such that* $x_k \to \bar{x}$*, where* $x_k \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$ *for all* $k \in \mathbb{N}$*, and* $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$*. Let* $\tau_k \to \tau$*, such that* $0 < \underline{\tau} \leq \tau_k \leq \bar{\tau} < 1/\bar{L}$*. Let there exist a bounded set* $B \subset \operatorname{int}\operatorname{dom} h$*, such that* $T_{\tau_k}(x_k) \subset B$*,* $x_k \in B$ *for all* $k \in \mathbb{N}$*. If* $\limsup_{k\to\infty} T_{\tau_k}(x_k) \subset \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$*, then* $\limsup_{k\to\infty} T_{\tau_k}(x_k) \subset T_\tau(\bar{x})$*.*

Proof. Consider any sequence $(y_k)_{k\in\mathbb{N}}$ such that for any $k \in \mathbb{N}$, the condition $y_k \in T_{\tau_k}(x_k)$ holds true. Recall that $f(x; y)$ is continuous on its domain due to Assumption G(iv). By optimality of $y_k \in T_{\tau_k}(x_k)$, for any $z \in \mathbb{R}^N$ we have the following:

$$f(y_k; x_k) + \frac{1}{\tau_k} D_h(y_k, x_k) \leq f(z; x_k) + \frac{1}{\tau_k} D_h(z, x_k)\,. \tag{9.5.17}$$

As a consequence of boundedness of the sequence $(y_k)_{k\in\mathbb{N}}$, by Bolzano–Weierstrass Theorem there exists a convergent subsequence. Let $y_k \underset{K}{\to} \pi$ such that $\pi \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$. Note that $\tau_k \underset{K}{\to} \tau$ for some $K \subset \mathbb{N}$. Applying limit on both sides of (9.5.17) using the continuity of the model function and the Bregman distance gives

$$f(\pi; \bar{x}) + \frac{1}{\tau} D_h(\pi, \bar{x}) \leq f(z; \bar{x}) + \frac{1}{\tau} D_h(z, \bar{x})\,, \quad \forall\, z \in \operatorname{dom} f \cap \operatorname{dom} h\,, \tag{9.5.18}$$

which implies that π minimizes the function $f(\,\cdot\,; \bar{x}) + \frac{1}{\tau} D_h(\,\cdot\,, \bar{x})$. This implies that $\pi \in T_\tau(\bar{x})$ and the result follows. □

The following result establishes the fact the sequence generated by Model BPG indeed converges to a critical point of the objective function.

Theorem 9.5.6.3 (Global convergence to a stationary point of the objective function)**.** *Under the conditions of Theorem 9.5.5.2, the sequence generated by Model BPG converges to a critical point of* f*.*

Proof. The sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model BPG under the assumptions as in Theorem 9.5.5.2 is globally convergent, thus let $x_k \to x$ and also $x_{k+1} \to x$. As $x_{k+1} \in T_{\tau_k}(x_k)$ and τ_k converges to τ, with Lemma 9.5.6.2 we deduce that $x \in T_\tau(x)$. Additionally, with the result in Lemma 9.5.6.2, we deduce that x is the fixed point of the mapping $T_\tau(x)$, i.e., $x \in T_\tau(x)$. Then, using Lemma 9.5.6.1 we conclude that x is a critical point of the function f. □

9.5.7 Convergence rates

It is possible to deduce convergence rates for a certain class of desingularizing functions. Based on [6, 26, 69], we provide the following result, which provides the convergence rates for the sequence generated by Model BPG.

Theorem 9.5.7.1 (Convergence rates)**.** *Under the conditions of Theorem 9.5.5.2, let the sequence* $(x_k)_{k\in\mathbb{N}}$ *generated by Model BPG converge to* $x \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$*, and let the Lyapunov function* $F^h_{\bar{L}}$ *satisfy Kurdyka–Łojasiewicz property with the following desingularizing function:*

$$\varphi(s) = cs^{1-\theta}\,,$$

for certain $c > 0$ *and* $\theta \in [0, 1)$*. Then, we have the following:*

- *If* $\theta = 0$*, then* $(x_k)_{k\in\mathbb{N}}$ *converges in finite number of steps.*

- *If $\theta \in (0, \frac{1}{2}]$, then there exists $\rho \in [0, 1)$ and $G > 0$ such that for all $k \geq 0$ we have*

$$\|x_k - x\| \leq G\rho^k\,.$$

- *If $\theta \in (\frac{1}{2}, 1)$, then there exists $G > 0$ such that for all $k \geq 0$ we have*

$$\|x_k - x\| \leq Gk^{-\frac{1-\theta}{2\theta-1}}\,.$$

Proof. Here, we consider the same notions as in the proof of Theorem 9.5.5.2. First, using the convexity of the function $-s^{1-\theta}$ we obtain

$$\begin{aligned}
&(F_{\tilde{L}}^{h}(x_k, x_{k-1}) - v(\mathcal{P}_M))^{1-\theta} - (F_{\tilde{L}}^{h}(x_{k+1}, x_k) - v(\mathcal{P}_M))^{1-\theta} \\
&\geq (1-\theta)(F_{\tilde{L}}^{h}(x_k, x_{k-1}) - v(\mathcal{P}_M))^{-\theta}(F_{\tilde{L}}^{h}(x_k, x_{k-1}) - F_{\tilde{L}}^{h}(x_{k+1}, x_k))\,, \\
&\geq (1-\theta)(F_{\tilde{L}}^{h}(x_k, x_{k-1}) - v(\mathcal{P}_M))^{-\theta}\frac{\varepsilon_k\sigma_B}{2}\|x_{k+1} - x_k\|^2\,, \\
&\geq (1-\theta)(F_{\tilde{L}}^{h}(x_k, x_{k-1}) - v(\mathcal{P}_M))^{-\theta}\frac{\underline{\varepsilon}\sigma_B}{2}\|x_{k+1} - x_k\|^2\,,
\end{aligned}$$

where in the second inequality we used the Proposition 9.5.2.1 along with the definition of σ_B, and in the last step we used $\varepsilon_k \geq \underline{\varepsilon}$. Denote $U := \omega_{F_{\tilde{L}}^{h}}^{(\operatorname{int} \operatorname{dom} h)^2}(x_0)$, and thanks to Theorem 9.5.4.1 we have $U \subset \operatorname{crit}(F_{\tilde{L}}^{h})$. Due to Proposition 9.5.4.1, we already know that U is a connected compact set and

$$\lim_{k\to\infty} \operatorname{dist}((x_{k+1}, x_k), U) = 0\,.$$

Continuing the calculation, following the proof technique of [26, Theorem 3.1], using Lemma 3.7.0.1 with $\Omega = U$, we deduce that there exists $l \in \mathbb{N}$, $C_1 > 0$ such that for any $k > l$, the following holds:

$$\sum_{i=l+1}^{k} \|x_{i+1} - x_i\| \leq \|x_{l+1} - x_l\| + C_1(F_{\tilde{L}}^{h}(x_{l+1}, x_l) - v(\mathcal{P}_M))^{1-\theta}\,.$$

Denote $\Delta_l := \sum_{i=l}^{\infty} \|x_{i+1} - x_i\|$. On application of Lemma 3.7.0.1 with $\Omega = U$, and Lemma 9.5.3.1, we deduce that there exists $C_2 > 0$ such that

$$\Delta_{l+1} \leq \Delta_l - \Delta_{l+1} + C_2(\Delta_l - \Delta_{l+1})^{\frac{1-\theta}{\theta}}\,.$$

The rest of the proof is only a slight modification to the proof of [6, Theorem 5]. □

9.6 Examples

In this section we consider special instances of $(\mathcal{P}_M)$, namely, additive composite problems and a broad class of composite problems. The goal is to quantify assumptions for these problems such that the global convergence result (Theorem 9.5.6.3) of Model BPG is applicable. To this regard, we only consider the functions that satisfy Assumption F. Typically, function is made up of function components and these components govern the function behavior. Thus, it is beneficial to introduce properties on the components of f, for which certain plausible conditions will enable the applicability of Model BPG. In this section, henceforth we enforce the following blanket assumptions.

(B1) The function h is a Legendre function that is $\mathcal{C}^2$ over $\operatorname{int}\operatorname{dom} h$. For any compact convex set $B \subset \operatorname{int}\operatorname{dom} h$, there exists $\sigma_B > 0$ such that h is σ_B-strongly convex over B. Also, h has bounded second derivative on any bounded subset $B_1 \subset \operatorname{int}\operatorname{dom} h$. Moreover, for bounded $(u_k)_{k\in\mathbb{N}}$, $(v_k)_{k\in\mathbb{N}}$ in $\operatorname{int}\operatorname{dom} h$, the following holds as $k \to \infty$:

$$D_h(u_k, v_k) \to 0 \iff \|u_k - v_k\| \to 0\,.$$

(B2) The function f is coercive and additionally the conditions $\operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h \neq \emptyset$, $\operatorname{crit} f \cap \operatorname{int}\operatorname{dom} h \neq \emptyset$, $\operatorname{dom} f \subset \operatorname{cl}\operatorname{dom} h$ hold true.

(B3) The functions $\tilde{f} : \mathbb{R}^N \times \mathbb{R}^N \to \overline{\mathbb{R}}$, $(x, \bar{x}) \mapsto f(x; \bar{x})$ with $\operatorname{dom} \tilde{f} := \operatorname{dom} f \times \operatorname{dom} f$, and $\tilde{h} : \mathbb{R}^N \times \mathbb{R}^N \to \overline{\mathbb{R}}$, $(x, \bar{x}) \mapsto h(\bar{x}) + \langle \nabla h(\bar{x}), x - \bar{x}\rangle$ with $\operatorname{dom} \tilde{h} := \operatorname{dom} h \times \operatorname{int}\operatorname{dom} h$ are definable in an o-minimal structure $\mathcal{O}$.

Note that h satisfies Assumption (B1) which considers the same conditions on h as in Assumptions G, I, K. The function satisfies Assumption (B2), which is a consolidation of function specific assumptions in Assumptions F, G. Clearly, Assumption (B3) implies Assumption J.

9.6.1 Additive composite problems

We consider the following non-convex additive composite problem:

$$\inf_{x\in\mathbb{R}^N} f(x)\,, \quad f(x) := f_0(x) + f_1(x)\,, \tag{9.6.1}$$

which is a special case of $(\mathcal{P}_M)$. Additive composite problems arise in several applications, such as standard phase retrieval [28], low rank matrix factorization (Chapter 6), deep linear neural networks (Chapter 7), and many more. We impose the following conditions that are common in the analysis of Forward–Backward algorithms [137], which are used to optimize additive composite problems.

(C1) $f_0 : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper, lsc function and is regular at any $x \in \operatorname{dom} f_0$. Also, the following qualification condition holds true:

$$\partial^\infty f_0(x) \cap (-N_{\operatorname{dom} h}(x)) = \{0\}\,, \quad \forall\, x \in \operatorname{dom} f_0 \cap \operatorname{dom} h\,. \tag{9.6.2}$$

(C2) $f_1 : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper, lsc function and is $\mathcal{C}^2$ on an open set that contains $\operatorname{dom} f_0$. Also, there exist $\bar{L}, \underline{L} > 0$ such that for any $\bar{x} \in \operatorname{dom} f_0 \cap \operatorname{int}\operatorname{dom} h$, the following condition holds true:

$$-\underline{L} D_h(x, \bar{x}) \le f_1(x) - f_1(\bar{x}) - \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle \le \bar{L} D_h(x, \bar{x})\,, \quad \forall\, x \in \operatorname{dom} f_0 \cap \operatorname{dom} h\,. \tag{9.6.3}$$

Note that with Assumption (C1), (C2) it is easy to deduce that $\operatorname{dom} f_0 = \operatorname{dom} f$. For $\bar{x} \in \operatorname{dom} f$, the model function $f(\,\cdot\,; \bar{x}) : \mathbb{R}^N \to \overline{\mathbb{R}}$ which, when evaluated at $x \in \operatorname{dom} f$ gives

$$f(x; \bar{x}) := f_0(x) + f_1(\bar{x}) + \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle\ . \tag{9.6.4}$$

Using the model function in (9.6.4) and the condition (9.6.3), we deduce that there exist $\underline{L}, \bar{L} > 0$ such that for any $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, MAP property is satisfied at $\bar{x}$ with $\underline{L}, \bar{L}$ as the following holds true:

$$-\underline{L} D_h(x, \bar{x}) \le f(x) - f(x; \bar{x}) \le \bar{L} D_h(x, \bar{x})\,, \quad \forall\, x \in \operatorname{dom} f \cap \operatorname{dom} h\,, \tag{9.6.5}$$

as $f(x) - f(x;\bar{x}) := f_1(x) - f_1(\bar{x}) - \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle$, thus satisfying Assumption G(i). The condition in (9.6.5) is similar to the popular L-smad property in [28]. The main addition is that $x \in \operatorname{dom} f \cap \operatorname{dom} h$ and $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, whereas the L-smad property requires $x, \bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$. We illustrate this below.

Remark. Consider $f_1(x) := \frac{1}{2}x^2$, $f_0(x) := \delta_{[0,\infty)}(x)$ and $h(x) = x\log(x)$ with $\operatorname{dom} h = [0,\infty)$ under $0\log(0) = 0$. Clearly, $\operatorname{dom} h \subset \operatorname{dom} f_1$ and $\operatorname{dom} f \subset \operatorname{dom} h$ hold true. The function f_1 is differentiable at $x = 0$, and condition in (9.6.3) holds true for $x = 0$. This scenario is not considered in the L-smad property (see [28, Lemma 2.1]).

We present below Model BPG algorithm that is applicable for additive composite problems. Using the model function in (9.6.4) in Model BPG we recover the BPG algorithm from [28].

BPG is Model BPG (Algorithm 7) with

$$f(x;x_k) := f_0(x) + f_1(x_k) + \langle \nabla f_1(x_k), x - x_k\rangle \ . \tag{9.6.6}$$

For $h(x) = \frac{1}{2}\|x\|^2$, Model BPG is equivalent to Proximal Gradient Method. Assumptions (C1), (C2) along with (B2) imply proper, lsc property of f and lower-boundedness of f, thus satisfying Assumption F. Considering (C1) we deduce that $f_0(x)$ is regular at $x \in \operatorname{dom} f_0$. Using [150, Proposition 10.5] we note that $f_0(x)$ is regular at all $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$. Let $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$, using [150, Proposition 10.5] on f_0, we obtain the following result:

$$\partial_{(x,y)} f_0(x) = (\partial_x f_0(x), 0)\,, \quad \partial^{\infty}_{(x,y)} f_0(x) = (\partial^{\infty}_x f_0(x), 0)\,. \tag{9.6.7}$$

Let $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$, we consider the following entity:

$$\partial f(x;\bar{x}) = \partial(f_0(x) + f_1(\bar{x}) + \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle)\,,$$

and in order for the summation rule of subdifferential ([150, Corollary 10.9]) to be applicable at $(x,\bar{x})$, we need finiteness of $f_0(x)$ and continuously differentiability of $\tilde{f}_1(x,\bar{x}) := f_1(\bar{x}) + \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle$ (also see [150, Exercise 8.8]). Clearly, f_0 is finite at $(x,\bar{x})$, and $\tilde{f}_1$ is finite and also continuously differentiable around $(x,\bar{x})$ due to Assumption (C2). Thus, using (9.6.7) and [150, Corollary 10.9] we obtain the following conditions:

$$\partial f(x;\bar{x}) = (\partial_x f_0(x) + \nabla f_1(\bar{x}), \nabla^2 f_1(\bar{x})(x - \bar{x}))\,, \quad \partial^{\infty} f(x;\bar{x}) = (\partial^{\infty}_x f_0(x), 0)\,, \tag{9.6.8}$$

and as a result (Assumption G(iii)') is satisfied. Using the condition (9.6.2) and (9.6.8), we deduce that Assumption G(ii) is satisfied. Now, we verify Assumption I(i). Consider a bounded subset S in $\operatorname{dom} f$. For fixed $x \in \operatorname{dom} f$, and for all $\bar{x} \in S$ we have

$$\partial_{\bar{x}} f(x;\bar{x}) = \{\nabla_{\bar{x}}(f(x;\bar{x}))\} = \{\nabla^2 f_1(\bar{x})(x - \bar{x})\}. \tag{9.6.9}$$

Note that ∇f_1 is Lipschitz continuous on any bounded subset of $\operatorname{dom} f$, as f_1 is $\mathcal{C}^2$ on $\operatorname{dom} f$. This implies that the Hessian is bounded on bounded sets of $\operatorname{dom} f$. Thus, based on the same notions in (9.6.9), we deduce that there exists a constant $M > 0$ such that

$$\|\nabla_{\bar{x}}(f(x;\bar{x}))\| \le M\|x - \bar{x}\|\,,$$

holds true, thus verifying Assumption I(i). As a simple consequence of Assumption (C1), (C2) the condition Assumption G(iv) is satisfied.

As discussed above, Assumptions (C1), (C2), (B1), (B2), (B3) imply Assumptions F, G, I, J, K. Thus, as a consequence of Theorem 9.5.5.2, 9.5.6.3 we obtain the following result which provides the global convergence of the sequence generated by BPG to a stationary point.

Theorem 9.6.1.1 (Global convergence of BPG sequence)**.** *Let Assumptions (C1), (C2), (B1), (B2), (B3) hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by BPG and the condition $\omega^{\operatorname{int}\operatorname{dom}h}(x_0) = \omega(x_0)$ holds true. Let $\tau_k \to \tau$ for certain $\tau > 0$. Then, the sequence $(x_k)_{k\in\mathbb{N}}$ has finite length, that is*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty\,,$$

and the sequence $(x_k)_{k\in\mathbb{N}}$ converges to x, which is a critical point of f.

9.6.2 Composite problems

We consider the following non-convex composite problem:

$$\inf_{x\in\mathbb{R}^N} f(x)\,, \quad f(x) := f_0(x) + g(F(x))\,, \tag{9.6.10}$$

which is a special case of the problem $(\mathcal{P}_M)$. Composite problems arise in robust phase retrieval, robust PCA, censored $\mathbb{Z}_2$ synchronization [59, 61, 62, 102, 125]. We require the following conditions.

(D1) $f_0 : \mathbb{R}^N \to \overline{\mathbb{R}}$ is a proper, lsc function and is regular at any $x \in \operatorname{dom} f_0$. Also, the following qualification condition holds true:

$$\partial^\infty f_0(x) \cap (-N_{\operatorname{dom}h}(x)) = \{0\}\,, \quad \forall\, x \in \operatorname{dom} f_0 \cap \operatorname{dom} h\,. \tag{9.6.11}$$

(D2) $g : \mathbb{R}^M \to \mathbb{R}$ is a Q-Lipschitz continuous function and a regular function. Also, there exists $P > 0$ such that at any $x \in \mathbb{R}^M$, the following condition holds true:

$$\sup_{v\in\partial g(x)} \|v\| \le P\,. \tag{9.6.12}$$

(D3) $F : \mathbb{R}^N \to \mathbb{R}^M$ is $\mathcal{C}^2$ over $\mathbb{R}^N$. Also, there exist $L > 0$ such that for any $\bar{x} \in \operatorname{dom} f_0 \cap \operatorname{int}\operatorname{dom} h$, the following condition holds true:

$$\|F(x) - F(\bar{x}) - \nabla F(\bar{x})(x - \bar{x})\| \le LD_h(x, \bar{x})\,, \quad \forall\, x \in \operatorname{dom} f_0 \cap \operatorname{dom} h\,,$$

where $\nabla F(\bar{x})$ is the Jacobian of F at $\bar{x}$.

Note that when $M = 1$, $g(x) = x$, the problem in (9.6.10) is a special case of (9.6.1). However, for a generic g satisfying (D2), the problem in (9.6.10) cannot be captured under the additive composite problem setting given in Section 9.6.1. Thus, in this section we consider a separate analysis for generic composite problems in (9.6.10).

The properties (D1), (D2), (D3) along with (B2) imply proper, lsc property and lower-boundedness of f, thus satisfying Assumption F. Note that with Assumption (D1), (D2), (D3) it is easy to deduce that

$\operatorname{dom} f_0 = \operatorname{dom} f$. Let $\bar{x} \in \operatorname{dom} f$ and we consider the following model function which, when evaluated at $x \in \operatorname{dom} f$ gives:

$$f(x;\bar{x}) = f_0(x) + g(F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x}))\,. \tag{9.6.13}$$

Using (D2), (D3) we deduce that there exists $\bar{L} := LQ > 0$ such that for any $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, the following MAP property holds at $\bar{x}$ with $\bar{L}$:

$$|f(x) - f(x;\bar{x})| = |g(F(x)) - g(F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x}))| \leq \bar{L} D_h(x,\bar{x})\,,$$

for all $x \in \operatorname{dom} f \cap \operatorname{dom} h$, as g is Q-Lipschitz continuous and (D3) holds true. Thus, Assumption G(i) is satisfied with $\bar{L} = \underline{L} = LQ$. Before we verify other assumptions, we present Prox-Linear BPG, a specialization of Model BPG that is applicable to composite problems.

Prox-Linear BPG is Model BPG (Algorithm 7) with

$$f(x;x_k) := f_0(x) + g(F(x_k) + \nabla F(x_k)(x-x_k))\,. \tag{9.6.14}$$

For $h(x) = \frac{1}{2}\|x\|^2$, Prox-Linear BPG is related to Prox-Linear method [61, 102]. Considering (D1), we deduce that $f_0(x)$ is regular at $x \in \operatorname{dom} f_0$. Using [150, Proposition 10.5] we note that $f_0(x)$ is regular at all $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$. Using [150, Theorem 10.6] and (D2) we deduce that $g(F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x}))$ is regular for all $(x,\bar{x}) \in \mathbb{R}^N \times \mathbb{R}^N$. Furthemore, as a consequence of [150, Corollary 10.9], the function $f(x;\bar{x})$ is regular at $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$.

Using [150, Proposition 10.5] on f_0 we deduce that for all $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$, the following conditions hold true:

$$\partial_{(x,\bar{x})} f_0(x) = (\partial_x f_0(x), 0)\,, \quad \partial^\infty_{(x,\bar{x})} f_0(x) = (\partial^\infty_x f_0(x), 0)\,. \tag{9.6.15}$$

For this section, henceforth, we set $(x,\bar{x}) \in \operatorname{dom} f \times \operatorname{dom} f$ and denote $F(x;\bar{x}) := F(\bar{x}) + \nabla F(\bar{x})(x-\bar{x})$. Note that as $\partial^\infty_{F(x;\bar{x})} g(F(x;\bar{x})) = \{0\}$ due to (D1) and [150, Theorem 9.13], we deduce that the only y such that

$$y \in \partial^\infty_{F(x;\bar{x})} g(F(x;\bar{x})) \text{ with } (\nabla F(\bar{x})^* y, (\nabla F(\bar{x}) + \nabla_{\bar{x}}(\nabla F(\bar{x})(x-\bar{x})))^* y) = (0,0) \text{ is } y = 0\,, \tag{9.6.16}$$

where $\nabla F(\bar{x})^*$ denotes the adjoint of $\nabla F(\bar{x})$, and $\nabla_{\bar{x}}(\nabla F(\bar{x})(x-\bar{x}))$ denotes the Jacobian of the mapping $\nabla F(\bar{x})(x-\bar{x})$ at $\bar{x}$ with fixed x. Due to (D3), regularity of g and (9.6.16) we have

$$\partial g(F(x;\bar{x})) = \left(\nabla F(\bar{x})^* \partial_{F(x;\bar{x})} g(F(x;\bar{x})), (\nabla F(\bar{x}) + \nabla_{\bar{x}}(\nabla F(\bar{x})(x-\bar{x})))^* \partial_{F(x;\bar{x})} g(F(x;\bar{x}))\right)\,.$$

A similar statement also holds for $\partial^\infty g(F(x;\bar{x}))$ which on using $\partial^\infty_{F(x;\bar{x})} g(F(x;\bar{x})) = \{0\}$ due to [150, Theorem 9.13] results in $\partial^\infty g(F(x;\bar{x})) = \{0,0\}$. This further implies that the following qualification condition holds true:

$$\partial^\infty_{(x,\bar{x})} f_0(x) \cap (-\partial^\infty g(F(x;\bar{x})))) = \{(0,0)\}\,. \tag{9.6.17}$$

Using the qualification condition (9.6.17) along with [150, Corollary 10.9], we obtain the following:

$$\partial f(x,\bar{x}) = \partial_{(x,\bar{x})} f_0(x) + \partial g(F(x;\bar{x})))\,, \quad \partial^\infty f(x;\bar{x}) = (\partial^\infty_x f_0(x), 0)\,. \tag{9.6.18}$$

Thus, (Assumption G(iii)') is satisfied. Additionally, using the condition (9.6.11) in (D1), we deduce that Assumption G(ii) is satisfied. Now, we verify Assumption I(i). Let's consider a bounded subset S in $\operatorname{dom} f$.

For $\bar{x} \in \operatorname{dom} f$, there exists a constant $M_S > 0$ (dependent on S) such that for all $w \in \partial_{\bar{x}} f(x;\bar{x}) := (\nabla F(\bar{x}) + \nabla_{\bar{x}}(\nabla F(\bar{x})(x-\bar{x})))^* \partial_{F(x;\bar{x})} g(F(x;\bar{x}))$ the following condition holds true:

$$\|w\| \leq M_S \|x - \bar{x}\| \,, \quad \forall\, x \in S \,, \tag{9.6.19}$$

where we have used the boundedness of second order derivatives of components of F over S, as F is a twice continuously differentiable mapping, and boundedness of subgradients of g as per (9.6.12). As a simple consequence of Assumption (D1), (D2), (D3) the condition Assumption G(iv) is satisfied.

As discussed above, Assumptions (D1), (D2), (D3), (B1), (B2), (B3) imply Assumptions F, G, I, J, K. Thus, as a consequence of Theorem 9.5.5.2, 9.5.6.3 we obtain the following result which provides the global convergence of the sequence generated by Prox-Linear BPG to a stationary point.

Theorem 9.6.2.1 (Global convergence of Prox-Linear BPG sequence)**.** *Let Assumptions (D1), (D2), (D3), (B1), (B2), (B3) hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by Prox-Linear BPG and the condition $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds true. Let $\tau_k \to \tau$ for certain $\tau > 0$. Then, the sequence $(x_k)_{k\in\mathbb{N}}$ has finite length, that is*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty \,,$$

and the sequence $(x_k)_{k\in\mathbb{N}}$ converges to x, which is a critical point of f.

9.7 Experiments

For the purpose of empirical evaluation we consider many practical problems, namely, standard phase retrieval problems, robust phase retrieval problems and Poisson linear inverse problems. We compare our algorithms with inexact Bregman proximal minimization line search (IBPM-LS) [138], which is a popular algorithm to solve generic non-smooth non-convex problems. Before we provide the empirical results, we comment below on a variant of Model BPG based on the backtracking technique, which we used in the experiments.

Model BPG with backtracking. It is possible that the value of $\bar{L}$ in the MAP property is unknown. This issue can be solved by using a backtracking technique, where in each iteration a local constant $\bar{L}_k$ is found such that the following condition holds:

$$f(x_{k+1}) \leq f(x_{k+1}; x_k) + \bar{L}_k D_h(x_{k+1}, x_k) \,. \tag{9.7.1}$$

The value of $\bar{L}_k$ is found by taking an initial guess $\bar{L}_k^0$. If the condition (B.8.2) fails to hold, then with a scaling parameter $\nu > 1$, we set $\bar{L}_k$ to the smallest value in the set $\{\nu \bar{L}_k^0, \nu^2 \bar{L}_k^0, \nu^3 \bar{L}_k^0, \ldots\}$ such that (B.8.2) holds true. Enforcing $\bar{L}_k \geq \bar{L}_k m$ for $k \geq 1$ ensures that after finite number of iterations there is no change in the value of $\bar{L}_k$, which takes us to the situation that we analyzed in the chapter. The condition $\bar{L}_k \geq \bar{L}_k m$ can be enforced by choosing $\bar{L}_k^0 = \bar{L}_k m$.

Code. The code is open sourced at the following link: `https://github.com/mmahesh/composite-optimization-code` It contains the implementation of the algorithms, the random synthetic datasets generation process, the choices for hyper-parameters, the plots generation process and all the other related details.

9.7.1 Standard phase retrieval

The phase retrieval problem involves approximately solving a system of quadratic equations. Let $b_i \in \mathbb{R}$ and $A_i \in \mathbb{R}^{N\times N}$ be a symmetric positive semi-definite matrix, for all $i = 1, \ldots, M$. The goal of standard phase retrieval problem is to find $x \in \mathbb{R}^N$ such that the following system of quadratic equations is satisfied:

$$x^T A_i x \approx b_i, \qquad \text{for } i = 1, \ldots, M. \tag{9.7.2}$$

In standard terminology, b_i's are measurements and A_i's are so-called sampling matrices. In the context of Bregman proximal algorithms, regarding the phase retrieval problem, we refer the reader to [28] and Chapter 5. Further references regarding the phase retrieval problem include [40, 110, 164]. The standard technique to solve such system of quadratic equations is to solve the following optimization problem:

$$\min_{x\in\mathbb{R}^N} \mathcal{P}_0(x)\,, \quad \mathcal{P}_0(x) := \frac{1}{M}\sum_{i=1}^{M}(x^T A_i x - b_i)^2 + \mathcal{R}(x)\,, \tag{9.7.3}$$

where $\mathcal{R}(x)$ is the regularization term. We consider here L1 regularization with $\mathcal{R}(x) = \lambda\|x\|_1$ and squared L2 regularization with $\mathcal{R}(x) = \frac{\lambda}{2}\|x\|^2$, with some $\lambda > 0$. We consider two model functions in order to solve the problem in (9.7.3).

Model 1. Here, the analysis falls under the category of additive composite problems given in Section 9.6.1, where we set the following:

$$f_0(x) := \mathcal{R}(x)\,, \text{ and } \quad f_1(x) := \frac{1}{M}\sum_{i=1}^{M}(x^T A_i x - b_i)^2\,.$$

We consider the standard model for additive composite problems from [28], where around $y \in \mathbb{R}^N$, the model function $\mathcal{P}_0(\,\cdot\,; y) : \mathbb{R}^N \to \mathbb{R}$ at $x \in \mathbb{R}^N$ is given by

$$\mathcal{P}_0(x; y) := \frac{1}{M}\sum_{i=1}^{M}\left((y^T A_i y - b_i)^2 + (y^T A_i y - b_i)\left\langle 2A_i y, x - y\right\rangle\right) + \mathcal{R}(x)\,. \tag{9.7.4}$$

Consider the following Legendre function:

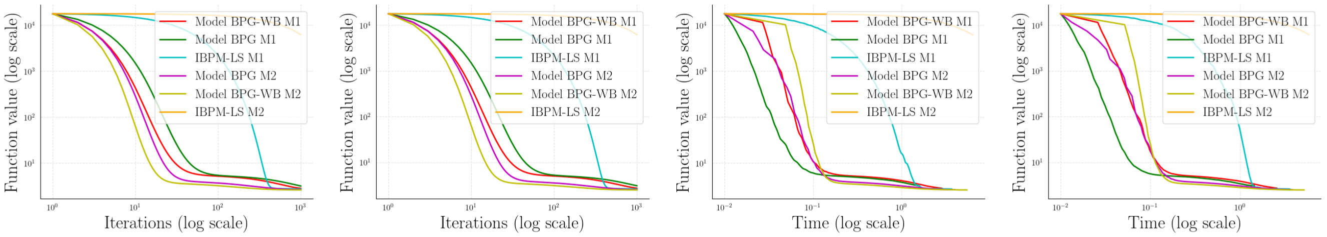
$$h(x) = \frac{1}{4}\|x\|^4 + \frac{1}{2}\|x\|^2\,.$$

Then, due to [28, Lemma 5.1] the following L-smad property or the MAP property is satisfied :

$$|\mathcal{P}_0(x) - \mathcal{P}_0(x; y)| \leq L_0 D_h(x, y)\,, \text{ for all } x, y \in \mathbb{R}^N, \tag{9.7.5}$$

where $L_0 \geq \sum_{i=1}^{M}(3\|A_i\|_F^2 + \|A_i\|_F\,|b_i|)$. In this setting, Model BPG subproblems have closed form solutions (see [28] and Chapter 5).

Model 2. The importance of finding better models suited to a particular problem was emphasized in [5]. The above provided model function in (9.7.4) is satisfactory, however, we would like take advantage of the structure of the function (9.7.3). Taking inspiration from [5], a simple observation that the objective is



(A) L1 reg (B) Squared L2 reg (C) L1 reg (D) Squared L2 reg

FIGURE 9.1: In this experiment we compare the performance of Model BPG, Model BPG with backtracking (denoted as Model BPG-WB), and IBPM-LS [138] on standard phase retrieval problems, with both L1 and squared L2 regularization. For this purpose, we consider M1 model function as in (9.7.4) without absolute sign (which is the same setting as [28]), and with M2 model function as in (9.7.6). Model BPG with M2 (9.7.6) is faster in both the settings and Model BPG variants perform significantly better than IBPM-LS. By reg, we mean regularization.

nonnegative can be exploited to create a new model function. We incorporate such a behavior in our second model function provided below. We use the Prox-Linear setting described in Section 9.6.2, where for any $x \in \mathbb{R}^N$ we set the following:

$$f_0(x) := \mathcal{R}(x)\,,$$

$$(F(x))_i = (x^T A_i x - b_i)^2\,, \text{ for all } i = 1, \ldots, M\,,$$

and for any $\tilde{y} \in \mathbb{R}^M$ we set

$$g(\tilde{y}) := \frac{1}{M}\|\tilde{y}\|_1\,, \text{ for } \tilde{y} \in \mathbb{R}^M\,.$$

Based on the model function (9.6.13), for fixed $y \in \mathbb{R}^N$, we consider the model function $\mathcal{P}_1(\,\cdot\,; y) : \mathbb{R}^N \to \mathbb{R}$ which, when evaluated at $x \in \mathbb{R}^N$ gives

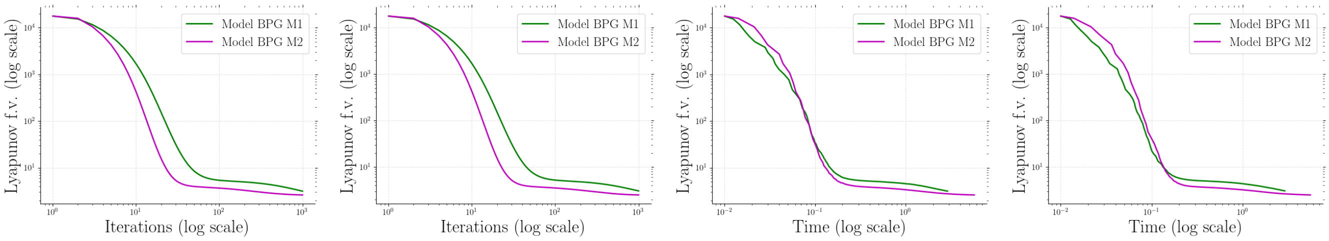
$$\mathcal{P}_1(x; y) := \frac{1}{M}\sum_{i=1}^{M}\left|(y^T A_i y - b_i)^2 + (y^T A_i y - b_i)\left\langle 2A_i y, x - y\right\rangle\right| + \mathcal{R}(x)\,. \tag{9.7.6}$$

Considering the Legendre function $h(x) = \frac{1}{4}\|x\|^4 + \frac{1}{2}\|x\|^2$ and [28, Lemma 5.1], a simple calculation reveals that the following MAP property holds true:

$$|\mathcal{P}_0(x) - \mathcal{P}_1(x; y)| \leq L_0 D_h(x, y)\,, \text{ for all } x, y \in \mathbb{R}^N\,, \tag{9.7.7}$$

with $L_0 \geq \sum_{i=1}^{M}(3\|A_i\|_F^2 + \|A_i\|_F\,|b_i|)$. In this setting, Model BPG subproblems are solved using Primal-Dual Hybrid Gradient algorithm (PDHG) [143].

We provide empirical results in Figure 9.1, where we show superior performance of Model BPG variants compared to IBPM-LS, in particular, with the model function provided in (9.7.6). For simplicity, we choose a constant step-size τ in all the iterations, such that $\tau \in (0, 1/L_0)$. We empirically validate Proposition 9.5.2.1 in Figure 9.2. All the assumptions required to deduce the global convergence of Model BPG are straightforward to verify, and we leave it as an exercise to the reader. Note that here $\operatorname{int}\operatorname{dom} h = \mathbb{R}^N$, thus the condition $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds trivially.



(A) L1 reg (B) Squared L2 reg (C) L1 reg (D) Squared L2 reg

FIGURE 9.2: We illustrate that when Model BPG applied to standard phase retrieval problem in (9.7.3), with model function chosen to be either Model 1 in (9.7.4) or Model 2 in (9.7.6), result in sequences where the Lyapunov function value evaluations are monotonically nonincreasing. In terms of iterations, Model BPG with Model 2 (Model BPG M2) is better than Model BPG with Model 1 (Model BPG M1). In terms of time, Model BPG M1 and Model BPG M2 perform almost the same, however, towards the end Model BPG M2 is faster in both the cases. By reg we mean regularization, and by Lyapunov f.v. we mean Lyapunov function values.

9.7.2 Robust phase retrieval

Now, we consider the robust phase retrieval problem, where the goal is the same as standard phase retrieval problem, that is to solve the system of quadratic equations in (9.7.2). It is well known that L1 loss is more robust to noise compared to squared L2 loss [70]. The problem in (9.7.3) uses squared L2 loss. Here, we consider L1 loss based robust phase retrieval problem, which involves solving the following optimization problem :

$$\min_{x\in\mathbb{R}^N} f(x), \quad f(x) := \frac{1}{M}\sum_{i=1}^{M}\left|x^T A_i x - b_i\right| + \mathcal{R}(x)\,,$$

where we set $\mathcal{R}(x) = \lambda\|x\|_1$ (L1 regularization) or $\mathcal{R}(x) = \frac{\lambda}{2}\|x\|^2$ (squared L2 regularization), for some $\lambda > 0$. Such an objective is preferred if the data obtained is noisy, and we require the solution that is robust to noise. We use the Prox-Linear setting described in Section 9.6.2, where for any $x \in \mathbb{R}^N$ we set the following:

$$f_0(x) := \mathcal{R}(x)\,,$$

$$(F(x))_i = x^T A_i x - b_i\,,\ \text{for all } i = 1,\ldots,M\,,$$

and for any $\tilde{y} \in \mathbb{R}^M$ we set

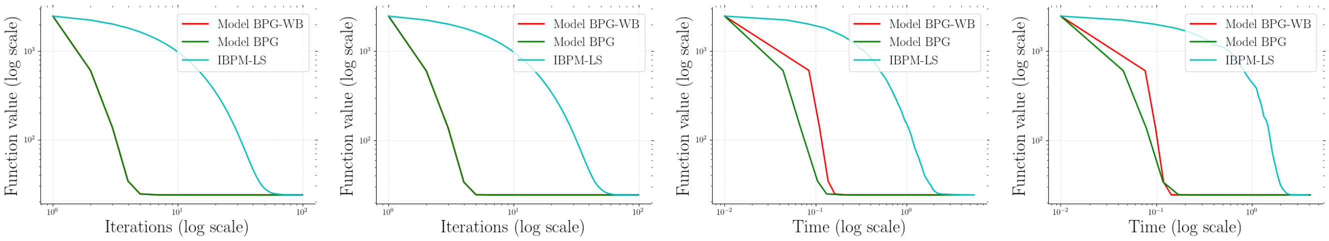
$$g(\tilde{y}) := \frac{1}{M}\|\tilde{y}\|_1\,,\ \text{for } \tilde{y} \in \mathbb{R}^M\,.$$

We consider the following model function. For fixed $y \in \mathbb{R}^N$, the model function $f(x;y)$ at $x \in \mathbb{R}^N$ is given by

$$f(x;y) := \frac{1}{M}\sum_{i=1}^{M}\left|y^T A_i y - b_i + \langle 2A_i y, x - y\rangle\right| + \mathcal{R}(x)\,. \tag{9.7.8}$$

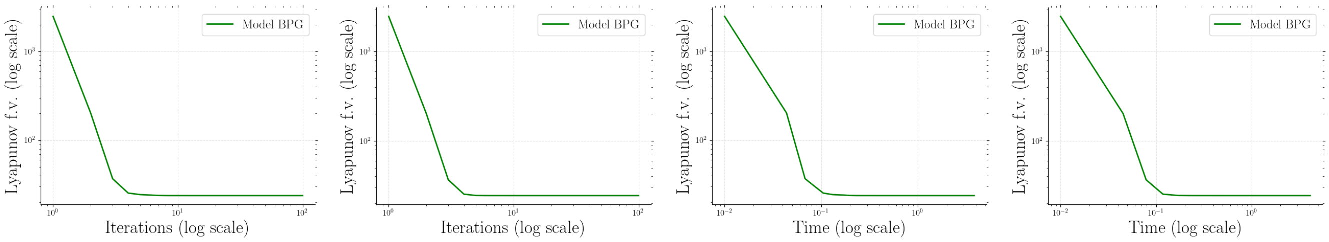
With the Legendre function $h(x) = \frac{1}{2}\|x\|^2$ and as a consequence of triangle property, a simple calculation reveals that for all $x, y \in \mathbb{R}^N$ we have

$$|f(x) - f(x;y)| \le 0.5L_1\|x - y\|^2\,,$$



(A) L1 reg (B) Squared L2 reg (C) L1 reg (D) Squared L2 reg

FIGURE 9.3: In this experiment we consider the performance of Model BPG vs Model BPG with Backtracking (denoted as Model BPG-WB) vs IBPM-LS [138] on robust phase retrieval problems, with both L1 and squared L2 regularization. Model BPG variants perform similarly and are better than IBPM-LS. By reg, we mean regularization.



(A) L1 reg (B) Squared L2 reg (C) L1 reg (D) Squared L2 reg

FIGURE 9.4: Under the same setting as in Figure 9.3, we illustrate that Model BPG when applied on robust phase retrieval problems, with both L1 and squared L2 regularization, results in sequences with monotonically decreasing Lyapunov function evaluations, thus validating Proposition 9.5.2.1. By reg we mean regularization, and by Lyapunov f.v. we mean Lyapunov function values.

with $L_1 \geq \frac{2\sum_{i=1}^{M}\lambda_{\max}(A_i)}{M}$. We use a constant step-size $\tau_k = \tau$ such that $\tau \in (0, 1/L_1)$. All the other assumptions of Model BPG are straightforward to verify and we leave it as an exercise to the reader. In each iteration of Model BPG, subproblems take the following form:

$$\underset{x\in\mathbb{R}^N}{\mathrm{Argmin}} \left\{ \frac{1}{M}\sum_{i=1}^{M} \left| y^T A_i y - b_i + \langle 2A_i y, x - y\rangle \right| + \mathcal{R}(x) + \frac{1}{2\tau}\|x - y\|^2 \right\},$$

which we solve using Primal-Dual Hybrid Gradient algorithm (PDHG) [143]. The empirical results are reported in Figure 9.3, where we illustrate the better performance of Model BPG based methods compared to IBPM-LS [138] on robust phase retrieval problems. We empirically validate Proposition 9.5.2.1 in Figure 9.4. Note that here $\mathrm{int\,dom}\, h = \mathbb{R}^N$, thus the condition $\omega^{\mathrm{int\,dom}\, h}(x_0) = \omega(x_0)$ holds trivially.

9.7.3 Poisson linear inverse problems

We now consider a broad class of problems with varied practical applications, known as Poisson inverse problems [10, 18, 131, 139]. The problem setting is as follows. For all $i = 1, \ldots, M$, let $b_i > 0$, $a_i \neq 0$ and $a_i \in \mathbb{R}_+^N$ be known. Moreover, we have for any $x \in \mathbb{R}_+^N$, $\langle a_i, x\rangle > 0$ and $\sum_{i=1}^{M}(a_i)_j > 0$, for all $j = 1, \ldots, N$, $i = 1, \ldots, M$. Equipped with these notions, one can write the optimization problem of Poisson linear inverse problems as following:

$$\min_{x\in\mathbb{R}_+}\left\{f(x):=\sum_{i=1}^{M}\left(\langle a_i,x\rangle-b_i\log(\langle a_i,x\rangle)\right)+\phi(x)\right\}\,, \tag{9.7.9}$$

where ϕ is the regularizing function, which is potentially non-convex. For simplicity, we set $\phi=0$. The function $f_1:\mathbb{R}^N\to\overline{\mathbb{R}}$ at any $x\in\mathbb{R}^N$ is defined as following:

$$f_1(x):=\sum_{i=1}^{M}\left(\langle a_i,x\rangle-b_i\log(\langle a_i,x\rangle)\right)\,.$$

Note that the function f_1 is coercive. Since f_1 is a continuous function, its level set restricted to $\mathbb{R}_+$, i.e., $C:=\{x\geq 0: f_1(x)\leq f_1(x_0)\}$ is compact, for any $x_0\in\mathbb{R}_+$. In order to apply Model BPG, we need h such that the MAP property is satisfied. We consider the Legendre function $h:\mathbb{R}^N_{++}\to\mathbb{R}$ that is given by

$$h(x)=-\sum_{i=1}^{N}\log(x_i)\,,\qquad \text{for all } x\in\mathbb{R}^N_{++}, \tag{9.7.10}$$

where x_i is the i^{th} coordinate of x. The above given function h is also known as Burg's entropy. Consider the following lemma.

Lemma 9.7.3.1. *Let h be defined as in (9.7.10). For $L\geq\sum_{i=1}^M b_i$, the function $Lh-f_1$ and $Lh+f_1$ is convex on $\mathbb{R}^N_{++}$, or equivalently the following L-smad property or the MAP property holds true:*

$$-LD_h(x,\bar{x})\leq f_1(x)-f_1(\bar{x})-\langle\nabla f_1(\bar{x}),x-\bar{x}\rangle\leq LD_h(x,\bar{x})\,,\ \textit{for all}\ x,\bar{x}\in\mathbb{R}^N_{++}\,. \tag{9.7.11}$$

Proof. The proof of convexity of $Lh-f_1$ follows from [10, Lemma 7]. The function $Lh+f_1$ is convex as f_1 is convex. □

When Model BPG is applied to solve (9.7.9) with h given in (9.7.10), if the limit points of the sequence generated by Model BPG lie in $\operatorname{int}\operatorname{dom}h$, our global convergence result is valid. However, it is difficult to guarantee such a condition. This is because, there can exist subsequences for which certain components of the iterates can tend to zero. In such a scenario, some components of $\nabla^2 h(x_k)$ will tend to ∞, which will lead to the failure of the relative error condition in Lemma 9.5.3.1. In that case, our analysis cannot guarantee the global convergence of the sequence generated by Model BPG.

Thus, in such a scenario it is important to guarantee that the iterates of Model BPG lie in $\mathbb{R}^N_{++}$. To this regard, we modify the problem (9.7.9), by adding certain constraint set, such that all the limit points lie in $\operatorname{int}\operatorname{dom}h$. Then, the global convergence of the sequence generated by Model BPG sequence can be guaranteed. The full objective after the modification is provided below

$$\min_{x\in\mathbb{R}^N}\left\{f(x):=\delta_{C_\varepsilon}(x)+\sum_{i=1}^{M}\left(\langle a_i,x\rangle-b_i\log(\langle a_i,x\rangle)\right)+\phi(x)\right\}\,, \tag{9.7.12}$$

where for certain $\varepsilon>0$ we denote

$$C_\varepsilon=\{x:x_i\geq\varepsilon,\ \forall i=1,\ldots,N\}\,,$$

and $\delta_{C_\varepsilon}(\,\cdot\,)$ is the indicator function of the set C_ε. We consider $\phi = 0$ or $\phi(x) = \lambda\|x\|_1$ or $\phi(x) = \lambda\frac{\|x\|^2}{2}$, with certain $\lambda > 0$. Note that $C_\varepsilon \subset \mathbb{R}_+$. For practical purposes, C_ε is almost the same as $\mathbb{R}_+$, when ε is chosen sufficiently small. Note that the choice of ε is only heuristic. To this end, with $\bar{x} \in C_\varepsilon$, we consider the following model function which, when evaluated at x gives:

$$f(x; \bar{x}) := \delta_{C_\varepsilon}(x) + f_1(\bar{x}) + \langle \nabla f_1(\bar{x}), x - \bar{x}\rangle + \phi(x)\,. \tag{9.7.13}$$

The Legendre function in (9.7.10) is still valid as $C_\varepsilon \subset \mathbb{R}_+$, and the MAP property holds true as a consequence of Lemma 9.7.3.1. The coercivity of the function f along with Proposition 9.3.0.1 implies that the iterates of Model BPG will lie in the compact convex set $\{x : f(x) \le f(x_0)\}$. Thus, the sequence generated by Model BPG is bounded. The analysis falls under the category of additive composite problems given in Section 9.6.1, where we set $f_1 := f_1$ and $f_0(\,\cdot\,) := \delta_{C_\varepsilon}(\,\cdot\,) + \phi(\,\cdot\,)$. In the earlier discussion, we have proved the crucial assumptions for applying Model BPG to Poisson linear inverse problems. The rest of the assumptions in Theorem 10.5.1.1 are straightforward to verify and we leave it as an exercise to the reader. We now provide closed form expressions for the update step (9.3.4) in three settings of ϕ.

Closed form update step - No regularization. Set $\phi = 0$. The update step of Model BPG involves solving the following subproblem:

$$x_{k+1} \in \operatorname{argmin}_x \delta_{C_\varepsilon}(x) + f(x_k) + \langle \nabla f(x_k), x - x_k\rangle + \frac{1}{\tau_k} D_h(x, x_k)\,.$$

The optimality condition for the i^{th} component of x_{k+1} due to Fermat's rule is given by

$$0 = (v_{k+1})_i + \nabla f(x_k)_i + \frac{1}{\tau_k}\left(\frac{1}{(x_k)_i} - \frac{1}{(x_{k+1})_i}\right),$$

for some $v_{k+1} \in N_{C_\varepsilon}(x_{k+1})$. Thus, we deduce that with τ_k chosen such that $1 + \tau_k \nabla f(x_k)_i (x_k)_i > 0$, for $i = 1, \ldots, N$, the solution is given by

$$x_{k+1} = \max\left\{\varepsilon, \frac{x_k}{1 + \tau_k \nabla f(x_k) x_k}\right\}, \tag{9.7.14}$$

where all the operations are performed element-wise.

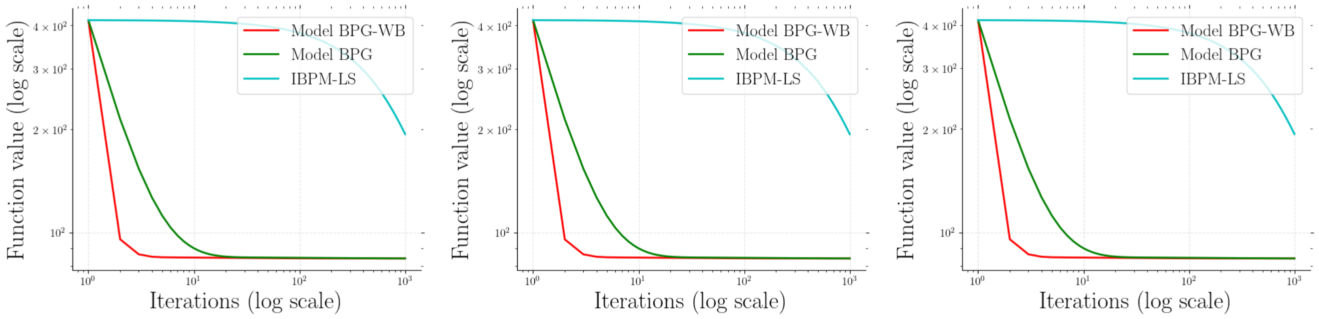
Closed form update step - L1 regularization. We consider here the standard L1 regularization setting, where with certain $\lambda > 0$ we set $\phi(x) = \lambda\|x\|_1$. The update step of Model BPG involves solving the following subproblem:

$$x_{k+1} \in \operatorname{argmin}_x \delta_{C_\varepsilon}(x) + \lambda\|x\|_1 + f(x_k) + \langle \nabla f(x_k), x - x_k\rangle + \frac{1}{\tau_k} D_h(x, x_k)\,.$$

Based on [10, Section 5.2] and Fermat's rule we deduce that with τ_k chosen such that $1 + \tau_k \lambda (x_k)_i + \tau_k \nabla f(x_k)_i (x_k)_i > 0$, for $i = 1, \ldots, N$, the closed form solution is given by

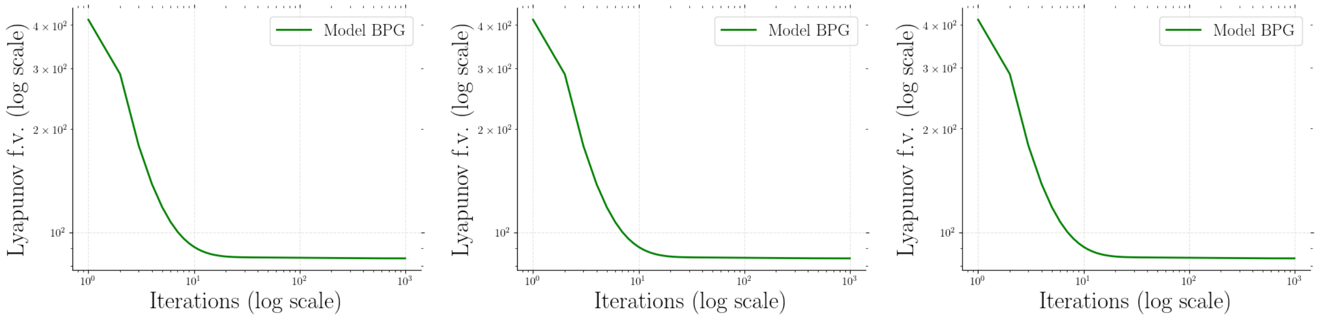
$$x_{k+1} = \max\left\{\varepsilon, \frac{x_k}{1 + \tau_k \lambda x_k + \tau_k \nabla f(x_k) x_k}\right\}, \tag{9.7.15}$$

where all the operations are performed element-wise.



(A) L1 regularization (B) Squared L2 regularization (C) No regularization

FIGURE 9.5: In this experiment we compare the performance of Model BPG, Model BPG with Backtracking (denoted as Model BPG-WB) and IBPM-LS [138] on Poisson linear inverse problems with L1 regularization, squared L2 regularization and with no regularization. We set the regularization parameter λ to 0.1. The plots illustrate that Model BPG-WB is faster in all the settings, followed by Model BPG.



(A) L1 regularization (B) Squared L2 regularization (C) No regularization

FIGURE 9.6: By Lyapunov f.v. we mean Lyapunov function values. Under the same setting as in Figure 9.5, we illustrate here that Model BPG results in sequences that have monotonically nonincreasing Lyapunov function value evaluations.

Closed form update step - L2 regularization. We consider here the standard L2 regularization setting, where with certain $\lambda > 0$ we set $\phi(x) = \frac{\lambda}{2}\|x\|_2^2$. The update step of Model BPG involves solving the following subproblem:

$$x_{k+1} \in \operatorname{argmin}_x \delta_{C_\varepsilon}(x) + \frac{\lambda}{2}\|x\|_2^2 + f(x_k) + \langle \nabla f(x_k), x - x_k \rangle + \frac{1}{\tau_k} D_h(x, x_k)\,.$$

The optimality condition for the i^{th} component of x_{k+1} due to Fermat's rule is given by

$$0 = (v_{k+1})_i + \lambda(x_{k+1})_i + \nabla f(x_k)_i + \frac{1}{\tau_k}\left(\frac{1}{(x_k)_i} - \frac{1}{(x_{k+1})_i}\right),$$

for some $v_{k+1} \in N_{C_\varepsilon}(x_{k+1})$. Based on [10, Section 5.2] we deduce that with τ_k chosen such that $1 + \tau_k \nabla f(x_k)_i (x_k)_i + \tau_k \lambda \varepsilon > 0$, for $i = 1, \ldots, N$, the closed form solution is given by

$$x_{k+1} = \max\left\{\varepsilon, \frac{\sqrt{(1 + \tau_k x_k \nabla f(x_k))^2 + 4\lambda\tau_k x_k^2} - (1 + \tau_k x_k \nabla f(x_k))}{2\lambda\tau_k x_k}\right\}, \tag{9.7.16}$$

where all the operations are performed element-wise.

The empirical results are reported in Figure 9.5, where we illustrate the better performance of Model BPG based methods compared to IBPM-LS [138], when applied on Poisson linear inverse problems. We empirically validate Proposition 9.5.2.1 in Figure 9.6. Note that here $\operatorname{int}\operatorname{dom} h = \mathbb{R}^N_{++}$. Based on the aforementioned closed form solutions it is clear that the sequence generated by Model BPG lies in C_ε. The condition $C_\varepsilon \subset \operatorname{int}\operatorname{dom} h$ implies that $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds true.

9.8 Chapter conclusion

Bregman proximal minimization framework is prominent in solving additive composite problems, in particular, using BPG [28] algorithm or its variants (Chapter 5). However, extensions to generic composite problems was an open problem. To this regard, based on foundations of [60, 139], we proposed Model BPG algorithm that is applicable to a vast class of non-convex non-smooth problems, including generic composite problems. Model BPG relies on certain function approximation, known as model function, which preserves first order information about the function. The model error is bounded via certain Bregman distance, which drives the global convergence analysis of the sequence generated by Model BPG. The analysis is nontrivial and requires significant changes compared to the standard analysis of [6, 7, 26, 28]. Moreover, we numerically illustrate the superior performance of Model BPG on various real world applications.

Chapter 10

Inertial Model BPG

10.1 Abstract

In this chapter, we propose an inertial variant of the Model BPG algorithm. In particular, we use the inertial technique of CoCaIn BPG algorithm along with Model BPG, to propose Model CoCaIn BPG algorithm. We also show that the sequence generated by Model CoCaIn BPG is globally convergent to a critical point of the function. We use similar convergence analysis techniques used in Model BPG and CoCaIn BPG convergence analysis, however, using a new Lyapunov function. We also supplement our theory with empirical results on robust phase retrieval problem, where we show that when function value vs iterations is considered, the Model CoCaIn BPG outperforms other state of the art optimization methods.

10.2 Introduction

We continue the setting of Chapter 9. In this chapter, we answer the question “Can we incorporate Nesterov’s inertia into Model BPG?”. In this regard, we incorporate the inertial strategy of CoCaIn BPG (proposed in

Chapter 5) in Model BPG setting to propose Model CoCaIn BPG. As mentioned in Chapter 9, the Model BPG framework is helpful in a unified analysis of several objectives, in particular the objectives arising in additive composite problems and composite problems. Thus, Model BPG can be used as a foundation to propose and analyse related algorithms, which otherwise need to be explored for each individual problem setting.

The crucial idea behind Model CoCaIn BPG is that the local lower and upper bounds of the function in the MAP property can be leveraged to use appropriate inertia, similar to CoCaIn BPG algorithm using the bounds in the L-smad property. In particular, the lower bounds governs the inertial parameter and the upper bound governs the step-size parameter. A straightforward advantage of Model CoCaIn BPG is the ready applicability to generic composite problems, whereas CoCaIn BPG is valid only for additive composite problems.

10.2.1 Contributions

Our main contribution is the proposal of Model CoCaIn BPG. Even though the high level idea of its convergence analysis remains the same as that of Model BPG, there are few significant changes that are crucial, which we detail below.

- Firstly, the Lyapunov function used for Model BPG analysis is not suitable anymore. Thus, we propose a new Lyapunov function which incorporates the inertial nature of Model CoCaIn BPG via a dependency on the inertial parameter.

- The analysis of CoCaIn BPG relies on a Lyapunov function that includes additive components of the objective function and the Bregman distance upto a scaling factor. However, as mentioned in the Model BPG setting, using the objective value in the measure of progress can be restrictive for analyzing generic composite setting. Thus, the Lyapunov function we use has additive components that involve the model function rather than the function. We note that we heavily rely on the ideas used in the CoCaIn BPG analysis (Chapter 5) to prove the descent property for the Lyapunov function.

- A new semi-convexity assumption (see Assumption L) of the model function is considered for the analysis of Model CoCaIn BPG and it is strictly weaker than the condition considered for CoCaIn BPG convergence analysis.

10.2.2 Related work

Using the Nesterov's inertial strategy [15, 126], in this chapter we propose Model CoCaIn BPG, an inertial variant of the Model BPG. The theoretical tools [136] we used for the global convergence analysis of Model CoCaIn BPG are essentially the same as that of Model BPG (Chapter 9). Our work generates ideas that can possibly be used to analyse several similar algorithms. For example, various other inertial variants using Polyak's momentum [145] or regularized nonlinear acceleration technique [153] can be analysed using similar ideas as that of Model CoCaIn BPG.

10.3 Model CoCaIn BPG

In addition to Assumptions F, G, I, J, K in Chapter 9, we also require the following assumption.

Assumption L (Algorithm)**.** We make the following assumptions:

(i) For any $\bar{x} \in \operatorname{int} \operatorname{dom} h \cap \operatorname{dom} f$, the function $f(x;\bar{x}) - \frac{\alpha(\bar{x})}{2}\|x\|_2^2$ is convex for certain constant $\alpha(\bar{x}) \leq 0$.

(ii) For all $x_1,\ x_2 \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, there exists $\gamma \in [0,1]$ such that $y := x_1 + \gamma(x_1 - x_2)$ lies in $\operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$.

Based on the above assumption, we propose the following algorithm.

Algorithm 8: Model CoCaIn BPG: Model based CoCaIn Bregman Proximal Gradient

- **Initialization:** Select $x_0 = x_1 \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$. Choose $\underline{\tau}, \bar{\tau}$ such that $0 < \underline{\tau} < \bar{\tau} < (1/\bar{L})$. Set $\delta, \epsilon > 0$ with $1 > \delta > \epsilon$ and $\bar{L}_0 \geq \frac{-\alpha(x_0)}{(1-\delta)\sigma}$.
- **For each $k \geq 1$:** Compute

$$y_k = x_k + \gamma_k \left(x_k - x_{k-1}\right) \in \operatorname{int} \operatorname{dom} h, \tag{10.3.1}$$

where γ_k is chosen such that

$$(\delta - \epsilon)\, D_h\left(x_{k-1}, x_k\right) \geq \left(1 + \underline{L}_k \tau_{k-1}\right) D_h\left(x_k, y_k\right) \tag{10.3.2}$$

holds and such that $\underline{L}_k$ satisfies

$$f(x_k) \geq f(x_k; y_k) - \underline{L}_k D_h\left(x_k, y_k\right). \tag{10.3.3}$$

- Now, choose $\bar{L}_k \geq \max\left\{\bar{L}_{k-1}, \frac{-\alpha(y_k)}{(1-\delta)\sigma}\right\}$, set $\tau_k \leq \min\left\{\tau_{k-1}, \bar{L}_k^{-1}\right\}$ and compute

$$x_{k+1} \in \operatorname*{Argmin}_{x \in \mathbb{R}^N} \left\{ f(x; y_k) + \frac{1}{\tau_k} D_h\left(x, y_k\right) \right\} \tag{10.3.4}$$

with $\bar{L}_k$ fulfilling

$$f(x_{k+1}) \leq f(x_{k+1}; y_k) + \bar{L}_k D_h\left(x_{k+1}, y_k\right). \tag{10.3.5}$$

The steps are essentially the same as that of CoCaIn BPG, however the update step is similar to Model BPG. For brevity, we skip the discussion regarding the steps of the algorithm.

10.3.1 Implementation and double backtracking

The crucial aspect of Model CoCaIn BPG is the double backtracking technique. Note that there are two backtracking steps in the algorithm, one to control step-size and the other to control inertia. The standard way of doing double backtracking might not be feasible as pointed out in Chapter 5, however double backtracking technique from CoCaIn BPG (see Chapter 5) makes this feasible. We extend this strategy to incorporate the model function hence the name Model CoCaIn BPG. The implementation of Model CoCaIn BPG is similar to the implementation of CoCaIn BPG (see Section 5.6.4), with only involving minor modification due to the usage of model functions. Thus, we skip the discussion here. Note that when $\gamma_k = 0$ in Model CoCaIn BPG, then the resulting algorithm is Model BPG with Backtracking.

10.4 Global convergence analysis of Model CoCaIn BPG

10.4.1 Descent property

Similar to Model BPG, we use a Lyapunov function in order to analyse Model CoCaIn BPG. Here, with $\delta > 0$ we consider the following Lyapunov function:

$$\begin{aligned}&G^h_{\bar{L}} : \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \overline{\mathbb{R}}\,,\\ &\quad (x_1, x_2, x_3, \gamma, \tau, \theta) \to \tau \left(f(x_1; y) + \theta D_h(x_1, y) - v(\mathcal{P}_M)\right) + \delta D_h\left(x_2, x_1\right).\end{aligned}$$

where $y = x_2 + \gamma(x_2 - x_3)$ for certain $\gamma \in \mathbb{R}$. Firstly, we need the following technical result.

Lemma 10.4.1.1 (Function descent property)**.** *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG. Then, for all $k \in \mathbb{N}$, we have*

$$\begin{aligned} f(x_k; y_{k-1}) + \bar{L}_{k-1} D_h(x_k, y_{k-1}) \geq f\left(x_{k+1}; y_k\right) + \bar{L}_k D_h(x_{k+1}, y_k) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 \\ + \frac{1}{\tau_k} D_h\left(x_k, x_{k+1}\right) - \left(\frac{1}{\tau_k} + \underline{L}_k\right) D_h\left(x_k, y_k\right). \end{aligned} \tag{10.4.1}$$

The proof is provide in Section G.1 in the appendix.

We now provide the descent property in terms of Lyapunov function values for the sequence generated by Model CoCaIn BPG.

Proposition 10.4.1.1. *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG. Then, for all $k \in \mathbb{N}$, we have*

$$G^h_{\bar{L}}\left(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}, \tau_{k-1}, \bar{L}_{k-1}\right) \geq G^h_{\bar{L}}\left(x_{k+1}, x_k, x_{k-1}, \gamma_k, \tau_k, \bar{L}_k\right) + \varepsilon D_h\left(x_{k-1}, x_k\right). \tag{10.4.2}$$

Proposition 10.4.1.2. *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG. Then, the following assertions hold:*

(i) *The sequence $\left\{G^h_{\bar{L}}\left(x_{k+1}, x_k, x_{k-1}, \gamma_k, \tau_k, \bar{L}_k\right)\right\}_{k\in\mathbb{N}}$ is nonincreasing.*

(ii) *$\sum_{k=1}^{\infty} D_h\left(x_{k-1}, x_k\right) < \infty$, and hence the sequence $\left\{D_h\left(x_{k-1}, x_k\right)\right\}_{k\in\mathbb{N}}$ converges to zero.*

(iii) *$\min_{1\leq k\leq n} D_h\left(x_{k-1}, x_k\right) \leq \frac{G^h_{\bar{L}}\left(x_1, x_0, x_{-1}, \gamma_0, \tau_0, \bar{L}_0\right)}{(\varepsilon n)}$.*

Proof. (i) This follows trivially from Proposition 10.4.1.1, since $\varepsilon > 0$.

(ii) Let n be a positive integer. Summing (10.4.2) from $k = 1$ to n we get

$$\begin{aligned}\sum_{k=1}^{n} D_h\left(x_{k-1}, x_k\right) &\leq \frac{1}{\varepsilon}\left(G^h_{\bar{L}}\left(x_1, x_0, x_{-1}, \gamma_0, \tau_0, \bar{L}_0\right) - G^h_{\bar{L}}\left(x_{n+1}, x_n, x_{n-1}, \gamma_n, \tau_n, \bar{L}_n\right)\right)\\ &\leq \frac{1}{\varepsilon} G^h_{\bar{L}}\left(x_1, x_0, x_{-1}, \gamma_0, \tau_0, \bar{L}_0\right),\end{aligned}$$

since $G^h_{\bar{L}}\left(x_{n+1}, x_n, x_{n-1}, \gamma_n, \tau_n, \bar{L}_n\right) \geq 0$. Taking the limit as $n \to \infty$, we obtain the first desired assertion, from which we immediately deduce that $\left\{D_h\left(x_{k-1}, x_k\right)\right\}_{k\in\mathbb{N}}$ converges to zero.

(iii) From (B.4.1) we also obtain,

$$n \min_{1\leq k\leq n} D_h\left(x_{k-1}, x_k\right) \leq \sum_{k=1}^{n} D_h\left(x_{k-1}, x_k\right) \leq \frac{1}{\varepsilon} G_{\bar{L}}^h\left(x_1, x_0, x_{-1}, \gamma_0, \tau_0, \bar{L}_0\right),$$

which after division by n yields the desired result. □

Note that there is a finite increase of $\bar{L}_k$ in each iteration. This implies that after a finite number of iterations there is no increase in the value of $\bar{L}_k$, resulting in stagnant value of τ_k. This means, there exists a $K > 1$, such that for all $k \geq K$, we set $\tau_k = \tau$.

For a comprehensive discussion, please see Section 5.6.2. Thus, for $k \geq K$, we use the Lyapunov function $G_{\bar{L}}^h : \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R} \to \overline{\mathbb{R}}$ that is given by

$$G_{\bar{L}}^h(x, y, z, \gamma) := f(x; y_1) + \bar{L}D_h(x, y_1) + \delta_1 D_h\left(y, x\right)$$

where $\delta_1 = \frac{\delta}{\tau}$, $x, y, z \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, $\gamma \in [0, 1]$ satisfying $y_1 := y + \gamma(y - z) \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$ and ∞ otherwise. Without loss of generality, we denote $k \geq K$ as $k \geq 1$, as the subsequent analysis relies only on $G_{\bar{L}}^h$ defined above. Note that we removed the dependency on $\tau_k, \bar{L}_k$ in the Lyapunov function, as $\bar{L}_k, \tau_k$ become constant.

Proposition 10.4.1.3. *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG. Then, the following assertions hold for $k \geq 1$:*

(i) *The sequence $\left\{G_{\bar{L}}^h\left(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}\right)\right\}_{k\in\mathbb{N}}$ is nonincreasing.*

(ii) $\sum_{k=1}^{\infty} D_h\left(x_{k-1}, x_k\right) < \infty$, *and hence the sequence $\left\{D_h\left(x_{k-1}, x_k\right)\right\}_{k\in\mathbb{N}}$ converges to zero.*

(iii) $\min_{1\leq k\leq n} D_h\left(x_{k-1}, x_k\right) \leq \frac{G_{\bar{L}}^h(x_1, x_0, x_{-1}, \gamma_0) - v(\mathcal{P}_M)}{(\varepsilon n)}$.

Proof. The proof is similar to the proof of Proposition 10.4.1.2, thus we skip it. □

10.4.2 Relative error condition

We now consider the relative error condition, that is crucial for the global convergence analysis of Model CoCaIn BPG.

Lemma 10.4.2.1. *Let Assumptions F, G, I hold and let $h \in C^2$. Then, there exists a constant $D_1, D_2, B_1, B_2 >$ 0 such that the following holds:*

$$\|\partial G_{\bar{L}}^h\left(x_{k+1}, x_k, x_{k-1}, \gamma_k\right)\|_- \leq D_1\|x_{k+1} - x_k\|_2 + D_2\|x_k - x_{k-1}\|_2 + B_1\|(x_{k+1} - x_k)\|_2^2 + B_2\|(x_k - x_{k-1})\|_2^2, \tag{10.4.3}$$

where $\|\partial G_{\bar{L}}^h\left(x_{k+1}, x_k, x_{k-1}, \gamma_k\right)\|_- := \inf_{\zeta\in\partial G_{\bar{L}}^h(x_{k+1}, x_k, x_{k-1}, \gamma_k)} \|\zeta\|$.

The proof of the above lemma is provided in Section G.3 in the appendix.

10.4.3 Subsequential convergence

We now show that the sequence generated by Model CoCaIn BPG $(x_k)_{k\in\mathbb{N}}$ indeed attains $\|x_{k+1} - x_k\| \to 0$ as $k \to \infty$, which in turn enables the application of Proposition 9.5.4.1 to deduce the properties of the sequence generated by Model BPG, which later proves to be crucial for the proof of global convergence.

Proposition 10.4.3.1. *Let Assumption F, G, I hold. Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG. Then, we have*

$$\varepsilon D_h(x_{k+1}, x_k) \to 0\,, \quad \textit{as } k \to \infty\,. \tag{10.4.4}$$

The condition $\varepsilon > 0$ implies that $x_{k+1} - x_k \to 0$ as $k \to \infty$.

Proof. Note that the sequence $(x_k)_{k\in\mathbb{N}}$ is a bounded sequence (see Remark 9.3.0.2). By the descent property (Proposition 10.4.2) we have

$$G^h_{\bar{L}}\left(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}\right) \geq G^h_{\bar{L}}\left(x_{k+1}, x_k, x_{k-1}, \gamma_k\right) + \varepsilon D_h\left(x_{k-1}, x_k\right).$$

Summing on both sides and due to the convergence of Lyapunov function, using Proposition 10.4.2, we obtain

$$\sum_{k=1}^{\infty} \Big(\varepsilon D_h(x_{k+1}, x_k)\Big) \leq G^h_{\bar{L}}(x_0, x_{-1}, x_{-2}, \gamma_{-1}) - \lim_{k\to\infty} G^h_{\bar{L}}\left(x_{k+1}, x_k, x_{k-1}, \gamma_k\right) < \infty\,,$$

which implies (10.4.4). For $\varepsilon > 0$, Assumption I(iii) together with (10.4.4) imply $x_{k+1} - x_k \to 0$ as $k \to \infty$. □

Analyzing the full set of limit points of the sequence generated by Model CoCaIn BPG is difficult, as illustrated in [139]. Obtaining the global convergence is still an open problem. Moreover, the work in [139] relies on convex model functions.

In order to simplify slightly the setting, we restrict the set of limit points to the set $\operatorname{int}\operatorname{dom} h$. Such a choice may appear to be restrictive, however, Model BPG when applied to many practical problems results in sequences that have this property as illustrated in Section 9.7.

The subset of $G^h_{\bar{L}}$-attentive (similar to f-attentive) limit points is

$$\begin{aligned}\omega_{G^h_{\bar{L}}}(x_0) := \Big\{&(x,y,z,\gamma) \in \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \times [0,1] \\ &\mid \exists K \subset \mathbb{N} \colon \left(x_k, x_{k-1}, x_{k-2}, \gamma_k, G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1})\right) \underset{K}{\to} \left(x, y, z, \gamma, G^h_{\bar{L}}(x,y,z,\gamma)\right)\Big\}\,.\end{aligned}$$

Also, we define $\omega^{(\operatorname{int}\operatorname{dom} h)^3\times[0,1]}_{G^h_{\bar{L}}} := \omega_{G^h_{\bar{L}}} \cap (\operatorname{int}\operatorname{dom} h \times \operatorname{int}\operatorname{dom} h \times \operatorname{int}\operatorname{dom} h \times [0,1])$.

Proposition 10.4.3.2. *Let Assumptions F, G, I hold. Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by Model CoCaIn BPG with $\gamma_k \to \gamma$. Then, the following holds:*

(i) $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$,

(ii) $x \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ *if and only if* $(x,x,x,\gamma) \in \omega^{(\operatorname{int}\operatorname{dom} h)^3\times[0,1]}_{G^h_{\bar{L}}}(x_0)$.

(iii) $G^h_{\bar{L}}$ *is constant and finite on* $\omega^{(\operatorname{int}\operatorname{dom} h)^3\times[0,1]}_{G^h_{\bar{L}}}(x_0)$ *and f is constant and finite on* $\omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ *with same value.*

The proof is provided in Section G.4 in the appendix. The set of critical points of $G^h_{\bar{L}}$ is denoted as

$$\operatorname{crit}(G^h_{\bar{L}}) := \left\{(x,y,z,\gamma) \in \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \times [0,1] :\ (0,0,0,0) \in \partial G^h_{\bar{L}}(x,y,z,\gamma)\right\}.$$

Theorem 10.4.3.1 (Sub-sequential convergence to a stationary point)**.** *Let Assumptions F, G, I hold and let $\varepsilon > 0$. If the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model CoCaIn BPG is bounded then*

$$\omega_{G^h_{\bar{L}}}^{(\mathrm{int\,dom}\,h)^3\times[0,1]}(x_0) \subset \mathrm{crit}\,(G^h_{\bar{L}})\,. \tag{10.4.5}$$

Proof. From (10.4.3), we have

$$\begin{aligned}\|\partial G^h_{\bar{L}}(x_{k+1}, x_k, x_{k-1}, \gamma_k)\|_- \leq\, & D_1\|x_{k+1} - x_k\|_2 + D_2\|x_k - x_{k-1}\|_2 \\ & + B_1\|x_{k+1} - x_k\|_2^2 + B_2\|x_k - x_{k-1}\|_2^2\end{aligned} \tag{10.4.6}$$

for some constants $D_1, D_2, B_1, B_2 > 0$. Using $\|x_{k+1} - x_k\|_2 \to 0$ and $\|x_k - x_{k-1}\|_2 \to 0$ and Proposition 10.4.3.2(i) yields (10.4.5), by the closedness property of the limiting subdifferential . □

10.4.4 Global convergence

Using the similar strategy as in Lemma 9.5.5.1, under Assumption J it is straightforward to deduce that the Lyapunov function $G^h_{\bar{L}}$ is definable in $\mathcal{O}$, and satisfies KL property at any point of $\mathrm{dom}\,\partial G^h_{\bar{L}}$. Based on the strategy used in the proof of Theorem 9.5.5.2, we arrive at the following global convergence result of Model CoCaIn BPG.

Theorem 10.4.4.1 (Global convergence to a stationary point under KL property)**.** *Let Assumptions F, G, I, J, K hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by Model CoCaIn BPG (Algorithm 8) with $\tau_k = \tau$ for certain $\tau > 0$, $\gamma_k \to \gamma$ for certain $\gamma > 0$, and the condition $\omega^{\mathrm{int\,dom}\,h}(x_0) = \omega(x_0)$ holds true. Then, convergent subsequences are $G^h_{\bar{L}}$-attentive convergent, and*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty \qquad \textit{(finite length property)}\,.$$

Moreover, the sequence $(x_k)_{k\in\mathbb{N}}$ converges to x such that (x, x, x, γ) is a critical point of $G^h_{\bar{L}}$. Additionally, the point x is a critical point of f.

The proof is provided in Section G.5 in the appendix.

10.4.5 Convergence rates

It is possible to deduce the following convergence rates for Model CoCaIn BPG for a certain class of desingularizing functions.

Theorem 10.4.5.1 (Convergence rates)**.** *Under the conditions of Theorem 10.4.4.1, let the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model CoCaIn BPG converge to $x \in \mathrm{dom}\,f \cap \mathrm{int\,dom}\,h$, and let the Lyapunov function $G^h_{\bar{L}}$ satisfy Kurdyka–Łojasiewicz property with the following desingularizing function:*

$$\varphi(s) = cs^{1-\theta}\,,$$

for certain $c > 0$ and $\theta \in [0, 1)$. Then, we have the following:

- *If $\theta = 0$, then $(x_k)_{k\in\mathbb{N}}$ converges in finite number of steps.*

- *If $\theta \in (0, \frac{1}{2}]$, then there exists $\rho \in [0, 1)$ and $G > 0$ such that for all $k \geq 0$ we have*

$$\|x_k - x\| \leq G\rho^k \,.$$

- *If $\theta \in (\frac{1}{2}, 1)$, then there exists $G > 0$ such that for all $k \geq 0$ we have*

$$\|x_k - x\| \leq Gk^{-\frac{1-\theta}{2\theta-1}} \,.$$

The proof technique remains the same as that of Theorem 9.5.7.1, hence we skip it for brevity.

10.5 Examples

Similar to the Chapter 9, we consider examples suitable for Model CoCaIn BPG. In particular, we consider the additive composite problems and the generic composite problems.

10.5.1 Additive composite problems

We consider the same setting as in 9.6.1. We require the following assumption, apart from the assumptions mentioned in 9.6.1.

($\tilde{C}1$) f_0 is semi-convex with modulus $\alpha \in \mathbb{R}$.

($\tilde{C}2$) For all x_1, $x_2 \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, there exists $\gamma \in [0, 1]$ such that $y := x_1 + \gamma(x_1 - x_2)$ lies in $\operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$.

The adaptation of Model CoCaIn BPG to additive composite setting is provided below.

CoCaIn BPG is Model CoCaIn BPG (Algorithm 8) with

$$f(x; y_k) := f_0(x) + f_1(y_k) + \langle \nabla f_1(y_k), x - y_k \rangle \,. \tag{10.5.1}$$

Clearly, the Assumption ($\tilde{C}1$) implies the Assumption L(i) and Assumption ($\tilde{C}2$) implies the Assumption L(ii). As discussed above and in Section 9.6.1, Assumptions (C1), ($\tilde{C}1$), ($\tilde{C}2$), (C2), (B1), (B2), (B3) imply Assumptions F, G, I, J, K. Thus, as a consequence of Theorem 10.4.4.1 we obtain the following result which provides the global convergence of the sequence generated by CoCaIn BPG to a stationary point.

Theorem 10.5.1.1 (Global convergence of CoCaIn BPG sequence)**.** *Let Assumptions (C1), ($\tilde{C}1$), ($\tilde{C}2$), (C2), (B1), (B2), (B3) hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by CoCaIn BPG and the condition $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds true. Let $\tau_k \to \tau$ for certain $\tau > 0$. Then, the sequence $(x_k)_{k\in\mathbb{N}}$ has finite length, that is*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty \,,$$

and the sequence $(x_k)_{k\in\mathbb{N}}$ converges to x, which is a critical point of f.

The conclusion we obtained in Chapter 5 matches with the above theorem.

10.5.2 Composite problems

We consider the same setting as in 9.6.2. We require the following assumption, apart from the assumptions mentioned in 9.6.2.

($\tilde{D}1$) f_0 is semi-convex with modulus $\alpha_1 \in \mathbb{R}$.

($\tilde{D}2$) Let $\bar{x} \in \mathbb{R}^N$, then $g(F(\bar{x}) + \nabla F(\bar{x})(\,\cdot\, - \bar{x})$ is semi-convex with modulus $\alpha_2(\bar{x}) \leq 0$. Additionally, $\sup_{\bar{x} \in \mathbb{R}^N}(-\alpha_2(\bar{x})) < +\infty$.

($\tilde{D}3$) For all x_1, $x_2 \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, there exists $\gamma \in [0,1]$ such that $y := x_1 + \gamma(x_1 - x_2)$ lies in $\operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$.

Prox-Linear CoCaIn BPG is Model CoCaIn BPG (Algorithm 8) with

$$f(x; y_k) := f_0(x) + g(F(y_k) + \nabla F(y_k)(x - y_k))\,. \tag{10.5.2}$$

Using the Assumption ($\tilde{D}1$) and ($\tilde{D}2$), at $\bar{x} \in \operatorname{dom} f \cap \operatorname{int}\operatorname{dom} h$, we deduce that the semi-convexity modulus of $f(\,\cdot\,; \bar{x})$ is $\alpha_1 + \alpha_2(\bar{x})$. Thus, the Assumption L(i) holds true. Clearly, the Assumption ($\tilde{C}2$) implies the Assumption L(ii).

As discussed above and in Section 9.6.2, Assumptions (D1), (D2), (D3), ($\tilde{D}1$), ($\tilde{D}2$), ($\tilde{D}3$), (B1), (B2), (B3) imply Assumptions F, G, I, J, K. Thus, as a consequence of Theorem 10.4.4.1 we obtain the following result which provides the global convergence of the sequence generated by Prox-Linear CoCaIn BPG to a stationary point.

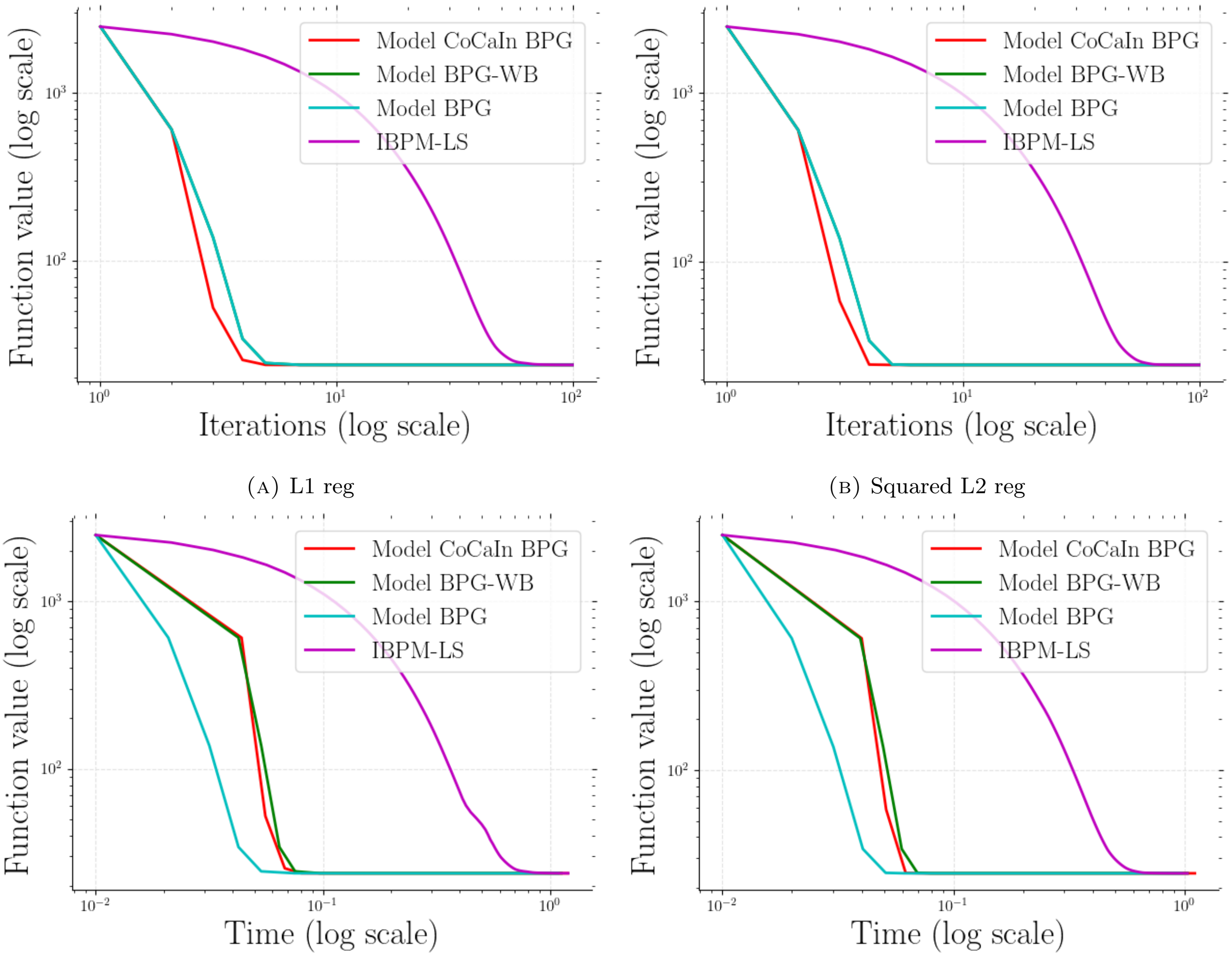
Theorem 10.5.2.1 (Global convergence of Prox-Linear CoCaIn BPG sequence)**.** *Let Assumptions (D1), (D2), (D3), ($\tilde{D}1$), ($\tilde{D}2$), ($\tilde{D}3$), (B1), (B2), (B3) hold. Let the sequence $(x_k)_{k\in\mathbb{N}}$ be generated by Prox-Linear CoCaIn BPG and the condition $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds true. Let $\tau_k \to \tau$ for certain $\tau > 0$. Then, the sequence $(x_k)_{k\in\mathbb{N}}$ has finite length, that is*

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| < +\infty\,,$$

and the sequence $(x_k)_{k\in\mathbb{N}}$ converges to x, which is a critical point of f.

10.6 Experiments

For the additive composite problems, the experiments in Chapter 5 illustrate that CoCaIn BPG is competitive to several existing state of the art optimization methods. For generic composite problems, we consider the same experiment as in Section 9.7.2 where we considered the robust phase retrieval problem. However, we additionally use Model CoCaIn BPG to the comparisons, and as evident from Figures 10.1, Model CoCaIn BPG outperforms Model BPG, Model BPG-WB and IBPM-LS methods, when function values vs iterations are considered.



(A) L1 reg

(B) Squared L2 reg

(C) L1 reg

(D) Squared L2 reg

FIGURE 10.1: In this experiment we consider the performance of Model CoCaIn BPG vs Model BPG vs Model BPG with Backtracking (denoted as Model BPG-WB) vs IBPM-LS [138] on robust phase retrieval problem setting given in Section 9.7.2, with both L1 and squared L2 regularization. We illustrate that Model CoCaIn BPG outperforms IBPM-LS by a significant margin and other methods by a small margin, in terms of function value vs iterations. However, when function value vs time plots are considered, Model BPG is faster compared to Model CoCaIn BPG.

10.7 Chapter conclusion

In this chapter, we propose an inertial variant of Model BPG algorithm, namely, Model CoCaIn BPG. The inertial strategy used is the same as that of CoCaIn BPG. For the purpose of global convergence analysis of Model CoCaIn BPG, a novel Lyapunov function is proposed and the global convergence guarantees are obtained. As a special case of Model CoCaIn BPG, in the context of generic composite problems we obtain a novel algorithm, namely, Prox-Linear CoCaIn BPG. We supplement our theoretical developments with appropriate experiments and show that Model CoCaIn BPG is competitive to Model BPG and other state of the art optimization methods. As a future work, extensions of Model BPG based on the popular inertial technique of Polyak's Heavy-ball method [145] can be explored.

Chapter 11

Conclusion and outlook

11.1 Conclusion

In this thesis, we focussed on non-convex and non-smooth optimization. Based on the L-smad property, a generalization of the Lipschitz continuous gradient property, we propose an inertial variant of the popular BPG algorithm, namely the CoCaIn BPG algorithm. We also show that the sequence generated by CoCaIn BPG is globally convergent. The applicability of the L-smad property is restricted to additive composite problems. In order to tackle composite problems, we develop an extension of the L-smad property, namely the MAP property. The transition from the L-smad property to the MAP property has many important consequences. In particular, one can design algorithms for generic composite problems using just the MAP property via the so-called model framework. We developed the relevant theory and propose the Model BPG algorithm based on the MAP property. We also show that under certain assumptions such that the Model BPG sequence is globally convergent to a critical point of the function. Later on, incorporating the ideas of CoCaIn BPG and Model BPG, we develop Model CoCaIn BPG that is essentially an inertial variant of Model BPG algorithm. Considering the practical component of the thesis, a major part of the thesis has been dedicated to the optimization of the objectives that arise in the context of matrix factorization, deep matrix factorization and deep non-linear neural networks. Other practical applications such Poisson linear inverse problems, standard phase retrieval, robust phase retrieval, image deblurring etc were also considered. For most of the above mentioned problems, we develop the relevant Bregman distances such that either the L-smad property or the MAP property holds true. Developing such Bregman distances entails further technical issues, such as developing closed form solutions for the sub-problems that arise in BPG based methods and developing closed form inertia for CoCaIn BPG. All such issues were successfully tackled. The Bregman distances proposed in this thesis have far reaching implications as they provide various insights into designing new Bregman distances to problems similar to what we have considered in this thesis. For example, Bregman distances for tensor factorization or tensor completion problems can be designed using similar ideas of Section 4.5, 4.6. We analysed the Model BPG algorithm via a Lyapunov function contrary to the usual technique of using the objective function as measure of progress. The exact implications of this technique need to be explored further and possibly several insights can be obtained for already known algorithms. We believe our work has several important consequences for the fields of Machine Learning, Computer Vision, Statistics, Natural Language Processing and many others. In particular, it would be interesting if certain future works in these fields use our algorithms and benchmark with already known state of the art problem-dependent algorithms.

11.2 Outlook

Our research can give rise to several new research problems. We list below a few of the them.

- **Non-linear inertia:** The inertial strategy that is used in CoCaIn BPG and Model CoCaIn BPG is based on the linear extrapolation due to Nesterov. However, it is relevant to consider any other strategies, where non-linear extrapolation can be incorporated. In particular, for the CoCaIn BPG algorithm in Chapter 5, the extrapolation parameter is found such that the following inequality is satisfied:
$$D_h(x^k, y^k) \leq \kappa D_h(x^{k-1}, x^k)\,. \tag{11.2.1}$$
for a constant $\kappa > 0$. However, for most of the proof of the global convergence of CoCaIn BPG, there is no requirement that y^k has to be a linear extrapolation. In principle, y^k can possibly obtained via some other strategy, such as non-linear extrapolation. Such strategies need to be explored in detail and possible implications need to be deduced.

- **Matrix factorization:** The Bregman distances we proposed in the context of matrix factorization in Section 4.5 arises after a lot of technical calculations. However, it is interesting to consider other approaches where such long calculations can be avoided. We relied on comparing the second order forms that arise in the Taylor expansion of the function in order to verify the L-smad property. In this regard, it would be interesting to explore if there are any other alternative techniques.

- **Deep non-linear neural networks:** We considered the objectives that arise in the context of deep non-linear neural networks and developed the relevant Bregman distances in Sections 4.7, 4.8. Our empirical observations in Chapter 8 are only preliminary, and it would be interesting to conduct a thorough comparisons of the algorithms on various deep neural networks, in particular using large scale datasets. In such a case, the numerical issues that arise in BPG methods might prove to be a challenge and novel techniques to resolves these issues must be sought. There are many other classes of deep non-linear neural networks, such as residual networks for which our theory is not valid. Hence, such extensions would be interesting and possibly a unified theory can be sought. Moreover, our analysis does not consider the effect of bias terms in the linear layers, and appropriate Bregman distances in such a realistic setting is interesting to be explored. The calculation of coefficients involved in the Legendre functions proposed in Sections 4.7, 4.8 is quite cumbersome and in this regard efficient techniques must be developed.

- **Stochastic and adaptive variants of BPG for deep neural networks:** It is well-established that the stochastic gradient methods are suitable for large-scale machine learning problems [32]. Several algorithms that rely on the so-called adaptive gradient technique, as in Adam [93], Adagrad [63], SC-Adagrad [119] are popular when used in conjunction with stochastic gradients. It is interesting to explore such a setting within the context of Bregman Proximal Gradient algorithms. Solving the subproblems can be challenging here as the stochastic gradient obtained is used along with a rotation matrix. Recently, a variant of stochastic BPG was proposed in [55]. It would be interesting to see if the Bregman distances proposed in this thesis are applicable in [55] setting. If not, suitable Bregman distances needs to be developed.

Appendix A

Appendix for Bregman distances - Chapter 4

A.1 Technical lemmas and proofs

Before we proceed to the proof of Proposition 4.5.0.1 we require the following technical lemma.

Lemma A.1.0.1. *Let* $f_1 := \frac{1}{2}\left\|A - UZ\right\|_F^2$*, then we have the following*

$$\nabla f_1(A, UZ) = \big(-(A-UZ)Z^T, -U^T(A-UZ)\big)$$

$$\big\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\big\rangle = -2\left\langle A - UZ, H_1H_2\right\rangle + \left\langle UH_2 + H_1Z, UH_2 + H_1Z\right\rangle\ .$$

Proof. With the Forbenius dot product, we have

$$\left\|A - UZ\right\|_F^2 = \left\langle A - UZ, A - UZ\right\rangle\ .$$

In the above expression by substituting U with $U + H_1$ and Z with $Z + H_2$, we obtain

$$\begin{aligned}
&\left\langle A - (U+H_1)(Z+H_2), A - (U+H_1)(Z+H_2)\right\rangle\ ,\\
&= \left\langle A - UZ - UH_2 - H_1Z - H_1H_2, A - UZ - UH_2 - H_1Z - H_1H_2\right\rangle\ ,\\
&= \left\langle A, A\right\rangle - \left\langle A, UZ\right\rangle - \left\langle A, UH_2\right\rangle - \left\langle A, H_1Z\right\rangle - \left\langle A, H_1H_2\right\rangle\ ,\\
&- \left\langle UZ, A\right\rangle + \left\langle UZ, UZ\right\rangle + \left\langle UZ, UH_2\right\rangle + \left\langle UZ, H_1Z\right\rangle + \left\langle UZ, H_1H_2\right\rangle\\
&- \left\langle UH_2, A\right\rangle + \left\langle UH_2, UZ\right\rangle + \left\langle UH_2, UH_2\right\rangle + \left\langle UH_2, H_1Z\right\rangle + \left\langle UH_2, H_1H_2\right\rangle\\
&- \left\langle H_1Z, A\right\rangle + \left\langle H_1Z, UZ\right\rangle + \left\langle H_1Z, UH_2\right\rangle + \left\langle H_1Z, H_1Z\right\rangle + \left\langle H_1Z, H_1H_2\right\rangle\\
&- \left\langle H_1H_2, A\right\rangle + \left\langle H_1H_2, UZ\right\rangle + \left\langle H_1H_2, UH_2\right\rangle + \left\langle H_1H_2, H_1Z\right\rangle + \left\langle H_1H_2, H_1H_2\right\rangle\ .
\end{aligned}$$

Collecting all the first order terms we have

$$\begin{aligned}
&- \langle A, UH_2\rangle - \langle A, H_1Z\rangle + \langle UZ, UH_2\rangle + \langle UZ, H_1Z\rangle \\
&- \langle UH_2, A\rangle + \langle UH_2, UZ\rangle - \langle H_1Z, A\rangle + \langle H_1Z, UZ\rangle \\
&= - \langle A, H_1Z\rangle + \langle UZ, H_1Z\rangle - \langle H_1Z, A\rangle + \langle H_1Z, UZ\rangle \\
&- \langle A, UH_2\rangle + \langle UZ, UH_2\rangle - \langle UH_2, A\rangle + \langle UH_2, UZ\rangle \ , \\
&= -2\langle A, H_1Z\rangle - 2\langle A, UH_2\rangle + 2\langle UZ, H_1Z\rangle + 2\langle UZ, UH_2\rangle \ , \\
&= -2tr((A - UZ)Z^TH_1^T) - 2tr((A - UZ)H_2^TU^T)\, , \\
&= -2tr((A - UZ)Z^TH_1^T) - 2tr(U^T(A - UZ)H_2^T)\, ,
\end{aligned}$$

and similarly collecting all the second order terms we have

$$\begin{aligned}
&- \langle A, H_1H_2\rangle + \langle UZ, H_1H_2\rangle + \langle UH_2, UH_2\rangle + \langle UH_2, H_1Z\rangle \\
&+ \langle H_1Z, UH_2\rangle + \langle H_1Z, H_1Z\rangle - \langle H_1H_2, A\rangle + \langle H_1H_2, UZ\rangle \\
&= -2\langle A - UZ, H_1H_2\rangle + \langle UH_2 + H_1Z, UH_2 + H_1Z\rangle \ .
\end{aligned}$$

Thus the statement follows using the second order Taylor expansion. □

Lemma A.1.0.2. *Given* $h_1 := \left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)^2$*, then we have the following*

$$\nabla h_1(U, Z) = \left(\left(\|U\|_F^2 + \|Z\|_F^2\right)U, \left(\|U\|_F^2 + \|Z\|_F^2\right)Z\right) \ ,$$

$$\left\langle (H_1, H_2), \nabla^2 h_1(U, Z)(H_1, H_2)\right\rangle = (\|H_1\|_F^2 + \|H_2\|_F^2)(\|U\|_F^2 + \|Z\|_F^2) + 2\left\|H_1U^T + ZH_2^T\right\|_F^2$$

Proof. By the definition of Forbenius dot product, we have

$$\frac{1}{4}\|U\|_F^4 + \frac{1}{4}\|Z\|_F^4 + \frac{1}{2}\|U\|_F^2\|Z\|_F^2 = \frac{1}{4}\langle U, U\rangle^2 + \frac{1}{4}\langle Z, Z\rangle^2 + \frac{1}{2}\langle U, U\rangle\langle Z, Z\rangle$$

Now, considering $h_1(U+H_1,Z+H_2)$ we have

$$\begin{aligned}
&\frac{1}{4}\left\langle U+H_1,U+H_1\right\rangle^2+\frac{1}{4}\left\langle Z+H_2,Z+H_2\right\rangle^2+\frac{1}{2}\left\langle U+H_1,U+H_1\right\rangle\left\langle Z+H_2,Z+H_2\right\rangle\\
&=\frac{1}{4}\left(\left\langle U,U\right\rangle+2\left\langle H_1,U\right\rangle+\left\langle H_1,H_1\right\rangle\right)^2+\frac{1}{4}\left(\left\langle Z,Z\right\rangle+2\left\langle Z,H_2\right\rangle+\left\langle H_2,H_2\right\rangle\right)^2\\
&+\frac{1}{2}\left(\left\langle U,U\right\rangle+2\left\langle H_1,U\right\rangle+\left\langle H_1,H_1\right\rangle\right)\left(\left\langle Z,Z\right\rangle+2\left\langle Z,H_2\right\rangle+\left\langle H_2,H_2\right\rangle\right)\\
&=\frac{1}{4}\Big(\left\langle U,U\right\rangle^2+4\left\langle H_1,U\right\rangle^2+\left\langle H_1,H_1\right\rangle^2+2\left\langle H_1,H_1\right\rangle\left\langle U,U\right\rangle\\
&+4\left\langle U,U\right\rangle\left\langle H_1,U\right\rangle+4\left\langle H_1,U\right\rangle\left\langle H_1,H_1\right\rangle)\\
&+\frac{1}{4}\Big(\left\langle Z,Z\right\rangle^2+4\left\langle Z,H_2\right\rangle^2+\left\langle H_2,H_2\right\rangle^2+2\left\langle H_2,H_2\right\rangle\left\langle Z,Z\right\rangle\\
&+4\left\langle Z,H_2\right\rangle\left\langle Z,Z\right\rangle+4\left\langle Z,H_2\right\rangle\left\langle H_2,H_2\right\rangle)\\
&+\frac{1}{2}\left(\left\langle U,U\right\rangle\left\langle Z,Z\right\rangle+2\left\langle U,U\right\rangle\left\langle Z,H_2\right\rangle+\left\langle U,U\right\rangle\left\langle H_2,H_2\right\rangle\right)\\
&+\frac{1}{2}\left(2\left\langle H_1,U\right\rangle\left\langle Z,Z\right\rangle+4\left\langle H_1,U\right\rangle\left\langle Z,H_2\right\rangle+2\left\langle H_1,U\right\rangle\left\langle H_2,H_2\right\rangle\right)\\
&+\frac{1}{2}\left(\left\langle H_1,H_1\right\rangle\left\langle Z,Z\right\rangle+2\left\langle H_1,H_1\right\rangle\left\langle Z,H_2\right\rangle+\left\langle H_1,H_1\right\rangle\left\langle H_2,H_2\right\rangle\right)
\end{aligned}$$

Collecting all the first order terms, we have

$$\left\langle U,U\right\rangle\left\langle H_1,U\right\rangle+\left\langle Z,H_2\right\rangle\left\langle Z,Z\right\rangle+\left\langle U,U\right\rangle\left\langle Z,H_2\right\rangle+\left\langle H_1,U\right\rangle\left\langle Z,Z\right\rangle\ ,$$

and similarly collecting all the second order terms we have

$$\begin{aligned}
&\frac{1}{4}\Big(4\left\langle H_1,U\right\rangle^2+2\left\langle H_1,H_1\right\rangle\left\langle U,U\right\rangle+4\left\langle Z,H_2\right\rangle^2+2\left\langle H_2,H_2\right\rangle\left\langle Z,Z\right\rangle\Big)\\
&+\frac{1}{2}\left(\left\langle U,U\right\rangle\left\langle H_2,H_2\right\rangle+4\left\langle H_1,U\right\rangle\left\langle Z,H_2\right\rangle+\left\langle H_1,H_1\right\rangle\left\langle Z,Z\right\rangle\right)\ ,\\
&=\frac{1}{2}\Big(2\left\langle H_1,U\right\rangle^2+(\left\langle H_1,H_1\right\rangle+\left\langle H_2,H_2\right\rangle)(\left\langle U,U\right\rangle+\left\langle Z,Z\right\rangle)\\
&+2\left\langle Z,H_2\right\rangle^2+4\left\langle H_1,U\right\rangle\left\langle Z,H_2\right\rangle\Big)\ ,\\
&=\frac{1}{2}\left((\left\langle H_1,H_1\right\rangle+\left\langle H_2,H_2\right\rangle)(\left\langle U,U\right\rangle+\left\langle Z,Z\right\rangle)+2(\left\langle H_1,U\right\rangle+\left\langle Z,H_2\right\rangle)^2\right)\ .
\end{aligned}$$

Thus the statement follows. □

Lemma A.1.0.3. *Given $h_2(U,Z):=\frac{\|U\|_F^2+\|Z\|_F^2}{2}$, then we have the following*

$$\nabla h_2(U,Z)=(U,Z)\,,$$

$$\left\langle (H_1,H_2),\nabla^2 h_2(U,Z)(H_1,H_2)\right\rangle=\|H_1\|_F^2+\|H_2\|_F^2\ .$$

Proof. Considering $h_2(U+H_1,Z+H_2)$, we have

$$\begin{aligned}
&\frac{1}{2}\left\langle U+H_1,U+H_1\right\rangle+\frac{1}{2}\left\langle Z+H_2,Z+H_2\right\rangle\\
&=\frac{1}{2}\left(\left\langle U,U\right\rangle+2\left\langle U,H_1\right\rangle+\left\langle H_1,H_1\right\rangle\right)+\frac{1}{2}\left(\left\langle Z,Z\right\rangle+2\left\langle Z,H_2\right\rangle+\left\langle H_2,H_2\right\rangle\right)\ .
\end{aligned}$$

Collecting all the first order terms we have

$$\langle U, H_1\rangle + \langle Z, H_2\rangle \ ,$$

and similarly collecting all the second order terms we have

$$\frac{1}{2}\left(\langle H_1, H_1\rangle + \langle H_2, H_2\rangle\right) \ .$$

Thus the statement holds. □

A.2 Proof of Proposition 4.5.0.1

Proof. We prove here the convexity of $Lh_a - f_1$ for a certain constant $L \geq 1$. With Lemma C.4.0.1 we obtain

$$\begin{aligned}
&\big\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\big\rangle \\
&= \|H_1 Z + U H_2\|_F^2 - 2\,\langle A - UZ, H_1 H_2\rangle \ , \\
&\leq 2\,\|H_1 Z\|_F^2 + 2\,\|U H_2\|_F^2 + 2\,\|A\|_F\,\|H_1 H_2\|_F + 2\,\|UZ\|_F\,\|H_1 H_2\|_F \ , \\
&\leq 2\,\|H_1\|_F^2\,\|Z\|_F^2 + 2\,\|U\|_F^2\,\|H_2\|_F^2 + 2\,\|A\|_F\,\|H_1\|_F\,\|H_2\|_F + 2\,\|U\|_F\,\|Z\|_F\,\|H_1\|_F\,\|H_2\|_F \ .
\end{aligned}$$

With AM-GM inequality, for non-negative real numbers a, b we have $2\sqrt{ab} \leq a + b$, we have

$$2\,\|U\|_F\,\|Z\|_F\,\|H_1\|_F\,\|H_2\|_F \leq \|H_1\|_F^2\,\|Z\|_F^2 + \|U\|_F^2\,\|H_2\|_F^2 \ ,$$

and similarly we have

$$2\,\|A\|_F\,\|H_1\|_F\,\|H_2\|_F \leq \|A\|_F\,\|H_1\|_F^2 + \|A\|_F\,\|H_2\|_F^2 \ .$$

Using the above two inequalities, we obtain

$$\big\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\big\rangle \leq \left(3\,\|Z\|_F^2 + \|A\|_F\right)\|H_1\|_F^2 + \left(3\,\|U\|_F^2 + \|A\|_F\right)\|H_2\|_F^2 \ . \tag{A.2.1}$$

Now, considering the kernel generating distances, via Lemma D.3.2.1 and A.1.0.3 we obtain

$$\begin{aligned}
&\big\langle (H_1, H_2), \nabla^2 h_1(U, Z)(H_1, H_2)\big\rangle \\
&= 2\,\|H_1 U + H_2 Z\|_F^2 + \left(\|U\|_F^2 + \|Z\|_F^2\right)\|H_1\|_F^2 + \left(\|U\|_F^2 + \|Z\|_F^2\right)\|H_2\|_F^2 \\
&\geq \|Z\|_F^2\,\|H_1\|_F^2 + \|U\|_F^2\,\|H_2\|_F^2 \ ,
\end{aligned}$$

$$\big\langle (H_1, H_2), \nabla^2 h_2(U, Z)(H_1, H_2)\big\rangle = \|H_1\|_F^2 + \|H_2\|_F^2 \ .$$

Now, it is easy to see that

$$\big\langle (H_1, H_2), \nabla^2 h_a(U, Z)(H_1, H_2)\big\rangle \geq \big\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\big\rangle \ .$$

A similar proof holds for the convexity of $Lh_a + f_1$, however the choice of L here need not be the same as it is for $Lh_a - f_1$ (see [28, Remark 2.1]). □

A.3 Bregman distance and L-smad property

Proposition A.3.0.1. *Denote $f_1(W_1, \ldots, W_N) := \frac{1}{2}\left\|W_1 W_2 \ldots W_N X - Y\right\|_F^2$ as in the setting of (4.6.1). Then the gradient with respect to weights W_i is given by*

$$\nabla_{W_i} f_1(W_1, \ldots, W_N) = \left(\Pi_{j=1}^{i-1} W_j\right)^T (W_1 W_2 \ldots W_N X - Y)\left(\left(\Pi_{j=i+1}^{N} W_j\right) X\right)^T .$$

We have for $N = 2$,

$$\begin{aligned}&\left\langle (H_1, \ldots, H_N), \nabla^2 f_1(W_1, \ldots, W_N)(H_1, \ldots, H_N)\right\rangle \\ &\leq 3\left\|X\right\|_F^2 \sum_{i=1}^{N} \left\|H_i\right\|_F^2 \Pi_{j=1, j\neq i}^{N} \left\|W_j\right\|_F^2 + \left\|Y\right\|_F \left\|X\right\|_F \left(\left\|H_1\right\|_F^2 + \left\|H_2\right\|_F^2\right)\end{aligned}$$

If $N > 2$ and even, we have

$$\begin{aligned}&\left\langle (H_1, \ldots, H_N), \nabla^2 f_1(W_1, \ldots, W_N)(H_1, \ldots, H_N)\right\rangle \\ &\leq (2N-1)\sum_{i=1}^{N} \left\|H_i\right\|_F^2 \Pi_{j=1, j\neq i}^{N} \left\|W_j\right\|_F^2 \left\|X\right\|_F^2 + \frac{\left\|Y\right\|_F \left\|X\right\|_F (N-1)}{(N-2)^{\frac{N-2}{2}}}\left(\sum_{i=1}^{N}\left\|H_i\right\|_F^2\right)\left(\sum_{k=1}^{N}\left\|W_k\right\|_F^2\right)^{\frac{N-2}{2}}\end{aligned}$$

If $N > 2$ and odd, we have

$$\begin{aligned}&\left\langle (H_1, \ldots, H_N), \nabla^2 f_1(W_1, \ldots, W_N)(H_1, \ldots, H_N)\right\rangle \\ &\leq (2N-1)\sum_{i=1}^{N} \left\|H_i\right\|_F^2 \Pi_{j=1, j\neq i}^{N} \left\|W_j\right\|_F^2 \left\|X\right\|_F^2 \\ &+ \frac{\left\|Y\right\|_F \left\|X\right\|_F (N-1)}{(N-1)^{\frac{N-1}{2}}}\left(\sum_{i=1}^{N}\left\|H_i\right\|_F^2\right)\left(\left(\sum_{k=1, k\notin\{i,j\}}^{N}\left\|W_k\right\|_F^2\right) + 1\right)^{\frac{N-1}{2}}\end{aligned}$$

Proof. Consider the following

$$\frac{1}{2}\left\|(W_1 + H_1)(W_2 + H_2)\ldots(W_N + H_N)X - Y\right\|_F^2 . \tag{A.3.1}$$

We are only interested in terms till second order, thus we have

$$\begin{aligned}(W_1 + H_1)(W_2 + H_2)\ldots(W_N + H_N)X = W_1 W_2 \ldots W_N X &+ \sum_{i=1}^{N}\left(\Pi_{j=1}^{i-1} W_j\right) H_i \left(\Pi_{j=i+1}^{N} W_j X\right) \\ &+ \sum_{i=1}^{N-1}\sum_{j>i}\left(\Pi_{k=1}^{i-1} W_k\right) H_i \left(\Pi_{k=i+1}^{j-1} W_k\right) H_j \left(\Pi_{k=j+1}^{N} W_k X\right) .\end{aligned}$$

Now expanding (A.3.1), we have terms upto second order as following

$$\begin{aligned}
&\frac{1}{2}\left\|W_1W_2\dots W_NX-Y\right\|_F^2\\
&+\left\langle W_1W_2\dots W_NX-Y,\sum_{i=1}^{N}\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\rangle\\
&+\frac{1}{2}\left\|\sum_{i=1}^{N}\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\|_F^2\\
&-\left\langle Y,\sum_{i=1}^{N-1}\sum_{j>i}^{N}\left(\Pi_{k=1}^{i-1}W_k\right)H_i\left(\Pi_{k=i+1}^{j-1}W_k\right)H_j\left(\Pi_{k=j+1}^{N}W_k\right)X\right\rangle\\
&+\left\langle W_1W_2\dots W_NX,\sum_{i=1}^{N-1}\sum_{j>i}^{N}\left(\Pi_{k=1}^{i-1}W_k\right)H_i\left(\Pi_{k=i+1}^{j-1}W_k\right)H_j\left(\Pi_{k=j+1}^{N}W_k\right)X\right\rangle\,.
\end{aligned}$$

Consider the first order terms, we have

$$\begin{aligned}
&\left\langle W_1W_2\dots W_NX-Y,\sum_{i=1}^{N}\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\rangle\\
&=\sum_{i=1}^{N}\left\langle W_1W_2\dots W_NX-Y,\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\rangle\,,
\end{aligned}$$

thus, the gradient is

$$\nabla_{W_i}f_1(W_1,\dots,W_N)=\left(\Pi_{j=1}^{i-1}W_j\right)^T\left(W_1W_2\dots W_NX-Y\right)\left(\left(\Pi_{j=i+1}^{N}W_j\right)X\right)^T\,.$$

Now, considering second order terms we have with repetitive application of Cauchy-Schwarz inequality, the following

$$\begin{aligned}
\frac{1}{2}\left\|\sum_{i=1}^{N}\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\|_F^2 &\leq\frac{N}{2}\sum_{i=1}^{N}\left\|\left(\Pi_{j=1}^{i-1}W_j\right)H_i\left(\Pi_{j=i+1}^{N}W_j\right)X\right\|_F^2\\
&\leq\frac{N}{2}\sum_{i=1}^{N}\left\|H_i\right\|_F^2\Pi_{j=1,j\neq i}^{N}\left\|W_j\right\|_F^2\left\|X\right\|_F^2
\end{aligned}$$

and

$$\left\langle W_1W_2\dots W_NX, \sum_{i=1}^{N-1}\sum_{j>i}^{N}\left(\Pi_{k=1}^{i-1}W_k\right)H_i\left(\Pi_{k=i+1}^{j-1}W_k\right)H_j\left(\Pi_{k=j+1}^{N}W_k\right)X\right\rangle$$

$$\leq \sum_{i=1}^{N-1}\sum_{j>i}^{N}\|X\|_F^2\|H_i\|_F\|H_j\|_F\|W_i\|_F\|W_j\|_F\,\Pi_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F^2$$

$$\leq \sum_{i=1}^{N-1}\sum_{j>i}^{N}\|X\|_F^2\left(\frac{\|H_i\|_F^2\|W_j\|_F^2+\|H_j\|_F^2\|W_i\|_F^2}{2}\right)\Pi_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F^2$$

$$\leq \|X\|_F^2\left(\frac{N-1}{2}\right)\sum_{i=1}^{N}\|H_i\|_F^2\,\Pi_{k=1,k\notin\{i\}}^{N}\|W_k\|_F^2$$

and we have

$$-\left\langle Y, \sum_{i=1}^{N-1}\sum_{j>i}^{N}\left(\Pi_{k=1}^{i-1}W_k\right)H_i\left(\Pi_{k=i+1}^{j-1}W_k\right)H_j\left(\Pi_{k=j+1}^{N}W_k\right)X\right\rangle$$

$$\leq \|Y\|_F\sum_{i=1}^{N-1}\sum_{j>i}^{N}\|H_i\|_F\|H_j\|_F\,\Pi_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F\|X\|_F \tag{A.3.2}$$

Now with the application of Generalized AM-GM inequality, we have the following three cases:

- When $N=2$ then we have

$$\|H_i\|_F\|H_j\|_F\|X\|_F \leq \|X\|_F\left(\frac{\|H_j\|_F^2+\|H_i\|_F^2}{2}\right),$$

- When N is even and $N>2$.

$$\|H_i\|_F\|H_j\|_F\,\Pi_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F\|X\|_F \leq \|X\|_F\left(\frac{\|H_j\|_F^2+\|H_i\|_F^2}{2}\right)\left(\frac{\sum_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F^2}{N-2}\right)^{\frac{N-2}{2}},$$

- If N is odd and $N>2$ we have

$$\|H_i\|_F\|H_j\|_F\,\Pi_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F\|X\|_F \leq \|X\|_F\left(\frac{\|H_j\|_F^2+\|H_i\|_F^2}{2}\right)\left(\frac{\left(\sum_{k=1,k\notin\{i,j\}}^{N}\|W_k\|_F^2\right)+1}{N-1}\right)^{\frac{N-1}{2}}.$$

Now using the above given results, on extending the calculation of (A.3.2), for even N and $N \geq 2$, we have

$$\begin{aligned}
&\|Y\|_F \sum_{i=1}^{N-1}\sum_{j>i}^{N} \|H_i\|_F \|H_j\|_F \Pi_{k=1,k\notin\{i,j\}}^{N} \|W_k\|_F \|X\|_F \\
&\leq \|Y\|_F \|X\|_F \sum_{i=1}^{N-1}\sum_{j>i}^{N} \left(\frac{\|H_j\|_F^2 + \|H_i\|_F^2}{2}\right)\left(\frac{\sum_{k=1,k\notin\{i,j\}}^{N} \|W_k\|_F^2}{N-2}\right)^{\frac{N-2}{2}} \\
&\leq \frac{\|Y\|_F \|X\|_F (N-1)}{2(N-2)^{\frac{N-2}{2}}}\left(\sum_{i=1}^{N} \|H_i\|_F^2\right)\left(\sum_{k=1}^{N} \|W_k\|_F^2\right)^{\frac{N-2}{2}} ,
\end{aligned}$$

where in the first step we used Cauchy-Schwarz inequality. Similarly, we have for $N > 2$ and odd,

$$\begin{aligned}
&\|Y\|_F \sum_{i=1}^{N-1}\sum_{j>i}^{N} \|H_i\|_F \|H_j\|_F \Pi_{k=1,k\notin\{i,j\}}^{N} \|W_k\|_F \|X\|_F \\
&\leq \|Y\|_F \|X\|_F \sum_{i=1}^{N-1}\sum_{j>i}^{N} \left(\frac{\|H_j\|_F^2 + \|H_i\|_F^2}{2}\right)\left(\frac{\left(\sum_{k=1,k\notin\{i,j\}}^{N} \|W_k\|_F^2\right)+1}{N-1}\right)^{\frac{N-1}{2}} \\
&\leq \frac{\|Y\|_F \|X\|_F (N-1)}{2(N-1)^{\frac{N-1}{2}}}\left(\sum_{i=1}^{N} \|H_i\|_F^2\right)\left(\left(\sum_{k=1}^{N} \|W_k\|_F^2\right)+1\right)^{\frac{N-1}{2}} .
\end{aligned}$$

□

Before we start with the proof of Proposition 4.6.0.1, we require the following technical results.

Lemma A.3.0.1. *Let $h \in \mathcal{G}(C)$ be twice continuously differentiable on C. Then, the following identity holds*

$$D_h(x^k, y^k) = \int_0^1 (1-t)\int_0^1 \left\langle \nabla^2 h\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - y^k), x^k - y^k\right\rangle dt_1 dt\,.$$

Proof. With repetitive application of fundamental theorem of calculus we have

$$\begin{aligned}
&h(x^k) - h(y^k) - \left\langle \nabla h(y^k), x^k - y^k\right\rangle \\
&= \int_0^1 \left\langle \nabla h(x^k + t(y^k - x^k)) - \nabla h(y^k), x^k - y^k\right\rangle dt\,, \\
&= \int_0^1 \left\langle \int_0^1 \nabla^2 h\left((1-t_1)(x^k + t(y^k - x^k)) + t_1 y^k\right)(1-t)\,(x^k - y^k) dt_1, x^k - y^k\right\rangle dt\,, \\
&= \int_0^1 \left\langle \int_0^1 \nabla^2 h\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(1-t)\,(x^k - y^k) dt_1, x^k - y^k\right\rangle dt\,, \\
&= \int_0^1 (1-t)\int_0^1 \left\langle \nabla^2 h\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - y^k), x^k - y^k\right\rangle dt_1 dt\,.
\end{aligned}$$

□

Henceforth, we use the following notation. Let n be a positive integer and let k_i be a non-negative integer for $i \in \{1, \ldots, m\}$ satisfying $k_1 + \ldots + k_m = n$, then we denote

$$\binom{n}{k_1, k_2, \ldots, k_m} := \frac{n!}{k_1!k_2! \ldots k_m!} \,,$$

which is also known as multinomial coefficient.

Lemma A.3.0.2. *With the following kernel generating distance*

$$H_1(W_1, \ldots, W_N) = \left(\frac{\|W_1\|_F^2 + \ldots \|W_N\|_F^2}{N} \right)^N \,,$$

the gradient with respect for W_i, for any $i \in \{1, \ldots, N\}$, is given by

$$\nabla_{W_i} H_1(W_1, \ldots, W_N) = \frac{2}{N^N} \binom{N}{N-1, 1} \left(\|W_1\|_F^2 + \ldots + \|W_N\|_F^2 \right)^{N-1} W_i \,,$$

and the following lower bound holds true

$$\left\langle (H_1, \ldots, H_N), \nabla^2 H_1(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \frac{2N!}{N^N} \sum_{i=1}^N \|H_i\|_F^2 \, \Pi_{k=1, k \notin \{i\}}^N \|W_k\|_F^2 \,,$$

and the following upper bound holds true

$$\left\langle (H_1, \ldots, H_N), \nabla^2 H_1(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \leq \left(\frac{2(2N-1)}{N^{N-1}} \right) \left(\sum_{k=1}^N \|H_k\|_F^2 \right) \left(\sum_{k=1}^N \|W_k\|_F^2 \right)^{N-1} \,.$$

Proof. Consider the following

$$\left(\frac{\|W_1 + H_1\|_F^2 + \ldots \|W_N + H_N\|_F^2}{N} \right)^N = \left(\frac{\|W_1\|_F^2 + \|H_1\|_F^2 + 2\langle W_1, H_1 \rangle + \ldots \|W_N\|_F^2 + \|H_N\|_F^2}{N} + \ldots \right)^N \,.$$

Consider only the first order terms in the expansion, from which the following gradient with respect for W_i, for any $i \in \{1, \ldots, N\}$, is obtained

$$\nabla_{W_i} H_1(W_1, \ldots, W_N) = \frac{2}{N^N} \binom{N}{N-1, 1} \left(\|W_1\|_F^2 + \ldots + \|W_N\|_F^2 \right)^{N-1} W_i \,.$$

Now considering only the second order terms, we have

$$\begin{aligned}
&\frac{1}{2} \left\langle (H_1, \ldots, H_N), \nabla^2 H_1(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \\
&= \frac{1}{2} \frac{2}{N^N} \sum_{i=1}^N \binom{N}{1, N-1} \|H_i\|_F^2 \left(\sum_{k=1}^N \|W_k\|_F^2 \right)^{N-1} \\
&+ \frac{1}{2} \frac{2^3}{N^N} \binom{N}{2, N-2} \left(\langle W_1, H_1 \rangle + \ldots + \langle W_N, H_N \rangle \right)^2 \left(\sum_{k=1}^N \|W_k\|_F^2 \right)^{N-2} \,.
\end{aligned}$$

Since, the second term in the right hand side is always non-negative, the following result holds

$$\frac{1}{2}\left\langle (H_1,\ldots,H_N), \nabla^2 H_1(W_1,\ldots,W_N)(H_1,\ldots,H_N)\right\rangle \geq \frac{1}{2}\frac{2N!}{N^N}\sum_{i=1}^{N} \|H_i\|_F^2\, \Pi_{k=1,k\notin\{i\}}^{N} \|W_k\|_F^2 \,.$$

This proves the lower bound. Now, we prove the upper bound. With application of Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&\frac{1}{2}\frac{2^3}{N^N}\binom{N}{2, N-2}\left(\langle W_1, H_1\rangle + \ldots + \langle W_N, H_N\rangle\right)^2 \left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-2}\\
&\leq \frac{1}{2}\frac{2^3}{N^N}\binom{N}{2, N-2}\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)\left(\sum_{k=1}^{N}\|H_k\|_F^2\right)\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-2}\\
&= \frac{1}{2}\frac{2^3}{N^N}\binom{N}{2, N-2}\left(\sum_{k=1}^{N}\|H_k\|_F^2\right)\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-1} .
\end{aligned}$$

Now we finally have

$$\begin{aligned}
\left\langle (H_1,\ldots,H_N), \nabla^2 H_1(W_1,\ldots,W_N)(H_1,\ldots,H_N)\right\rangle &\leq \frac{2}{N^N}\binom{N}{1, N-1}\left(\sum_{i=1}^{N}\|H_i\|_F^2\right)\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-1}\\
&\quad + \frac{2^3}{N^N}\binom{N}{2, N-2}\left(\sum_{k=1}^{N}\|H_k\|_F^2\right)\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-1}\\
&= \left(\frac{2(2N-1)}{N^{N-1}}\right)\left(\sum_{k=1}^{N}\|H_k\|_F^2\right)\left(\sum_{k=1}^{N}\|W_k\|_F^2\right)^{N-1} . \square
\end{aligned}$$

Lemma A.3.0.3. *Denote for any $k \geq 1$, $x^k = (W_1^k, \ldots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{B}_k := \left(\frac{(2N-1)}{N^{N-1}}\right)\|\Delta_k\|^2\left(2\left\|x^k\right\|^2 + 2\left\|\Delta_k\right\|^2\right)^{(N-1)} .$$

The following upper bound holds true

$$D_{H_1}(x^k, y^k) \leq \gamma_k^2 \mathcal{B}_k \,.$$

Proof. From Lemma A.3.0.1, we have

$$\begin{aligned}
&\int_0^1 (1-t)\int_0^1 \left\langle \nabla^2 H_1\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - y^k), x^k - y^k\right\rangle dt_1 dt\\
=&\gamma_k^2\int_0^1 (1-t)\int_0^1 \left\langle \nabla^2 H_1\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - x^{k-1}), x^k - x^{k-1}\right\rangle dt_1 dt\,,\\
\leq&\gamma_k^2\int_0^1 (1-t)\int_0^1 \frac{2(2N-1)}{N^{N-1}}\left\|x^k - x^{k-1}\right\|^2\left\|x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right\|^{(2N-2)} dt_1 dt\,,
\end{aligned}$$

where in the last step we used the upper bound from Lemma A.3.0.2. Using the following inequality

$$\left\|x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right\|^2 \leq 2\left\|x^k\right\|^2 + 2(t_1 + (1-t_1)t)^2\gamma_k^2\left\|x^k - x^{k-1}\right\|^2 \leq 2\left\|x^k\right\|^2 + 2\left\|x^k - x^{k-1}\right\|^2$$

where in the last step we used $\gamma_k^2 \leq 1$ and $(t_1 + (1-t_1)t)^2 \leq 1$. With $\int_0^1 (1-t)dt = \frac{1}{2}$ the result follows. □

Lemma A.3.0.4. *With the following kernel generating distance*

$$H_2(W_1, \ldots, W_N) = \left(\frac{\|W_1\|_F^2 + \|W_2\|_F^2 + \ldots \|W_N\|_F^2}{N} \right)^{\frac{N}{2}},$$

the gradient with respect for W_i, for any $i \in \{1, \ldots, N\}$, is given by

$$\nabla_{W_i} H_2(W_1, \ldots, W_N) = \frac{1}{N^{\frac{N}{2}-1}} \left(\|W_1\|_F^2 + \ldots + \|W_N\|_F^2 \right)^{\frac{N}{2}-1} W_i \,,$$

and the following lower bound holds true

$$\left\langle (H_1, \ldots, H_N), \nabla^2 H_2(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \frac{1}{N^{\frac{N}{2}-1}} \left(\|H_1\|_F^2 + \ldots + \|H_N\|_F^2 \right) \left(\sum_{k=1}^{N} \|W_k\|_F^2 \right)^{\frac{N-2}{2}},$$

and the following upper bound holds true

$$\left\langle (H_1, \ldots, H_N), \nabla^2 H_2(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \leq \left(\frac{N-1}{N^{\frac{N}{2}-1}} \right) \left(\sum_{k=1}^{N} \|H_k\|_F^2 \right) \left(\sum_{k=1}^{N} \|W_k\|_F^2 \right)^{\frac{N-2}{2}}.$$

The proof of Lemma A.3.0.4 is similar to the proof of Lemma A.3.0.2, thus we skip the details for brevity.

Lemma A.3.0.5. *Denote for any $k \geq 1$, $x^k = (W_1^k, \ldots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{C}_k := \left(\frac{N-1}{N^{\frac{N}{2}-1}} \right) \|\Delta_k\|^2 \left(2 \left\| x^k \right\|^2 + 2 \|\Delta\|^2 \right)^{\frac{N-2}{2}}.$$

The following holds

$$D_{H_2}(x^k, y^k) \leq \gamma_k^2 \mathcal{C}_k \,.$$

The proof of Lemma A.3.0.5 is similar to the proof of Lemma A.3.0.3, thus we skip the details for brevity.

A.3.1 Proof of Proposition 4.6.0.1

We need to prove the convexity of $LH_a - g$. From Lemma A.3.0.2 we obtain

$$\frac{N^N}{2N!} \left\langle (H_1, \ldots, H_N), \nabla^2 H_1(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \sum_{i=1}^{N} \|H_i\|_F^2 \, \Pi_{k=1, k \notin \{i,j\}}^{N} \|W_k\|_F^2$$

Similarly from Lemma A.3.0.4 we obtain

$$\frac{N^{\frac{N}{2}}}{2\binom{\frac{N}{2}}{\frac{N-2}{2},1}} \left\langle (H_1, \ldots, H_N), \nabla^2 H_2(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \left(\|H_1\|_F^2 + \ldots + \|H_N\|_F^2 \right) \left(\sum_{k=1}^{N} \|W_k\|_F^2 \right)^{\frac{N-2}{2}}$$

Thus, now invoking Proposition A.3.0.1, we obtain the result. □

A.3.2 Results for H_3.

Lemma A.3.2.1. *With the following kernel generating distance*

$$H_3(W_1, \ldots, W_N) = \left(\frac{\|W_1\|_F^2 + \|W_2\|_F^2 + \ldots \|W_N\|_F^2 + 1}{N+1} \right)^{\frac{N+1}{2}},$$

the gradient with respect for W_i, for any $i \in \{1, \ldots, N\}$, is given by

$$\nabla_{W_i} H_3(W_1, \ldots, W_N) = \frac{2\binom{\frac{N+1}{2}}{\frac{N-1}{2}, 1}}{(N+1)^{\frac{N+1}{2}}} \left(\|W_1\|_F^2 + \ldots + \|W_N\|_F^2 + 1 \right)^{\frac{N-1}{2}} W_i\,,$$

and the following lower bound holds true

$$\begin{aligned} &\left\langle (H_1, \ldots, H_N), \nabla^2 H_3(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \\ &\geq \frac{2}{(N+1)^{\frac{N+1}{2}}} \binom{\frac{N+1}{2}}{\frac{N-1}{2}, 1} \left(\|H_1\|_F^2 + \ldots + \|H_N\|_F^2 \right) \left(\left(\sum_{k=1}^{N} \|W_k\|_F^2 \right) + 1 \right)^{\frac{N-1}{2}}, \end{aligned}$$

and the following upper bound holds true

$$\begin{aligned} &\left\langle (H_1, \ldots, H_N), \nabla^2 H_3(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \\ &\leq \frac{N}{(N+1)^{\frac{N-1}{2}}} \left(\sum_{k=1}^{N} \|H_k\|_F^2 \right)^2 \left(\left(\sum_{k=1}^{N} \|W_k\|_F^2 \right) + 1 \right)^{\frac{N-1}{2}}. \end{aligned}$$

The proof of Lemma A.3.2.1 is similar to the proof of Lemma A.3.0.2, thus we skip the details for brevity.

Lemma A.3.2.2. *Denote for any $k \geq 1$, $x^k = (W_1^k, \ldots, W_N^k)$, $\Delta_k := x^k - x^{k-1}$ and the following*

$$\mathcal{D}_k := \frac{N}{(N+1)^{\frac{N-1}{2}}} \|\Delta_k\|^2 \left(2 \left\| x^k \right\|^2 + 2 \|\Delta\|^2 + 1 \right)^{\frac{N-1}{2}}.$$

Then, the condition $D_{H_3}(x^k, y^k) \leq \gamma_k^2 \mathcal{D}_k$ holds true.

The proof of Lemma A.3.2.2 is similar to the proof of Lemma A.3.0.3, thus we skip the details for brevity.

A.3.3 Proof of Proposition 4.6.0.2.

We need to prove the convexity of $LH_b - g$. From Lemma A.3.0.2 we obtain

$$\frac{N^N}{2N!} \left\langle (H_1, \ldots, H_N), \nabla^2 H_1(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \geq \sum_{i=1}^{N} \|H_i\|_F^2 \,\Pi_{k=1, k \notin \{i,j\}}^{N} \|W_k\|_F^2$$

Similarly, from Lemma A.3.2.1 we obtain

$$(N+1)^{\frac{N-1}{2}} \left\langle (H_1, \ldots, H_N), \nabla^2 H_3(W_1, \ldots, W_N)(H_1, \ldots, H_N) \right\rangle \left(\sum_{i=1}^{N} \|H_i\|_F^2 \right) \left(\left(\sum_{k=1}^{N} \|W_k\|_F^2 \right) + 1 \right)^{\frac{N-1}{2}}$$

and invoking Proposition A.3.0.1, we obtain the result. The proof of $LH_b + g$ is similar (see Remark 2.1 of [28]). □

A.4 Proof of Lemma 4.7.2.2

Proof. In the expansion $S_N(W_1 + H_1, \dots, W_N + H_N)$, in order to obtain the first order term containing H_i, we can set the other H_i terms to zero and perform the expansion. In that case, we obtain the following first order term containing H_i:

$$\begin{aligned} \Delta_{i,N} &= \sigma_N'(W_N S_{N-1}) \circ W_N \Delta_{i,N-1}\,, \\ \Delta_{i,N-1} &= \sigma_{N-1}'(W_{N-1} S_{N-2}) \circ W_{N-1} \Delta_{i,N-2}\,, \\ &\dots\,, \\ \Delta_{i,i} &= \sigma_i'(W_i S_{i-1}) \circ H_i S_{i-1}\,. \end{aligned} \tag{A.4.1}$$

Here, the only technique we have applied is the first order Taylor expansion as in Lemma 4.7.2.1. Similarly, for the second order term which couples H_i, H_j where $i \neq j$ is the following:

$$\begin{aligned} \Delta_{i,j,N} &= \sigma_N'(W_N S_{N-1}) \circ W_N \Delta_{i,j,N-1}\,, \\ \Delta_{i,j,N-1} &= \sigma_{N-1}'(W_{N-1} S_{N-2}) \circ W_{N-1} \Delta_{i,j,N-2}\,, \\ &\dots\,, \\ \Delta_{i,j,i+1} &= \sigma_{i+1}'(W_{i+1} S_i) \circ W_{i+1} \Omega_{i,j,i}\,, \\ \Omega_{i,j,i} &= \sigma_i'(W_i S_{i-1}) \circ H_i \Delta_{i,j,i-1}\,, \\ \Delta_{i,j,i-1} &= \sigma_{i-1}'(W_{i-1} S_{i-2}) \circ W_{i-1} \Delta_{i,j,i-2}\,, \\ \Delta_{i,j,i-2} &= \sigma_{i-2}'(W_{i-2} S_{i-3}) \circ W_{i-2} \Delta_{i,j,i-3}\,, \\ &\dots\,, \\ \Delta_{i,j,j} &= \sigma_j'(W_j S_{j-1}) \circ H_j S_{j-1}\,. \end{aligned} \tag{A.4.2}$$

Using the second order Taylor expansion and first order Taylor expansion, the second order term containing just H_i is the following:

$$\begin{aligned} \Delta_{i,i,N} &= \sigma_N'(W_N S_{N-1}) \circ W_N \Delta_{i,i,N-1}\,, \\ \Delta_{i,i,N-1} &= \sigma_{N-1}'(W_{N-1} S_{N-2}) \circ W_{N-1} \Delta_{i,i,N-2}\,, \\ &\dots\,, \\ \Delta_{i,i,i+1} &= \sigma_{i+1}'(W_{i+1} S_i) \circ W_{i+1} \Delta_{i,i,i}\,, \\ \Delta_{i,i,i} &= \frac{1}{2}\sigma_i''(W_i S_{i-1}) \circ H_i S_{i-1} \circ H_i S_{i-1}\,. \end{aligned} \tag{A.4.3}$$

□

A.5 Proof of Lemma 4.7.2.3

Proof. Considering $f_1(W_1 + H_1, \ldots, W_N + H_N)$ we obtain

$$\frac{1}{2}\left\| Y - S_N - \sum_{i=1}^{N} \Delta_{i,N} - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \Delta_{i,j,N} \right\|^2 = \frac{1}{2}\left\| Y - S_N - \sum_{i=1}^{N} \Delta_{i,N} - \frac{1}{2}\sum_{i=1}^{N} \Delta_{i,i,N} - \sum_{i=1}^{N}\sum_{j=1,j>i}^{N} \Delta_{i,j,N} \right\|^2 .$$

Considering upto second order terms we obtain

$$\frac{1}{2}\left\| Y - S_N \right\|^2 - \sum_{i=1}^{N} \langle \Delta_{i,N}, Y - S_N \rangle + \frac{1}{2}\left\| \sum_{i=1}^{N} \Delta_{i,N} \right\|^2 - \frac{1}{2}\sum_{i=1}^{N} \langle \Delta_{i,i,N}, Y - S_N \rangle - \sum_{i=1}^{N}\sum_{j=1,j>i}^{N} \langle \Delta_{i,j,N}, Y - S_N \rangle .$$

Note that the first order term is the following:

$$-\sum_{i=1}^{N} \langle \Delta_{i,N}, Y - S_N \rangle ,$$

and the second order term is the following:

$$\frac{1}{2}\left\| \sum_{i=1}^{N} \Delta_{i,N} \right\|^2 - \frac{1}{2}\sum_{i=1}^{N} \langle \Delta_{i,i,N}, Y - S_N \rangle - \sum_{i=1}^{N}\sum_{j=1,j>i}^{N} \langle \Delta_{i,j,N}, Y - S_N \rangle .$$

The goal is to obtain an upper bound on the second order term. Note that the following holds true:

$$\frac{1}{2}\left\| \sum_{i=1}^{N} \Delta_{i,N} \right\|^2 \leq \frac{N}{2}\sum_{i=1}^{N} \left\| \Delta_{i,N} \right\|^2 .$$

For certain $i \in \{1, \ldots, N\}$, consider the following calculation using the Assumption B:

$$\begin{aligned}
\left\| S_i \right\|^2 &= \left\| \sigma_i(W_i \ldots \sigma_1(W_1 X)) \right\|^2 \\
&\leq 2C_i^2 \left\| W_i \sigma_{i-1}(\ldots) \right\|^2 + 2D_i^2 d_i d_0 \\
&\leq 2C_i^2 \left\| W_i \right\|^2 \left\| \sigma_{i-1}(\ldots \sigma_1(W_1 X)) \right\|^2 + 2D_i^2 d_i d_0 \\
&\leq 2C_i^2 \left\| W_i \right\|^2 \left\| S_{i-1} \right\|^2 + 2D_i^2 d_i d_0
\end{aligned}$$

On recursive application of the above result, for certain $i \in \{1, \ldots, N\}$, we have:

$$\left\| S_i \right\|^2 \leq \left(\prod_{j=1}^{i} 2C_j^2 \left\| W_j \right\|^2 \right) \left\| X \right\|^2 + 2D_i^2(d_i d_0) + \sum_{j=1}^{i-1} 2D_j^2(d_j d_0) \left(\prod_{p=j+1}^{i} 2C_p^2 \left\| W_p \right\|^2 \right) = \sum_{j=0}^{i} \delta_{i,j} \left(\prod_{p=j+1}^{i} \left\| W_p \right\|^2 \right) ,$$

where we denoted coefficients of the second term with $\delta_{i,j}$ in the third term. Using Generalized AM-GM inequality we obtain the following:

$$\sum_{j=0}^{i} \delta_{i,j} \left(\prod_{p=j+1}^{i} \left\| W_p \right\|^2 \right) \leq \sum_{j=0}^{i} \delta_{i,j} \left(\frac{\sum_{p=j+1}^{i} \left\| W_p \right\|^2}{i-j} \right)^{i-j} = \sum_{j=0}^{i} \omega_{i,j} \left(\sum_{p=1}^{i} \left\| W_p \right\|^2 \right)^{i-j} ,$$

where $\omega_{i,j} := \frac{\delta_{i,j}}{(i-j)^{i-j}}$ for $i > j$. Thus, the following holds true:

$$\|S_i\|^2 \leq \sum_{j=0}^{i} \delta_{i,j} \left(\prod_{p=j+1}^{i} \|W_p\|^2 \right) \leq \sum_{j=0}^{i} \omega_{i,j} \left(\sum_{p=1}^{i} \|W_p\|^2 \right)^{i-j} . \tag{A.5.1}$$

For certain $i \in \{1, \dots, N\}$, considering the second order term $\|\Delta_{i,N}\|^2$, using the previously calculation of $\|S_i\|^2$ and Generalized AM-GM inequality we obtain the following:

$$\begin{aligned}
\|\Delta_{i,N}\|^2 &\leq \left(\prod_{j=(i+1)}^{N} E_j^2 \|W_j\|^2 \right) E_i^2 \|H_i\|^2 \|S_{i-1}\|^2 , \\
&\leq (E^2)^{N-i+1} \left(\prod_{j=(i+1)}^{N} \|W_j\|^2 \right) \|H_i\|^2 \|S_{i-1}\|^2 , \\
&\leq (E^2)^{N-i+1} \sum_{j=0}^{i-1} \delta_{i-1,j} \left(\prod_{p=(j+1),p\neq i}^{N} \|W_j\|^2 \right) \|H_i\|^2 , \\
&\leq (E^2)^{N-i+1} \sum_{j=0}^{i-1} \tilde{\theta}_{i,j} \left(\sum_{p=j+1,p\neq i}^{N} \|W_p\|^2 \right)^{N-j-1} \|H_i\|^2 , \\
&\leq (E^2)^{N-i+1} \sum_{j=0}^{i-1} \tilde{\theta}_{i,j} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-j-1} \|H_i\|^2 ,
\end{aligned}$$

where $\tilde{\theta}_{i,j} = \frac{\delta_{i-1,j}}{(N-j-1)^{N-j-1}}$.
For certain $i \in \{1, \dots, N\}$, considering the second order term $\|\Delta_{i,i,N}\|$, using the previously calculation of $\|S_i\|^2$ and Generalized AM-GM inequality we obtain the following:

$$\begin{aligned}
&\|\Delta_{i,i,N}\| \\
&\leq \frac{1}{2} \left(\prod_{j=(i+1)}^{N} E_j \|W_j\| \right) F_i \|H_i\|^2 \|S_{i-1}\|^2 , \\
&\leq \frac{1}{4} F(E)^{N-i} \left(\left(\prod_{j=(i+1)}^{N} \|W_j\|^2 \right) \|H_i\|^2 \|S_{i-1}\|^2 \right) + \frac{1}{4} F(E)^{N-i} \|H_i\|^2 \|S_{i-1}\|^2 , \\
&\leq \frac{1}{4} F(E)^{N-i} \left(\left(\prod_{j=(i+1)}^{N} \|W_j\|^2 \right) \|H_i\|^2 \|S_{i-1}\|^2 \right) + \frac{1}{4} F(E)^{N-i} \sum_{j=0}^{i-1} \omega_{i-1,j} \left(\sum_{p=1}^{i} \|W_p\|^2 \right)^{i-j-1} \|H_i\|^2 , \\
&\leq \frac{1}{4} F(E)^{N-i} \sum_{j=0}^{i-1} \tilde{\theta}_{i,j} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-j-1} \|H_i\|^2 + \frac{1}{4} F(E)^{N-i} \sum_{j=0}^{i-1} \omega_{i-1,j} \left(\sum_{p=1}^{i} \|W_p\|^2 \right)^{i-j-1} \|H_i\|^2 ,
\end{aligned}$$

Similarly, we obtain the following upper bound on $\|\Delta_{i,i,N}\| \, \|S_N\|$:

$$\begin{aligned}
\|\Delta_{i,i,N}\| \, \|S_N\| &\leq \frac{1}{2}\left(\prod_{j=(i+1)}^{N} E_j \|W_j\|\right) F_i \|H_i\|^2 \|S_{i-1}\|^2 \|S_N\| \ , \\
&\leq \frac{1}{4} E^{N-i} F \left(\left(\prod_{j=(i+1)}^{N} \|W_j\|^2\right) \|H_i\|^2 \|S_{i-1}\|^2 + \|H_i\|^2 \|S_{i-1}\|^2 \|S_N\|^2\right) \ , \\
&\leq \frac{1}{4} E^{N-i} F \sum_{j=0}^{i-1} \tilde{\theta}_{i,j} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-j-1} \|H_i\|^2 \\
&+ \frac{1}{4} E^{N-i} F \left(\sum_{\widehat{j}=0}^{i-1} \sum_{j=0}^{N} \omega_{i-1,\widehat{j}} \, \omega_{N,j} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-j+i-\widehat{j}-1}\right) \|H_i\|^2 \ .
\end{aligned}$$

where we use the following bound to bound the second term:

$$\begin{aligned}
&\|H_i\|^2 \|S_{i-1}\|^2 \|S_N\|^2 \\
&\leq \left(\sum_{\widehat{j}=0}^{i-1} \omega_{i-\widehat{j}} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{i-\widehat{j}-1}\right) \left(\sum_{j=0}^{N} \omega_{N,j} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-j}\right) \|H_i\|^2 \ , \\
&= \left(\sum_{\widehat{j}=0}^{i-1} \sum_{j=0}^{N} \omega_{i-\widehat{j}} \, \omega_{N,j} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-j+i-\widehat{j}-1}\right) \|H_i\|^2 \ .
\end{aligned}$$

Using similar techniques as above, we obtain the following upper bound on $\|\Delta_{i,j,N}\|$:

$$\begin{aligned}
&\|\Delta_{i,j,N}\| \\
&= \left(\prod_{p=i}^{N} E_p\right) \left(\prod_{p=i+1, p \neq j}^{N} \|W_p\|\right) \|H_j\| \, \|H_i\| \, \|S_{i-1}\| \ , \\
&\leq \left(\prod_{p=i}^{N} E_p\right) \left(\frac{\|H_j\|^2 + \|H_i\|^2}{2}\right) \left(\frac{\left(\prod_{p=i+1, p \neq j}^{N} \|W_p\|^2\right) + \|S_{i-1}\|^2}{2}\right) \ , \\
&\leq E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \left(\prod_{p=i+1, p \neq j}^{N} \|W_p\|^2\right) + E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \|S_{i-1}\|^2 \ , \\
&\leq \frac{E^{N-i+1}}{(N-i-1)^{N-i-1}} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \left(\sum_{p=i+1, p \neq j}^{N} \|W_p\|^2\right)^{N-i-1} + E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \|S_{i-1}\|^2 \ , \\
&\leq E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \left(\frac{\left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-i-1}}{(N-i-1)^{N-i-1}} + \sum_{q=0}^{i-1} \omega_{i-1,q} \left(\sum_{p=1}^{i-1} \|W_p\|^2\right)^{i-1-q}\right) \ ,
\end{aligned}$$

where we used

$$\left(\sum_{p=i+1,\, p\neq j}^{N} \|W_p\|^2\right)^{N-i-1} \leq \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-i-1} \quad \text{and} \quad \|S_{i-1}\|^2 \leq \sum_{q=0}^{i-1} \omega_{i-1,q} \left(\sum_{p=1}^{i-1} \|W_p\|^2\right)^{i-1-q} .$$

Based on the previously calculated terms and using similar techniques we obtain the following:

$$\begin{aligned}
\|\Delta_{i,j,N}\| \, \|S_N\| &\leq \left(\prod_{p=(j+1)}^{N} E_p \|W_p\|\right) E_j \|H_j\| \left(\prod_{q=(i+1)}^{j-1} E_q \|W_q\|\right) E_i \|H_i\| \, \|S_{i-1}\| \, \|S_N\| \, , \\
&\leq E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{2}\right) \frac{\left(\prod_{p=(j+1)}^{N} \|W_p\|^2\right)\left(\prod_{q=(i+1)}^{j-1} \|W_q\|^2\right) \|S_{i-1}\|^2 + \|S_N\|^2}{2} \, , \\
&\leq E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4}\right) \left(\sum_{u=0}^{i-1} \tilde{\delta}_{i-1,u} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-u-2} + \sum_{p=0}^{N} \omega_{N,p} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-p}\right) ,
\end{aligned}$$

where with $\tilde{\delta}_{i-1,u} = \frac{\delta_{i-1,u}}{(N-u-2)^{N-u-2}}$, we used

$$\left(\prod_{p=(i+1),\, p\neq j}^{N} \|W_p\|^2\right) \|S_{i-1}\|^2 \leq \sum_{u=0}^{i-1} \delta_{i-1,u} \left(\prod_{p=(u+1), p\neq j,i}^{N} \|W_p\|^2\right) \leq \sum_{u=0}^{i-1} \tilde{\delta}_{i-1,u} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-u-2} ,$$

$$\|S_N\|^2 \leq \sum_{p=0}^{N} \omega_{N,p} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-p} .$$

Using the previously calculated entities, we obtain the following upper bound on the second order term:

$$\frac{1}{2} \left\| \sum_{i=1}^{N} \Delta_{i,N} \right\|^2 \leq \frac{N}{2} \sum_{i=1}^{N} \|\Delta_{i,N}\|^2 \leq \frac{N}{2} \sum_{i=1}^{N} \sum_{j=0}^{i-1} (E^2)^{N-i+1} \tilde{\theta}_{i,j} \left(\sum_{p=1}^{N} \|W_p\|^2\right)^{N-j-1} \|H_i\|^2 .$$

Consider the other second terms containing $\Delta_{i,i,N}$, we obtain the following result:

$$\begin{aligned}
&\left\langle \frac{1}{2}\sum_{i=1}^{N}\Delta_{i,i,N}, Y - S_N \right\rangle \leq \frac{1}{2}\sum_{i=1}^{N}\|\Delta_{i,i,N}\|\,\|Y\| + \frac{1}{2}\sum_{i=1}^{N}\|\Delta_{i,i,N}\|\,\|S_N\|\ ,\\
&\leq \frac{\|Y\|\,F}{8}\sum_{i=1}^{N}\sum_{j=0}^{i-1}E^{N-i}\tilde{\theta}_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2 + \frac{\|Y\|\,F}{8}\sum_{i=1}^{N}\sum_{j=0}^{i-1}E^{N-i}\omega_{i,j}\left(\sum_{p=1}^{i}\|W_p\|^2\right)^{i-j-1}\|H_i\|^2\ ,\\
&+\sum_{i=1}^{N}\sum_{j=0}^{i-1}\frac{F}{8}(E^2)^{N-i}\tilde{\theta}_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2\\
&+\frac{F}{8}\sum_{i=1}^{N}E^{N-i}\left(\sum_{\widehat{j}=0}^{i-1}\sum_{j=0}^{N}\omega_{i-1,\widehat{j}}\,\omega_{N,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j+i-\widehat{j}-1}\right)\|H_i\|^2
\end{aligned}$$

Using the notation that $0^0 := 1$ and considering the other second terms containing $\Delta_{i,j,N}$, we obtain the following result:

$$\begin{aligned}
&\left\langle \sum_{i=1}^{N}\sum_{j=1,j>i}^{N}\Delta_{i,j,N}, Y - S_N \right\rangle \leq \sum_{i=1}^{N}\sum_{j=1,j>i}^{N}\|\Delta_{i,j,N}\|\,\|Y\| + \sum_{i=1}^{N}\sum_{j=1,j>i}^{N}\|\Delta_{i,j,N}\|\,\|S_N\|\ ,\\
&\leq \sum_{i=1}^{N}\sum_{j=1,j>i}^{N}\|Y\|\,E^{N-i+1}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right)\left(\frac{\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-i-1}}{(N-i-1)^{N-i-1}} + \sum_{q=0}^{i-1}\omega_{i-1,q}\left(\sum_{p=1}^{i-1}\|W_p\|^2\right)^{i-1-q}\right)\ ,\\
&+\sum_{i=1}^{N}\sum_{j=1,j>i}^{N}E^{N-i+1}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right)\left(\sum_{u=0}^{i-1}\tilde{\delta}_{i-1,u}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-u-2} + \sum_{p=0}^{N}\omega_{N,p}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-p}\right)\ .
\end{aligned}$$

Now, we consider all the terms we obtain in the second order terms to upper bound them even further such that they become manageable for further calculation. Simple manipulations provide the following term:

$$\begin{aligned}
&\frac{N}{2}\sum_{i=1}^{N}\sum_{j=0}^{i-1}(E^2)^{N-i+1}\tilde{\theta}_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2\\
&\leq \frac{N}{2}\left(\max_{i\in\{1,\dots,N\}}\max_{j\in\{0,\dots,i-1\}}(E^2)^{N-i+1}\tilde{\theta}_{i,j}\right)\sum_{i=1}^{N}\sum_{j=0}^{i-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2\ ,\\
&\leq \Theta_1\left(\sum_{j=0}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\right)\sum_{i=1}^{N}\|H_i\|^2\ ,
\end{aligned}$$

where

$$\Theta_1 := \frac{N}{2}\left(\max_{i\in\{1,\dots,N\}}\max_{j\in\{0,\dots,i-1\}}(E^2)^{N-i+1}\tilde{\theta}_{i,j}\right)\ . \tag{A.5.2}$$

Similarly, the following calculation considers all the other remaining terms obtained in the upper bounds of the second order terms:

$$
\begin{aligned}
&\frac{\|Y\|\,F}{8}\sum_{i=1}^{N}\sum_{j=0}^{i-1}E^{N-i}\tilde{\theta}_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2\\
&\le \frac{\|Y\|\,F}{8}\left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}\tilde{\theta}_{i,j}\right)\left(\sum_{j=0}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\right)\left(\sum_{i=1}^{N}\|H_i\|^2\right),\\
&=\Theta_2\left(\sum_{j=0}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\right)\left(\sum_{i=1}^{N}\|H_i\|^2\right),
\end{aligned}
$$

where $\Theta_2 := \frac{\|Y\|_F}{8}\left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}E^{N-i}\tilde{\theta}_{i,j}\right)$.

Simple manipulations result in the following:

$$
\begin{aligned}
&\frac{\|Y\|}{8}\sum_{i=1}^{N}\sum_{j=0}^{i-1}E^{N-i}\omega_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{i-j-1}\|H_i\|^2\\
&\le \frac{\|Y\|}{8}\left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}E^{N-i}\omega_{i,j}\right)\sum_{i=1}^{N}\sum_{j=0}^{i-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{i-j-1}\|H_i\|^2,\\
&\le \Theta_3\left(\sum_{j=0}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\right)\sum_{i=1}^{N}\|H_i\|^2 .
\end{aligned}
$$

where $\Theta_3 := \frac{\|Y\|_F}{8}\left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}\omega_{i,j}E^{N-i}\right)$.

Simple manipulations result in the following:

$$
\begin{aligned}
&\sum_{i=1}^{N}\sum_{j=0}^{i-1}\frac{F}{8}(E)^{N-i}\tilde{\theta}_{i,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2\\
&\le \left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}\frac{F}{8}(E)^{N-i}\tilde{\theta}_{i,j}\right)\sum_{i=1}^{N}\sum_{j=0}^{i-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\|H_i\|^2,\\
&\le \Theta_4\left(\sum_{j=0}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j-1}\right)\sum_{i=1}^{N}\|H_i\|^2 ,
\end{aligned}
$$

where $\Theta_4 := \left(\max_{i\in\{1,\ldots,N\}}\max_{j\in\{0,\ldots,i-1\}}\frac{F}{8}(E)^{N-i}\tilde{\theta}_{i,j}\right)$.

Simple manipulations result in the following:

$$\begin{aligned}
&\sum_{i=1}^{N}\frac{1}{8}E^{N-i}F\left(\sum_{\widehat{j}=0}^{i-1}\sum_{j=0}^{N}\omega_{i-\widehat{j}}\,\omega_{N,j}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j+i-\widehat{j}-1}\right)\|H_i\|^2\\
&\leq \Theta_5\left(\sum_{\widehat{j}=0}^{N-1}\sum_{j=0}^{N}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-j+N-\widehat{j}-1}\right)\left(\sum_{i=1}^{N}\|H_i\|^2\right)\\
&\leq \Theta_5\left(\sum_{j=0}^{2N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{j}\right)\left(\sum_{i=1}^{N}\|H_i\|^2\right)
\end{aligned}$$

where $\Theta_5 = \left(\max_{i=\{1,\ldots,N\}}\max_{\widehat{j}=\{0,\ldots,i-1\}}\max_{j=\{0,\ldots,N\}}\frac{F}{8}E^{N-i}\omega_{i-1,\widehat{j}}\omega_{N,j}\right)$.

Simple manipulations result in the following:

$$\begin{aligned}
&\sum_{i=1}^{N-1}\sum_{j=1,j>i}^{N}\frac{E^{N-i+1}\|Y\|}{(N-i-1)^{N-i-1}}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right)\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-i-1}\\
&\leq\left(\max_{i\in\{1,\ldots,N-1\}}\frac{E^{N-i+1}\|Y\|}{(N-i-1)^{N-i-1}}\right)\left(\sum_{i=1}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-i-1}\right)\sum_{i=1}^{N}\sum_{j=1,j>i}^{N}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right),\\
&\leq\Theta_6\left(\sum_{i=1}^{N-1}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{N-i-1}\right)\left(\sum_{i=1}^{N}\|H_i\|^2\right),
\end{aligned}$$

where $\Theta_6 = \frac{N-1}{4}\left(\max_{i\in\{1,\ldots,N-1\}}\frac{E^{N-i+1}\|Y\|}{(N-i-1)^{N-i-1}}\right)$.

Simple manipulations result in the following:

$$\begin{aligned}
&\sum_{i=1}^{N-1}\sum_{j=1,j>i}^{N}\|Y\|\,E^{N-i+1}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right)\sum_{q=0}^{i-1}\omega_{i-1,q}\left(\sum_{p=1}^{i-1}\|W_p\|^2\right)^{i-1-q}\\
&\leq\left(\max_{i\in\{1,\ldots,N-1\}}\max_{q\in\{0,\ldots,i-1\}}\|Y\|\,E^{N-i+1}\omega_{i-1,q}\right)\sum_{i=1}^{N-1}\sum_{j=1,j>i}^{N}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right)\sum_{q=0}^{i-1}\left(\sum_{p=1}^{i-1}\|W_p\|^2\right)^{q},\\
&\leq\left(\max_{i\in\{1,\ldots,N-1\}}\max_{q\in\{0,\ldots,i-1\}}\|Y\|\,E^{N-i+1}\omega_{i-1,q}\right)\left(\sum_{q=0}^{N-2}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{q}\right)\sum_{i=1}^{N-1}\sum_{j=1,j>i}^{N}\left(\frac{\|H_j\|^2+\|H_i\|^2}{4}\right),\\
&\leq\Theta_7\left(\sum_{q=0}^{N-2}\left(\sum_{p=1}^{N}\|W_p\|^2\right)^{q}\right)\left(\sum_{i=1}^{N-1}\|H_i\|^2\right),
\end{aligned}$$

where $\Theta_7 = \frac{N-1}{4}\left(\max_{i\in\{1,\ldots,N-1\}}\max_{q\in\{0,\ldots,i-1\}}\|Y\|\,E^{N-i+1}\omega_{i-1,q}\right)$.

Simple manipulations result in the following:

$$\begin{aligned}
&\sum_{i=1}^{N-1} \sum_{j=1,j>i}^{N} E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4} \right) \sum_{u=0}^{i-1} \tilde{\delta}_{i-1,u} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-u-2} \\
&\leq \left(\max_{i \in \{1,\dots,N-1\}} \max_{u \in \{0,\dots,i-1\}} E^{N-i+1} \tilde{\delta}_{i-1,u} \right) \sum_{i=1}^{N-1} \sum_{j=1,j>i}^{N} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4} \right) \sum_{u=0}^{i-1} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-u-2}, \\
&\leq \left(\max_{i \in \{1,\dots,N-1\}} \max_{u \in \{0,\dots,i-1\}} E^{N-i+1} \tilde{\delta}_{i-1,u} \right) \left(\sum_{u=0}^{N-2} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-u-2} \right) \sum_{i=1}^{N-1} \sum_{j=1,j>i}^{N} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4} \right), \\
&\leq \Theta_8 \left(\sum_{u=0}^{N-2} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-u-2} \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right),
\end{aligned}$$

where $\Theta_8 = \frac{N-1}{4} \left(\max_{i \in \{1,\dots,N-1\}} \max_{u \in \{0,\dots,i-1\}} E^{N-i+1} \tilde{\delta}_{i-1,u} \right)$.

Simple manipulations result in the following:

$$\begin{aligned}
&\sum_{i=1}^{N-1} \sum_{j=1,j>i}^{N} E^{N-i+1} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4} \right) \sum_{q=0}^{N} \omega_{N,q} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-q} \\
&\leq \left(\max_{i=\{1,\dots,N-1\}} \max_{q=\{0,\dots,N\}} E^{N-i+1} \omega_{N,q} \right) \sum_{i=1}^{N-1} \sum_{j=1,j>i}^{N} \left(\frac{\|H_j\|^2 + \|H_i\|^2}{4} \right) \sum_{q=0}^{N} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-q}, \\
&\leq \Theta_9 \left(\sum_{q=0}^{N} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{q} \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right),
\end{aligned}$$

where $\Theta_9 = \frac{N-1}{4} \left(\max_{i=\{1,\dots,N-1\}} \max_{q=\{0,\dots,N\}} E^{N-i+1} \omega_{N,q} \right)$.

Combining all the previously calculated terms, we obtain the following upper bound:

$$\begin{aligned}
&\langle (H_1, \dots, H_N), \nabla^2 f_1(W)(H_1, \dots, H_N) \rangle, \\
&\leq \frac{1}{2} \left\| \sum_{i=1}^{N} \Delta_{i,N} \right\|^2 + \frac{1}{2} \sum_{i=1}^{N} \|\Delta_{i,i,N}\| \|Y\| + \frac{1}{2} \sum_{i=1}^{N} \|\Delta_{i,i,N}\| \|S_N\| + \sum_{i=1}^{N} \sum_{j=1,j>i}^{N} \|\Delta_{i,j,N}\| \|Y\| + \sum_{i=1}^{N} \sum_{j=1,j>i}^{N} \|\Delta_{i,j,N}\| \|S_N\|, \\
&\leq \left(\sum_{i=1}^{9} \Theta_i \right) \sum_{u=0}^{N-2} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{u} \left(\sum_{i=1}^{N} \|H_i\|^2 \right) + (\Theta_1 + \Theta_2 + \Theta_3 + \Theta_4 + \Theta_6) \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N-1} \left(\sum_{i=1}^{N} \|H_i\|^2 \right) \\
&+ \Theta_9 \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{N} \left(\sum_{i=1}^{N} \|H_i\|^2 \right) + \Theta_5 \left(\sum_{j=0}^{2N-1} \left(\sum_{p=1}^{N} \|W_p\|^2 \right)^{j} \right) \left(\sum_{i=1}^{N} \|H_i\|^2 \right).
\end{aligned}$$

□

Appendix B

Appendix for CoCaIn BPG - Chapter 5

B.1 Proof of Lemma 5.5.0.1

Proof. From the three points identity (see (4.3.2)) we have

$$\begin{aligned} D_h\left(y, x_2\right) &= D_h\left(y, x_1\right) + D_h\left(x_1, x_2\right) + \left\langle \nabla h\left(x_1\right) - \nabla h\left(x_2\right), y - x_1 \right\rangle \\ &= D_h\left(y, x_1\right) + D_h\left(x_1, x_2\right) + \gamma \left\langle \nabla h\left(x_1\right) - \nabla h\left(x_2\right), x_1 - x_2 \right\rangle \\ &= D_h\left(y, x_1\right) + D_h\left(x_1, x_2\right) + \gamma\left(D_h\left(x_1, x_2\right) + D_h\left(x_2, x_1\right)\right). \end{aligned}$$

Now, from (5.5.1), we obtain that

$$D_h\left(y, x_2\right) \leq \frac{1}{\alpha\left(h\right)}\left[D_h\left(x_1, y\right) + \left(\gamma\alpha\left(h\right) + 1 + \gamma\right) D_h\left(x_2, x_1\right)\right].$$

On the other hand, since $x_1 = \left(y + \gamma x_2\right) / \left(1 + \gamma\right)$, we can use the fact that $u \to D_h\left(u, v\right)$, for a fixed $v \in \operatorname{int} \operatorname{dom} h$, is a convex function and therefore

$$D_h\left(x_1, y\right) \leq \frac{\gamma}{1+\gamma} D_h\left(x_2, y\right) \leq \frac{\gamma}{\alpha\left(h\right)\left(1+\gamma\right)} D_h\left(y, x_2\right),$$

where the last inequality follows from (5.5.1). By combining the last two inequalities we derive that

$$D_h\left(x_1, y\right) \leq \frac{\gamma}{\alpha\left(h\right)^2\left(1+\gamma\right)}\left[D_h\left(x_1, y\right) + \left(\gamma\alpha\left(h\right) + 1 + \gamma\right) D_h\left(x_2, x_1\right)\right],$$

and, by re-arranging we have

$$D_h\left(x_1, y\right) \leq \frac{\gamma\left(\gamma\alpha\left(h\right) + 1 + \gamma\right)}{\alpha\left(h\right)^2\left(1+\gamma\right) - \gamma} D_h\left(x_2, x_1\right).$$

First, it is easy to verify that for $\gamma < \alpha\left(h\right)^2 / \left(1 - \alpha\left(h\right)^2\right)$, the denominator is positive. In addition, to find γ such that

$$\frac{\gamma\left(\gamma\alpha\left(h\right) + 1 + \gamma\right)}{\alpha\left(h\right)^2\left(1+\gamma\right) - \gamma} \leq \kappa,$$

we will use simple algebraic manipulations. Indeed, by re-arranging we have

$$\gamma^2 \underbrace{\left(\alpha\left(h\right)+1\right)}_{a} + \gamma \underbrace{\left(1+\kappa-\alpha\left(h\right)^2\kappa\right)}_{b} - \alpha\left(h\right)^2\kappa \leq 0.$$

Since $\alpha\left(h\right)^2 \leq 1$, it follows that $b > 0$. We also have that $\Delta = b^2 + 4a\alpha\left(h\right)^2\kappa > 0$, and thus there exists a positive root denoted by γ^*. Therefore, for any $\gamma \in [0, \gamma^*]$, the desired result follows. □

B.2 Proof of Lemma 5.6.0.1

Proof. Fix $k \geq 1$. From the convexity of $f\left(\cdot\right) - (\alpha/2)\left\|\cdot\right\|^2$, which holds thanks to Assumption D(iii), we obtain from the sub-gradient inequality [150, Example 8.8 and Proposition 8.12] that

$$f_0\left(x^k\right) - \frac{\alpha}{2}\left\|x^k\right\|^2 \geq f_0\left(x^{k+1}\right) - \frac{\alpha}{2}\left\|x^{k+1}\right\|^2 + \left\langle \xi^{k+1} - \alpha x^{k+1}, x^k - x^{k+1}\right\rangle,$$

where $\xi^{k+1} \in \partial f_0\left(x^{k+1}\right)$. By rearranging the inequality we obtain

$$f_0\left(x^k\right) \geq f_0\left(x^{k+1}\right) + \frac{\alpha}{2}\left\|x^{k+1} - x^k\right\|^2 + \left\langle \xi^{k+1}, x^k - x^{k+1}\right\rangle. \tag{B.2.1}$$

From the optimality condition of step (5.4.5), we have that

$$\xi^{k+1} + \nabla f_1\left(y^k\right) + \frac{1}{\tau_k}\left(\nabla h\left(x^{k+1}\right) - \nabla h\left(y^k\right)\right) = 0,$$

which combined with (B.2.1) yields that

$$\begin{aligned}
&f_0\left(x^k\right)\\
&\geq f_0\left(x^{k+1}\right) + \frac{\alpha}{2}\left\|x^{k+1} - x^k\right\|^2 - \left\langle \nabla f_1\left(y^k\right), x^k - x^{k+1}\right\rangle + \frac{1}{\tau_k}\left\langle \nabla h\left(y^k\right) - \nabla h\left(x^{k+1}\right), x^k - x^{k+1}\right\rangle\\
&= f_0\left(x^{k+1}\right) + \frac{\alpha}{2}\left\|x^{k+1} - x^k\right\|^2 - \left\langle \nabla f_1\left(y^k\right), x^k - x^{k+1}\right\rangle + \frac{1}{\tau_k}\left(D_h\left(x^k, x^{k+1}\right) + D_h\left(x^{k+1}, y^k\right) - D_h\left(x^k, y^k\right)\right),
\end{aligned}$$

where the last equality follows from the three-points identity (see (4.3.2)). On the other hand, using the lower approximation given in (5.4.4) and the upper approximation given in (5.4.6), we have that

$$f_1\left(x^k\right) \geq f_1\left(x^{k+1}\right) + \left\langle \nabla f_1\left(y^k\right), x^k - x^{k+1}\right\rangle - \underline{L}_k D_h\left(x^k, y^k\right) - \bar{L}_k D_h\left(x^{k+1}, y^k\right).$$

Combining the last two inequalities and using the fact that $\tau_k^{-1} \geq \bar{L}_k$, implies that

$$f\left(x^k\right) \geq f\left(x^{k+1}\right) + \frac{\alpha}{2}\left\|x^{k+1} - x^k\right\|^2 + \frac{1}{\tau_k} D_h\left(x^k, x^{k+1}\right) - \left(\frac{1}{\tau_k} + \underline{L}_k\right) D_h\left(x^k, y^k\right),$$

which completes the proof. □

B.3 Proof of Proposition 5.6.1.1

Proof. Multiplying (5.6.1) with τ_k, we obtain

$$\tau_k\left(f\left(x^k\right)-v(\mathcal{P})\right) \geq \tau_k\left(f\left(x^{k+1}\right)-v(\mathcal{P})\right)+\frac{\alpha\tau_k}{2}\left\|x^{k+1}-x^k\right\|^2+D_h\left(x^k,x^{k+1}\right)-\left(1+\underline{L}_k\tau_k\right)D_h\left(x^k,y^k\right).$$

By the definition of the Lyapunov function f_δ^k and the fact that $\tau_k \leq \tau_{k-1}$ we have

$$\begin{aligned} f_\delta^k\left(x^k,x^{k-1}\right) \geq f_\delta^{k+1}\left(x^{k+1},x^k\right)+\frac{\alpha\tau_k}{2}\left\|x^{k+1}-x^k\right\|^2+(1-\delta)\,D_h\left(x^k,x^{k+1}\right)\\ +\delta D_h\left(x^{k-1},x^k\right)-\left(1+\underline{L}_k\tau_k\right)D_h\left(x^k,y^k\right). \end{aligned}$$

With $1-\delta>0$ and the strong convexity of $h(\,\cdot\,)$, that follows from Assumption D(i), we obtain

$$\frac{\alpha\tau_k}{2}\left\|x^{k+1}-x^k\right\|^2+(1-\delta)\,D_h\left(x^k,x^{k+1}\right) \geq \left(\frac{\alpha\tau_k}{2}+(1-\delta)\frac{\sigma}{2}\right)\left\|x^{k+1}-x^k\right\|^2 \geq 0,$$

where the last inequality holds, since $\tau_k^{-1} \geq \bar{L}_k$ and $\bar{L}_k \geq -\alpha/\left(1-\delta\right)\sigma$. Next, we observe that

$$D_h\left(x^k,y^k\right) \leq \frac{\delta-\epsilon}{\left(1+\underline{L}_k\tau_{k-1}\right)}D_h\left(x^{k-1},x^k\right) \leq \frac{\delta-\epsilon}{\left(1+\underline{L}_k\tau_k\right)}D_h\left(x^{k-1},x^k\right),$$

where the first inequality is due to the step (5.4.3) of the algorithm and the second inequality is due to fact that $\tau_k \leq \tau_{k-1}$. By rearranging we obtain,

$$\delta D_h\left(x^{k-1},x^k\right)-\left(1+\underline{L}_k\tau_k\right)D_h\left(x^k,y^k\right) \geq \epsilon D_h\left(x^{k-1},x^k\right)$$

thus completing the proof. □

B.4 Proof of Proposition 10.4.1.2

Proof. (i) This follows trivially from Proposition 5.6.1.1, since $\epsilon > 0$.

(ii) Let n be a positive integer. Summing (5.6.3) from $k=1$ to n we get

$$\sum_{k=1}^n D_h\left(x^{k-1},x^k\right) \leq \frac{1}{\epsilon}\left(f_\delta^1\left(x^1,x^0\right)-f_\delta^{n+1}\left(x^{n+1},x^n\right)\right) \leq \frac{1}{\epsilon}f_\delta^1\left(x^1,x^0\right), \tag{B.4.1}$$

since $f_\delta^{n+1}\left(x^{n+1},x^n\right) \geq 0$. Taking the limit as $n\to\infty$, we obtain the first desired assertion, from which we immediately deduce that $\left\{D_h\left(x^{k-1},x^k\right)\right\}_{k\in\mathbb{N}}$ converges to zero.

(iii) From (B.4.1) we also obtain,

$$n\min_{1\leq k\leq n} D_h\left(x^{k-1},x^k\right) \leq \sum_{k=1}^n D_h\left(x^{k-1},x^k\right) \leq \frac{1}{\epsilon}f_\delta^1\left(x^1,x^0\right),$$

which after division by n yields the desired result.

□

B.5 Proof of Theorem 5.6.2.1

The set of all limit points of $\{x^k\}_{k\in\mathbb{N}}$ is defined by

$$\omega\left(x^0\right) := \left\{\overline{x} \in \mathbb{R}^N : \exists \text{ an increasing sequence of integers } \{k_l\}_{l\in\mathbb{N}} \text{ s.t. } x^{k_l} \to \overline{x} \text{ as } l \to \infty\right\}.$$

We first prove the following result.

Lemma B.5.0.1. *Let $\{x^k\}_{k\in\mathbb{N}}$ be a bounded gradient-like descent sequence for minimizing f_{δ_1}. Then, $\omega\left(x^0\right)$ is a nonempty and compact subset of* crit f*, and we have*

$$\lim_{k\to\infty} \operatorname{dist}\left(x^k, \omega\left(x^0\right)\right) = 0. \tag{B.5.1}$$

In addition, the objective function f is finite and constant on $\omega\left(x^0\right)$.

Proof. Since $\{x^k\}_{k\in\mathbb{N}}$ is bounded there is $x^* \in \mathbb{R}^N$ and a subsequence $\left\{x^{k_q}\right\}_{q\in\mathbb{N}}$ such that $x^{k_q} \to x^*$ as $q \to \infty$ and hence $\omega\left(x^0\right)$ is nonempty. Moreover, the set $\omega\left(x^0\right)$ is compact since it can be viewed as an intersection of compact sets. Now, from conditions (C1) and (C3), and the lower semicontinuity of f (which follows from the lower semi-continuity of f_0 and f_1, see Assumption A), we obtain

$$\lim_{k\to\infty} D_h\left(x^{k-1}, x^k\right) \leq \lim_{k\to\infty} \left\|x^k - x^{k-1}\right\|^2 = 0$$

and therefore

$$\lim_{q\to\infty} f_{\delta_1}\left(x^{k_q+1}, x^{k_q}\right) = \lim_{q\to\infty} f\left(x^{k_q}\right) = f\left(x^*\right). \tag{B.5.2}$$

On the other hand, from conditions (C1) and (C2), there is $w^{k+1} \in \partial f_{\delta_1}\left(x^{k+1}, x^k\right)$, $k \in \mathbb{N}$, such that $w^{k+1} \to 0$ as $k \to \infty$. The closedness property of ∂f_{δ_1} implies thus that $0 \in \partial f_{\delta_1}\left(x^*, x^*\right) = \left(\partial f\left(x^*\right), 0\right)$. This proves that x^* is a critical point of f, and hence (B.5.1) is valid.
To complete the proof, let $\lim_{k\to\infty} f_{\delta_1}\left(x^{k+1}, x^k\right) = l \in \mathbb{R}$. Then $\left\{f_{\delta_1}\left(x^{k_q+1}, x^{k_q}\right)\right\}_{q\in\mathbb{N}}$ converges to l and from (B.5.2) we have $f\left(x^*\right) = l$. Hence the restriction of f_{δ_1} to $\omega\left(x^0\right)$ equals l. □

We can now restate and prove Theorem 5.6.2.1.

Theorem B.5.0.2. *Let $\{x^k\}_{k\in\mathbb{N}}$ be a bounded gradient-like descent sequence for minimizing f_{δ_1}. If f and h satisfy the KL property, then the sequence $\{x^k\}_{k\in\mathbb{N}}$ has finite length, i.e., $\sum_{k=1}^{\infty} \left\|x^{k+1} - x^k\right\| < \infty$ and it converges to $x^* \in$* crit f*.*

Proof. Since $\{x^k\}_{k\in\mathbb{N}}$ is bounded there exists a subsequence $\left\{x^{k_q}\right\}_{q\in\mathbb{N}}$ such that $x^{k_q} \to \overline{x}$ as $q \to \infty$. In a similar way as in Lemma B.5.0.1 we get that

$$\lim_{k\to\infty} f_{\delta_1}\left(x^{k+1}, x^k\right) = \lim_{k\to\infty} f\left(x^k\right) = f\left(\overline{x}\right). \tag{B.5.3}$$

If there exists an integer $\bar{k}$ for which $f_{\delta_1}\left(x^{\bar{k}+1}, x^{\bar{k}}\right) = f\left(\overline{x}\right)$ then condition (C1) would imply that $x^{\bar{k}+1} = x^{\bar{k}}$. A trivial induction show then that the sequence $\{x^k\}_{k\in\mathbb{N}}$ is stationary and the announced results are obvious. Since $\left\{f_{\delta_1}\left(x^{k+1}, x^k\right)\right\}_{k\in\mathbb{N}}$ is a nonincreasing sequence, it is clear from (B.5.3) that $f\left(\overline{x}\right) < f_{\delta_1}\left(x^{k+1}, x^k\right)$ for all $k > 0$. Again from (B.5.3) for any $\eta > 0$ there exists a nonnegative integer k_0 such that $f_{\delta_1}\left(x^{k+1}, x^k\right) < f\left(\overline{x}\right) + \eta$ for all $k > k_0$. From Lemma B.5.0.1 we know that $\lim_{k\to\infty} \operatorname{dist}\left(x^k, \omega\left(x^0\right)\right) = 0$. This means that for any $\tilde{\epsilon} > 0$ there exists a positive integer k_1 such that $\operatorname{dist}\left(x^k, \omega\left(x^0\right)\right) < \tilde{\epsilon}$ for all $k > k_1$.

From Lemma B.5.0.1 applied to f_{δ_1}, we know that $\omega\left(x^0\right)$ is nonempty and compact and that the function f is finite and constant on $\omega\left(x^0\right)$. Hence, we can apply the Uniformization Lemma 3.7.0.1 applied to f_{δ_1}, which satisfies the KL property since f and h do, with $\Omega = \omega\left(x^0\right)$. Therefore, for any $k \geq l := \max\left\{k_0, k_1\right\} + 1$, we have

$$\phi'\left(f_{\delta_1}\left(x^k, x^{k-1}\right) - f(\overline{x})\right) \operatorname{dist}\left(0, \partial f_{\delta_1}\left(x^k, x^{k-1}\right)\right) \geq 1. \tag{B.5.4}$$

This makes sense since we know that $f_{\delta_1}\left(x^k, x^{k-1}\right) > f\left(\overline{x}\right)$ for any $k > l$. Combining (B.5.4) with condition (C2), see Proposition 5.6.2.2, we get that

$$\phi'\left(f_{\delta_1}\left(x^k, x^{k-1}\right) - f\left(\overline{x}\right)\right) \geq \rho_2^{-1}\left(\left\|x^{k-1} - x^{k-2}\right\| + \left\|x^k - x^{k-1}\right\|\right)^{-1}. \tag{B.5.5}$$

For convenience, we define for all $p, q \in \mathbb{N}$ and $\overline{x}$ the following quantity

$$\Delta_{p,q} := \phi\left(f_{\delta_1}\left(x^p, x^{p-1}\right) - f\left(\overline{x}\right)\right) - \phi\left(f_{\delta_1}\left(x^q, x^{q-1}\right) - f\left(\overline{x}\right)\right).$$

From the concavity of ϕ we get that

$$\Delta_{k,k+1} \geq \phi'\left(f_{\delta_1}\left(x^k, x^{k-1}\right) - f\left(\overline{x}\right)\right)\left(f_{\delta_1}\left(x^k, x^{k-1}\right) - f_{\delta_1}\left(x^{k+1}, x^k\right)\right). \tag{B.5.6}$$

Combining condition (C1) with (B.5.5) and (B.5.6) yields, for any $k > l$, that

$$\Delta_{k,k+1} \geq \frac{\left\|x^k - x^{k-1}\right\|^2}{\rho\left(\left\|x^{k-1} - x^{k-2}\right\| + \left\|x^k - x^{k-1}\right\|\right)}, \qquad \text{where } \rho := \rho_2/\rho_1.$$

Using the fact that $2\sqrt{\alpha\beta} \leq \alpha + \beta$ for all $\alpha, \beta \geq 0$, we infer from the later inequality that

$$4\left\|x^k - x^{k-1}\right\| \leq \left\|x^{k-1} - x^{k-2}\right\| + \left\|x^k - x^{k-1}\right\| + 4\rho\Delta_{k,k+1},$$

and thus

$$3\left\|x^k - x^{k-1}\right\| \leq \left\|x^{k-1} - x^{k-2}\right\| + 4\rho\Delta_{k,k+1}. \tag{B.5.7}$$

Summing up (B.5.7) for $i = l+2, \ldots, k$ yields

$$\begin{aligned}
3\sum_{i=l+2}^{k}\left\|x^i - x^{i-1}\right\| &\leq \sum_{i=l+2}^{k}\left\|x^{i-1} - x^{i-2}\right\| + 4\rho\sum_{i=l+2}^{k}\Delta_{i,i+1} \\
&\leq \sum_{i=l+2}^{k}\left\|x^i - x^{i-1}\right\| + \left\|x^{l+1} - x^l\right\| + 4\rho\sum_{i=l+2}^{k}\Delta_{i,i+1} \\
&= \sum_{i=l+2}^{k}\left\|x^i - x^{i-1}\right\| + \left\|x^{l+1} - x^l\right\| + 4\rho\Delta_{l+2,k+1},
\end{aligned}$$

where the last equality follows from the fact that $\Delta_{p,q} + \Delta_{q,r} = \Delta_{p,r}$ for all $p, q, r \in \mathbb{N}$. Since $\phi \geq 0$, recalling the definition of $\Delta_{l+2,k+1}$, we thus have for any $k > l$ that

$$2\sum_{i=l+2}^{k}\left\|x^i - x^{i-1}\right\| \leq \left\|x^{l+1} - x^l\right\| + 4\rho\phi\left(f_{\delta_1}\left(x^{l+2}, x^{l+1}\right) - f\left(\overline{x}\right)\right),$$

which implies that $\sum_{k=1}^{\infty}\left\|x^{k+1}-x^k\right\|<\infty$, i.e., $\{x^k\}_{k\in\mathbb{N}}$ is a Cauchy sequence and hence together with Lemma B.5.0.1, we obtain the global convergence to a critical point. □

B.6 Proof of Proposition 5.6.2.2

Proof. Fix $k \geq K$. By the definition of the Lyapunov function $f_{\delta_1}(\cdot,\cdot)$ we obtain that

$$\partial f_{\delta_1}\left(x^{k+1},x^k\right)=\left(\partial f\left(x^{k+1}\right)+\delta_1\nabla^2 h\left(x^{k+1}\right)\left(x^{k+1}-x^k\right),\delta_1\left(\nabla h\left(x^k\right)-\nabla h\left(x^{k+1}\right)\right)\right).$$

Writing the optimality condition of the optimization problem which defines x^{k+1} (see (5.4.5) and recall that for $k \geq K$, we have that $\tau_k=\tau$) yields that

$$0\in\partial f_0\left(x^{k+1}\right)+\nabla f_1\left(y^k\right)+\frac{1}{\tau}\left(\nabla h\left(x^{k+1}\right)-\nabla h\left(y^k\right)\right).$$

Therefore

$$\nabla f_1\left(x^{k+1}\right)-\nabla f_1\left(y^k\right)+\frac{1}{\tau}\left(\nabla h\left(y^k\right)-\nabla h\left(x^{k+1}\right)\right)\in\partial f\left(x^{k+1}\right),$$

and by defining

$$w_1^{k+1}\equiv\nabla f_1\left(x^{k+1}\right)-\nabla f_1\left(y^k\right)+\frac{1}{\tau}\left(\nabla h\left(y^k\right)-\nabla h\left(x^{k+1}\right)\right)+\delta_1\nabla^2 h\left(x^{k+1}\right)\left(x^{k+1}-x^k\right),$$

and $w_2^{k+1}\equiv\delta_1\left(\nabla h\left(x^k\right)-\nabla h\left(x^{k+1}\right)\right)$ we obviously obtain that $w^{k+1}\in\partial f_{\delta_1}\left(x^{k+1},x^k\right)$ where $w^{k+1}=\left(w_1^{k+1},w_2^{k+1}\right)$. Since $\{x^k\}_{k\in\mathbb{N}}$ is a bounded sequence and both ∇h and ∇g are Lipschitz continuous on bounded subsets of $\mathbb{R}^N$ (see Assumption E(ii)), there exists $M>0$ such that

$$\begin{aligned}\left\|w_1^{k+1}\right\| &\leq \left\|\nabla f_1\left(x^{k+1}\right)-\nabla f_1\left(y^k\right)\right\|+\frac{1}{\tau}\left\|\nabla h\left(y^k\right)-\nabla h\left(x^{k+1}\right)\right\|+\delta_1\left\|\nabla^2 h\left(x^{k+1}\right)\right\|\left\|x^{k+1}-x^k\right\|\\ &\leq M\left(1+\frac{1}{\tau}\right)\left\|x^{k+1}-y^k\right\|+\delta_1 M\left\|x^{k+1}-x^k\right\|,\end{aligned}$$

where the last inequality follows also from the fact that $\left\|\nabla^2 h\left(x^{k+1}\right)\right\|\leq M$, since ∇h is Lipschitz continuous on bounded subsets of $\mathbb{R}^N$. Using step (5.4.2) we obtain that

$$\begin{aligned}\left\|w_1^{k+1}\right\| &\leq M\left(1+\frac{1}{\tau}\right)\left(\left\|x^{k+1}-x^k\right\|+\gamma_k\left\|x^k-x^{k-1}\right\|\right)+\delta_1 M\left\|x^{k+1}-x^k\right\|\\ &\leq M\left(1+\delta_1+\frac{1}{\tau}\right)\left\|x^{k+1}-x^k\right\|+M\left(1+\frac{1}{\tau}\right)\left\|x^k-x^{k-1}\right\|,\end{aligned}$$

where we have used the fact that $\gamma_k\leq 1$, $k\in\mathbb{N}$. Since, we also have that

$$\left\|w_2^{k+1}\right\|=\delta_1\left\|\nabla h\left(x^k\right)-\nabla h\left(x^{k+1}\right)\right\|\leq\delta_1 M\left\|x^{k+1}-x^k\right\|,$$

the desired result is proved and condition (C2) also holds true. □

B.7 Proof of Proposition 5.6.2.3

Proof. Consider a subsequence $\{x^{n_k}\}_{k\in\mathbb{N}}$ which converges to x^* (there exists such a subsequence since the sequence $\{x^k\}_{k\in\mathbb{N}}$ is assumed to be bounded). Using Proposition 5.6.2.1(ii) and the strong convexity of $h\left(\cdot\right)$, we obtain that $\lim_{k\to\infty}\left\|x^k - x^{k-1}\right\| = 0$. Therefore, the sequence $\left\{x^{n_k-1}\right\}_{k\in\mathbb{N}}$ also converges to x^*. From the definition of y^k, see (5.4.2), it also follows that $\left\{y^{n_k-1}\right\}_{k\in\mathbb{N}}$ also converges to x^*. In addition, since h is continuously differentiable on $\mathbb{R}^N$ we have that $\lim_{k\to\infty} D_h\left(x^*, y^{n_k-1}\right) = 0$. Now, from (5.4.5), it follows (after some simplifications), for all $k \geq K$, that

$$f\left(x^k\right) \leq f_0\left(x^*\right) + \left\langle x^* - x^k, \nabla f_1\left(y^{k-1}\right)\right\rangle + \frac{1}{\tau}D_h\left(x^*, y^{k-1}\right) - \frac{1}{\tau}D_h\left(x^k, y^{k-1}\right).$$

Substituting k by n_k and letting $k \to \infty$, we obtain from the fact that f_1 is continuously differentiable on $\mathbb{R}^N$, that

$$\limsup_{k\to\infty} f_0\left(x^{n_k}\right) \leq f_0\left(x^*\right).$$

Using this, and recalling that here f_1 is continuous, we obtain that $\limsup_{k\in\mathcal{K}\subset\mathbb{N}} f\left(x^{n_k}\right) \leq f\left(x^*\right)$, where $\mathcal{K} = \{n_k : k \geq K\}$. □

B.8 Proof of Lemma B.8.0.1

Lemma B.8.0.1 (Closed form inertia)**.** *For h defined in (5.7.6), we obtain the following gradient*

$$\nabla h(x) = (\|x\|_2^2 + 1)x\,, \tag{B.8.1}$$

and for any $a \in \mathbb{R}^N$, we have

$$\frac{1}{2}\left\langle a, \nabla^2 h(x) a\right\rangle \leq \frac{3}{2}\|x\|_2^2\|a\|_2^2 + \frac{1}{2}\|a\|_2^2\,. \tag{B.8.2}$$

Proof. Consider the expansion at $x + a$ till second order terms, we thus have

$$\begin{aligned} h(x+a) &= \frac{1}{4}\|x+a\|_2^4 + \frac{1}{2}\|x+a\|_2^2\,, \\ &= \frac{1}{4}\left(\|x\|_2^2 + \|a\|_2^2 + 2\left\langle a, x\right\rangle\right)^2 + \frac{1}{2}\|x+a\|_2^2\,, \\ &= \frac{1}{4}\left(\|x\|_2^4 + 4(\langle a, x\rangle)^2 + 4\|x\|_2^2\langle a, x\rangle + 2\|x\|_2^2\|a\|_2^2\right) + \frac{1}{2}\left(\|x\|_2^2 + \|a\|_2^2 + 2\langle a, x\rangle\right). \end{aligned}$$

The first order terms result in (B.8.1) and we also have

$$\frac{1}{2}\left\langle a, \nabla^2 h(x) a\right\rangle = \langle a, x\rangle^2 + \frac{1}{2}\|x\|_2^2\|a\|_2^2 + \frac{1}{2}\|a\|_2^2 \leq \frac{3}{2}\|x\|_2^2\|a\|_2^2 + \frac{1}{2}\|a\|_2^2\,,$$

where the inequality follows due to Cauchy-Schwarz inequality. □

B.9 Proof of Proposition 5.7.3.1

Proof. We use the strategy of Lemma A.3.0.3. From Lemma A.3.0.1, we have

$$
\begin{aligned}
&\int_0^1 (1-t) \int_0^1 \left\langle \nabla^2 h \left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - y^k), x^k - y^k \right\rangle dt_1 dt \\
=&\gamma_k^2 \int_0^1 (1-t) \int_0^1 \left\langle \nabla^2 h \left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - x^{k-1}), x^k - x^{k-1} \right\rangle dt_1 dt\,, \\
\leq& 2\gamma_k^2 \int_0^1 (1-t) \int_0^1 \frac{3}{2} \left\|x^k - x^{k-1}\right\|^2 \left\|x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right\|^2 dt_1 dt + 2\gamma_k^2 \int_0^1 (1-t) \frac{1}{2} \left\|x^k - x^{k-1}\right\|^2 dt_1 dt\,, \\
\leq& 2\gamma_k^2 \int_0^1 (1-t) \int_0^1 \left(3\left\|x^k - x^{k-1}\right\|^2 \left\|x^k\right\|^2 + 3\left\|x^k - x^{k-1}\right\|^2\right) dt_1 dt + 2\gamma_k^2 \int_0^1 (1-t) \frac{1}{2} \left\|x^k - x^{k-1}\right\|^2 dt_1 dt\,, \\
\leq& 3\gamma_k^2 \left(\left\|x^k - x^{k-1}\right\|^2 \left\|x^k\right\|^2 + \left\|x^k - x^{k-1}\right\|^2\right) + \frac{\gamma_k^2}{2} \left\|x^k - x^{k-1}\right\|^2\,,
\end{aligned}
$$

where in the last step we used the upper bound (B.8.2) from Lemma B.8.0.1. Also, we used the following inequality

$$
\begin{aligned}
\left\|x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right\|^2 &\leq 2\left\|x^k\right\|^2 + 2(t_1 + (1-t_1)t)^2 \gamma_k^2 \left\|x^k - x^{k-1}\right\|^2\,, \\
&\leq 2\left\|x^k\right\|^2 + 2\left\|x^k - x^{k-1}\right\|^2\,,
\end{aligned}
$$

where in the last step we used $\gamma_k^2 \leq 1$ and $(t_1 + (1-t_1)t)^2 \leq 1$. With $\int_0^1 (1-t) dt = \frac{1}{2}$ the result follows. □

Appendix C

Appendix for matrix factorization - Chapter 6

C.1 Overview of the results

Below, we provide a table with the problem or content description and corresponding section where the results are presented.

Matrix factorization problem	**Section**
Standard matrix factorization	Section C.2
L2-regularized matrix factorization	Section C.2.1
Graph regularized matrix factorization	Section C.2.2
L1-regularized matrix factorization	Section C.2.3
Nuclear norm regularized matrix factorization	Section C.2.4
Non-negative matrix factorization (NMF)	Section C.3
L2-regularized NMF	Section C.3.1
L1-regularized NMF	Section C.3.2
Graph Regularized NMF	Section C.3.3
Symmetric NMF via non-symmetric relaxation	Section C.3.4
Sparse NMF	Section C.3.5
Matrix completion	Section C.4
Closed form solution with 5th-order polynomials	Section C.5
Conversion to cubic equation	Section C.5.1
Extensions to mixed regularization terms	Section C.5.2
Technical proofs	Section A.1

C.2 Closed form solutions: Part I for matrix factorization

Since, the update steps of BPG-MF and CoCaIn BPG-MF have same structure, we provide the closed form expressions to just BPG-MF. We start with the following technical lemma.

Lemma C.2.0.1. *Let $Q \in \mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t \geq 0$ and $\|Q\|_F \neq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 = t^2\right\} \equiv \min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 \leq t^2\right\} = -t\,\|Q\|_F\ ,$$

with the minimizer at $X^* = -tQ/\left\|Q\right\|_F$.

Proof. The proof is inspired from [111, Lemma 9]. On rewriting we have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 \le t^2\right\} \equiv -\max_{X\in\mathbb{R}^{A\times B}} \left\{\langle -Q, X\rangle : \|X\|_F^2 \le t^2\right\}.$$

The expression $\langle -Q, X\rangle$ is maximized at $X^* = c(-Q)$ for certain constant c. On substituting we have

$$\langle -Q, X^*\rangle = c\left\|Q\right\|_F^2 .$$

Since, the dependence on c is linear and we additionally require $\|X\|_F^2 \le t^2$, we can set $c = \frac{t}{\|Q\|_F}$ if $\|Q\|_F \neq 0$ else $c = 0$. Hence, the minimizer to

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 \le t^2\right\}$$

is attained at $X^* = -t\frac{Q}{\|Q\|_F}$ for $\|Q\|_F \neq 0$ else $X^* = 0$. The equivalence in the statement follows as $\|X^*\|_F^2 = t^2$. □

Consider the following non-convex matrix factorization problem

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\left\|A - UZ\right\|_F^2 \right\}. \tag{C.2.1}$$

Denote $f_1 = f$, $f_0 := 0$, $h = h_a$.

Proposition C.2.0.1. *In BPG-MF, with above defined* f_1, f_0, h *the update steps in each iteration are given by* $U^{k+1} = -r\,P^k$, $Z^{k+1} = -r\,Q^k$ *where* r *is the non-negative real root of*

$$c_1\left(\left\|Q^k\right\|_F^2 + \left\|P^k\right\|_F^2\right) r^3 + c_2 r - 1 = 0\,, \tag{C.2.2}$$

with $c_1 = 3$ *and* $c_2 = \|A\|_F$. *Another equivalent formulation of the update steps in each iteration is given by* $U^{k+1} = -r\frac{\sqrt{2}P^k}{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}$ *and* $Z^{k+1} = -r\frac{\sqrt{2}Q^k}{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}$ *for some* $r \ge 0$ *such that* r *satisfies the following cubic equation*

$$2c_1 r^3 + c_2 r - \frac{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}{\sqrt{2}} = 0\,.$$

Proof. Consider the following subproblem

$$(U^{k+1}, Z^{k+1}) \in \underset{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}}{\operatorname{argmin}} \left\{ \left\langle P^k, U\right\rangle + \left\langle Q^k, Z\right\rangle + c_1\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)^2 + c_2\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right) \right\}.$$

Denote the objective in the above minimization problem as $\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})$. Now, the following holds

$$\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right) \equiv \min_{t_1\ge 0, t_2\ge 0}\left\{ \min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F = t_1, \|Z\|_F = t_2} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\}, \tag{C.2.3}$$

$$\equiv \min_{t_1\ge 0, t_2\ge 0}\left\{ \min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F \le t_1, \|Z\|_F \le t_2} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\}, \tag{C.2.4}$$

where the first step is a simple rewriting of the objective. The second step is non-trivial. In order to prove (C.2.4) we rewrite (C.2.3) as

$$\min_{t_1 \geq 0, t_2 \geq 0} \left\{ \min_{U_1 \in \mathbb{R}^{M \times K}} \left\{ \left\langle P^k, U_1 \right\rangle : \|U_1\|_F^2 = t_1^2 \right\} + \min_{Z_1 \in \mathbb{R}^{K \times N}} \left\{ \left\langle Q^k, Z_1 \right\rangle : \|Z_1\|_F^2 = t_2^2 \right\} \right.$$
$$\left. + c_1 \left(\frac{t_1^2 + t_2^2}{2} \right)^2 + c_2 \left(\frac{t_1^2 + t_2^2}{2} \right) \right\} .$$

Now, note the following equivalence due to Lemma D.2.0.1

$$\min_{U_1 \in \mathbb{R}^{M \times K}} \left\{ \left\langle P^k, U_1 \right\rangle : \|U_1\|_F^2 = t_1^2 \right\} \equiv \min_{U_1 \in \mathbb{R}^{M \times K}} \left\{ \left\langle P^k, U_1 \right\rangle : \|U_1\|_F^2 \leq t_1^2 \right\} ,$$
$$\min_{Z_1 \in \mathbb{R}^{K \times N}} \left\{ \left\langle Q^k, Z_1 \right\rangle : \|Z_1\|_F^2 = t_2^2 \right\} \equiv \min_{Z_1 \in \mathbb{R}^{K \times N}} \left\{ \left\langle Q^k, Z_1 \right\rangle : \|Z_1\|_F^2 \leq t_2^2 \right\} .$$

This proves (C.2.4). Now, we solve for (U^{k+1}, Z^{k+1}) via the following strategy. Denote

$$U_1^*(t_1) \in \operatorname{argmin} \left\{ \left\langle P^k, U_1 \right\rangle : U_1 \in \mathbb{R}^{M \times K} , \|U_1\|_F^2 \leq t_1^2 \right\} ,$$

$$Z_1^*(t_2) \in \operatorname{argmin} \left\{ \left\langle Q^k, Z_1 \right\rangle : Z_1 \in \mathbb{R}^{K \times N} , \|Z_1\|_F^2 \leq t_2^2 \right\} .$$

Then we obtain $(U^{k+1}, Z^{k+1}) = (U_1^*(t_1^*), Z_1^*(t_2^*))$, where t_1^* and t_2^* are obtained by solving the following two dimensional subproblem

$$(t_1^*, t_2^*) \in \operatorname*{argmin}_{t_1 \geq 0, t_2 \geq 0} \left\{ \min_{U_1 \in \mathbb{R}^{M \times K}} \left\{ \left\langle P^k, U_1 \right\rangle : \|U_1\|_F^2 \leq t_1^2 \right\} + \min_{Z_1 \in \mathbb{R}^{K \times N}} \left\{ \left\langle Q^k, Z_1 \right\rangle : \|Z_1\|_F^2 \leq t_2^2 \right\} \right.$$
$$\left. + c_1 \left(\frac{t_1^2 + t_2^2}{2} \right)^2 + c_2 \left(\frac{t_1^2 + t_2^2}{2} \right) \right\} .$$

Note that inner minimization subproblems can be trivially solved once we obtain $U_1^*(t_1)$ and $Z_1^*(t_2)$ via Lemma D.2.0.1. Then the solution to the subproblem in each iteration is as follows:

$$U^{k+1} = \begin{cases} t_1^* \frac{-P^k}{\left\|P^k\right\|_F}, & \text{for } \left\|P^k\right\|_F \neq 0 , \\ 0 & \textit{otherwise} . \end{cases}$$
$$Z^{k+1} = \begin{cases} t_2^* \frac{-Q^k}{\left\|Q^k\right\|_F}, & \text{for } \left\|Q^k\right\|_F \neq 0 , \\ 0 & \textit{otherwise} . \end{cases}$$

We solve for t_1^* and t_2^* with the following two dimensional minimization problem

$$\operatorname*{argmin}_{t_1 \geq 0, t_2 \geq 0} \left\{ -t_1 \left\|P^k\right\|_F - t_2 \left\|Q^k\right\|_F + c_1 \left(\frac{t_1^2 + t_2^2}{2} \right)^2 + c_2 \left(\frac{t_1^2 + t_2^2}{2} \right) \right\} .$$

Thus, the solutions t_1^* and t_2^* are the non-negative real roots of the following equations

$$- \left\|P^k\right\|_F + c_1 (t_1^2 + t_2^2) t_1 + c_2 t_1 = 0 , \quad - \left\|Q^k\right\|_F + c_1 (t_1^2 + t_2^2) t_2 + c_2 t_2 = 0 .$$

Now, there are two methods to solve the above equations.

Method 1: Further simplifications lead to $t_1 = r\left\|P^k\right\|_F$ and $t_2 = r\left\|Q^k\right\|_F$ for some $r \geq 0$ such that r satisfies the following cubic equation

$$c_1\left(\left\|Q^k\right\|_F^2 + \left\|P^k\right\|_F^2\right)r^3 + c_2 r - 1 = 0\,.$$

Method 2: Further simplifications lead to $t_1 = r\frac{\sqrt{2}\|P^k\|_F}{\sqrt{\|P^k\|_F^2+\|Q^k\|_F^2}}$ and $t_2 = r\frac{\sqrt{2}\|Q^k\|_F}{\sqrt{\|P^k\|_F^2+\|Q^k\|_F^2}}$ for some $r \geq 0$ such that r satisfies the following cubic equation

$$2c_1r^3 + c_2r - \frac{\sqrt{\|P^k\|_F^2 + \|Q^k\|_F^2}}{\sqrt{2}} = 0\,.$$

□

C.2.1 Extensions to L2-regularized matrix factorization

We consider the following L2-regularized matrix factorization problem [104].

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A - UZ\|_F^2 + \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)\right\}\,. \tag{C.2.5}$$

Denote $f_1 := \frac{1}{2}\|A-UZ\|_F^2$, $f_0 := \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)$ and $h = h_a$.

Proposition C.2.1.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = -r\,P^k$, $Z^{k+1} = -r\,Q^k$ where r is the non-negative real root of*

$$c_1\left(\left\|Q^k\right\|_F^2 + \left\|P^k\right\|_F^2\right)r^3 + (c_2+\lambda_0)r - 1 = 0\,, \tag{C.2.6}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

We skip the proof as it is very similar to Proposition C.2.0.1 and only change is in c_2.

C.2.2 Extensions to graph regularized matrix factorization

Graph regularized matrix factorization was proposed in [38]. However, they used non-negativity constraints. We simplify the problem here by not considering the non-negativity constraints. We later show in Section C.3.3, how the non-negativity constraints are handled. Here, given $\mathcal{L} \in \mathbb{R}^{M\times M}$ we are interested to solve

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A - UZ\|_F^2 + \frac{\mu_0}{2}tr(U^T\mathcal{L}U) + \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)\right\}\,.$$

In such a case, it is easy to extend the following ideas to Graph regularized non-negative matrix factorization. We show here L-smad property. We first need the following technical lemma.

Lemma C.2.2.1. *Let $g_1(U) = tr(U^T\mathcal{L}U)$, then for any $H \in \mathbb{R}^{M\times K}$ we have $\nabla g_1(U) = \mathcal{L}U + \mathcal{L}^TU$,*

$$\left\langle H, \nabla^2 g_1(U)H\right\rangle = 2\left\langle \mathcal{L}H, H\right\rangle\,.$$

Proof. Note that $tr(U^T\mathcal{L}U) = \langle \mathcal{L}U, U\rangle$, now we obtain for $H \in \mathbb{R}^{M\times K}$ the following

$$\begin{aligned}\langle \mathcal{L}(U+H), U+H\rangle &= \langle \mathcal{L}(U+H), U+H\rangle \\ &= \langle \mathcal{L}U, U\rangle + \langle \mathcal{L}U, H\rangle + \langle \mathcal{L}H, U\rangle + \langle \mathcal{L}H, H\rangle \ , \\ &= \langle \mathcal{L}U, U\rangle + \langle \mathcal{L}U, H\rangle + \left\langle \mathcal{L}^T U, H\right\rangle + \langle \mathcal{L}H, H\rangle \ .\end{aligned}$$

Thus the statement holds, by collecting the first and second order terms. □

Now, we prove the L-smad property.

Proposition C.2.2.1. *Let $f_1(U,Z) = \frac{1}{2}\|A-UZ\|_F^2 + \frac{\mu_0}{2} tr(U^T\mathcal{L}U)$. Then, for a certain constant $L \geq 1$, the function f_1 satisfies L-smad property with respect to the following kernel generating distance,*

$$h_c(U,Z) = 3h_1(U,Z) + (\|A\|_F + \mu_0\|\mathcal{L}\|_F)\, h_2(U,Z)\,.$$

Proof. The proof is similar to Proposition 4.5.0.1 and Lemma C.2.2.1 must be applied for the result. □

Denote $f_1 := \frac{1}{2}\|A-UZ\|_F^2 + \frac{\mu_0}{2} tr(U^T\mathcal{L}U)$, $f_0 := \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)$ and $h = h_c$.

Proposition C.2.2.2. *In BPG-MF, with the above defined f_0, f_1, h the update steps in each iteration are given by $U^{k+1} = -r\,P^k$, $Z^{k+1} = -r\,Q^k$ where $r \geq 0$ and satisfies*

$$c_1\left(\left\|Q^k\right\|_F^2 + \left\|P^k\right\|_F^2\right) r^3 + (c_2 + \mu_0\|\mathcal{L}\|_F + \lambda_0)r - 1 = 0\,, \tag{C.2.7}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

The proof is similar to Proposition C.2.0.1 and only c_2 changes.

C.2.3 Extensions to L1-regularized matrix factorization

Now consider the following matrix factorization problem with L1-regularization

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A-UZ\|_F^2 + \lambda_1(\|U\|_1 + \|Z\|_1)\right\}\,. \tag{C.2.8}$$

Recall that soft-thresholding operator is defined for any $y \in \mathbb{R}^N$ by

$$\mathcal{S}_\theta(y) = \text{argmin}_{x\in\mathbb{R}^N}\left\{\theta\|x\|_1 + \frac{1}{2}\|x-y\|^2\right\} = \max\{|y|-\theta, 0\}\,\text{sgn}(y)\ , \tag{C.2.9}$$

where $\theta > 0$ and the operations are applied element-wise. We require the following technical result.

Lemma C.2.3.1. *Let $Q \in \mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t_0 > 0$ and let $t \geq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q, X\rangle + t_0\|X\|_1 : \|X\|_F^2 \leq t^2\right\} = -t\,\|S_{t_0}(-Q)\|_F\ .$$

with the minimizer at $X^* = t\frac{S_{t_0}(-Q)}{\|S_{t_0}(-Q)\|_F}$ for $\|S_{t_0}(-Q)\|_F \neq 0$ and otherwise all X such that $\|X\|_F^2 \leq t^2$ are minimizers. Moreover we have the following equivalence,

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q, X\rangle + t_0\|X\|_1 : \|X\|_F^2 \leq t^2\right\} \equiv \min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q, X\rangle + t_0\|X\|_1 : \|X\|_F^2 = t^2\right\}\,. \tag{C.2.10}$$

Proof. We have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle+t_0\left\|X\right\|_1:\left\|X\right\|_F^2\leq t^2\right\}\equiv-\max_{X\in\mathbb{R}^{A\times B}}\left\{\langle -Q,X\rangle-t_0\left\|X\right\|_1:\left\|X\right\|_F^2\leq t^2\right\}.$$

Then the result follows due to [111, Proposition 14] with the minimizer at $X^*=t\frac{S_{t_0}(-Q)}{\left\|S_{t_0}(-Q)\right\|_F}$ for $\left\|S_{t_0}(-Q)\right\|_F\neq 0$ and 0 otherwise. The equivalence statement in (C.2.10) follows as $\left\|X^*\right\|_F^2=t^2$ for $\left\|S_{t_0}(-Q)\right\|_F\neq 0$ and otherwise all the points satisfying $\left\|X\right\|_F^2=t^2$ are minimizers. □

Denote $f_1:=\frac{1}{2}\left\|A-UZ\right\|_F^2$, $f_0:=\lambda_1\left(\left\|U\right\|_1+\left\|Z\right\|_1\right)$ and $h=h_a$.

Proposition C.2.3.1. *In BPG-MF, with the above defined f_1,f_0,h the update steps in each iteration are given by $U^{k+1}=r\mathcal{S}_{\lambda_1\lambda}(-P^k)$, $Z^{k+1}=r\mathcal{S}_{\lambda_1\lambda}(-Q^k)$ where $r\geq 0$ and satisfies*

$$c_1\left(\left\|\mathcal{S}_{\lambda_1\lambda}\left(-Q^k\right)\right\|_F^2+\left\|\mathcal{S}_{\lambda_1\lambda}\left(-P^k\right)\right\|_F^2\right)r^3+c_2r-1=0\,, \tag{C.2.11}$$

with $c_1=3$ and $c_2=\left\|A\right\|_F$.

Proof. The proof is similar to that of Proposition C.2.0.1, however with certain changes due to the L1 norm in the objective. Consider the following subproblem

$$\begin{aligned}(U^{k+1},Z^{k+1})\in\underset{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}}{\operatorname{argmin}}&\left\{\lambda\lambda_1\left(\left\|U\right\|_1+\left\|Z\right\|_1\right)+\left\langle P^k,U\right\rangle+\left\langle Q^k,Z\right\rangle\right.\\&\left.+c_1\left(\frac{\left\|U\right\|_F^2+\left\|Z\right\|_F^2}{2}\right)^2+c_2\left(\frac{\left\|U\right\|_F^2+\left\|Z\right\|_F^2}{2}\right)\right\},\end{aligned}$$

Denote the objective in the above minimization problem as $\mathcal{O}(\mathcal{U}^{\|},\mathcal{Z}^{\|})$. Now, we show that the following holds

$$\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}}\left(\mathcal{O}(\mathcal{U}^{\|},\mathcal{Z}^{\|})\right)\equiv\min_{t_1\geq0,t_2\geq0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N},\left\|U\right\|_F=t_1,\left\|Z\right\|_F=t_2}\left(\mathcal{O}(\mathcal{U}^{\|},\mathcal{Z}^{\|})\right)\right\}, \tag{C.2.12}$$

$$\equiv\min_{t_1\geq0,t_2\geq0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N},\left\|U\right\|_F\leq t_1,\left\|Z\right\|_F\leq t_2}\left(\mathcal{O}(\mathcal{U}^{\|},\mathcal{Z}^{\|})\right)\right\}. \tag{C.2.13}$$

where the first step is a simple rewriting of the objective. The second step is non-trivial. In order to prove (C.2.13) we rewrite (C.2.12) as

$$\begin{aligned}\min_{t_1\geq0,t_2\geq0}&\left\{\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k,U_1\right\rangle+\lambda\lambda_1\left\|U\right\|_1:\left\|U_1\right\|_F^2=t_1^2\right\}+\min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k,Z_1\right\rangle+\lambda\lambda_1\left\|Z\right\|_1:\left\|Z_1\right\|_F^2=t_2^2\right\}\right.\\&\left.+c_1\left(\frac{t_1^2+t_2^2}{2}\right)^2+c_2\left(\frac{t_1^2+t_2^2}{2}\right)\right\}.\end{aligned}$$

where the second step (C.2.13) uses Lemma C.2.3.1 and strong convexity of h. Now, note the following equivalence due to Lemma C.2.3.1

$$\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k,U_1\right\rangle+\lambda\lambda_1\left\|U\right\|_1:\left\|U_1\right\|_F^2=t_1^2\right\}\equiv\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k,U_1\right\rangle+\lambda\lambda_1\left\|U\right\|_1:\left\|U_1\right\|_F^2\leq t_1^2\right\}, \tag{C.2.14}$$

$$\min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k,Z_1\right\rangle+\lambda\lambda_1\left\|Z\right\|_1:\left\|Z_1\right\|_F^2=t_2^2\right\}\equiv\min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k,Z_1\right\rangle+\lambda\lambda_1\left\|Z\right\|_1:\left\|Z_1\right\|_F^2\le t_2^2\right\}. \tag{C.2.15}$$

We solve the subproblems via the following strategy. Denote

$$U_1^*(t_1)\in\operatorname{argmin}\left\{\left\langle P^k,U_1\right\rangle+\lambda\lambda_1\left\|U\right\|_1:U_1\in\mathbb{R}^{M\times K},\left\|U_1\right\|_F^2\le t_1^2\right\}$$

$$Z_1^*(t_2)\in\operatorname{argmin}\left\{\left\langle Q^k,Z_1\right\rangle+\lambda\lambda_1\left\|Z\right\|_1:Z_1\in\mathbb{R}^{K\times N},\left\|Z_1\right\|_F^2\le t_2^2\right\}$$

Then we obtain $(U^{k+1},Z^{k+1})=(U_1^*(t_1^*),Z_1^*(t_2^*))$, where t_1^* and t_2^* are obtained by solving the following two dimensional subproblem

$$\begin{aligned}(t_1^*,t_2^*)\in\operatorname*{argmin}_{t_1\ge0,t_2\ge0}&\left\{\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k,U_1\right\rangle+\lambda\lambda_1\left\|U\right\|_1:\left\|U_1\right\|_F^2\le t_1^2\right\}\right.\\&\left.+\min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k,Z_1\right\rangle+\lambda\lambda_1\left\|Z\right\|_1:\left\|Z_1\right\|_F^2\le t_2^2\right\}+c_1\left(\frac{t_1^2+t_2^2}{2}\right)^2+c_2\left(\frac{t_1^2+t_2^2}{2}\right)\right\}.\end{aligned}$$

Note that inner minimization subproblems can be trivially solved once we obtain $U_1^*(t_1)$ and $Z_1^*(t_2)$. Due to Lemma C.2.3.1 we obtain the solution to the subproblem in each iteration as follows

$$U^{k+1}=\begin{cases}t_1^*\frac{S_{\lambda\lambda_1}(-P^k)}{\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F}, & \text{for }\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F\neq0\,,\\0 & \textit{otherwise}\,.\end{cases}$$

$$Z^{k+1}=\begin{cases}t_2^*\frac{S_{\lambda\lambda_1}(-Q^k)}{\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F}, & \text{for }\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F\neq0\,,\\0 & \textit{otherwise}\,.\end{cases}$$

We solve for t_1^* and t_2^* with the following two dimensional minimization problem

$$\operatorname*{argmin}_{t_1\ge0,t_2\ge0}\left\{-t_1\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F-t_2\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F+c_1\left(\frac{t_1^2+t_2^2}{2}\right)^2+c_2\left(\frac{t_1^2+t_2^2}{2}\right)\right\}.$$

Thus, the solutions t_1^* and t_2^* are the non-negative real roots of the following equations

$$-\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F+c_1(t_1^2+t_2^2)t_1+c_2t_1=0\,,\quad-\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F+c_1(t_1^2+t_2^2)t_2+c_2t_2=0\,.$$

Set $t_1=r\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F$ and $t_2=r\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F$ for some $r\ge0$. This results in the following cubic equation,

$$c_1\left(\left\|S_{\lambda\lambda_1}(-Q^k)\right\|_F^2+\left\|S_{\lambda\lambda_1}(-P^k)\right\|_F^2\right)r^3+c_2r-1=0\,,$$

where the solution is the non-negative real root. □

C.2.4 Extensions with nuclear norm regularization

We start with the notion of Singular Value Shrinkage Operator [39], where given a matrix $Q\in\mathbb{R}^{A\times B}$ of rank K with Singular Value Decomposition given by $U\Sigma V^T$ with $U\in\mathbb{R}^{A\times K}$, $\Sigma\in\mathbb{R}^{K\times K}$ and $V\in\mathbb{R}^{K\times N}$ for $t\ge0$

the output is

$$\mathcal{D}_t(Q) = U\mathcal{S}_t(\Sigma)V^T\,, \tag{C.2.16}$$

where the soft-thresholding operator is applied only to the singular values. Before we proceed, we require the following technical lemma.

Lemma C.2.4.1. *Let $Q \in \mathbb{R}^{A\times B}$ of rank K with Singular Value Decomposition given by $U\Sigma V^T$ with $U \in \mathbb{R}^{A\times K}$, $\Sigma \in \mathbb{R}^{K\times K}$ and $Z \in \mathbb{R}^{K\times N}$. Let $t \geq 0$ and $\|Q\|_F \neq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle + t_0 \|X\|_* : \|X\|_F^2 \leq t^2\right\} = -t\,\|\mathcal{S}_{t_0}(-\Sigma)\|\;.$$

with $X^* = t\frac{\mathcal{D}_{t_0}(-Q)}{\|\mathcal{D}_t(-Q)\|_F}$ if $\|\mathcal{D}_{t_0}(-Q)\| \neq 0$ else any X such that $\|X\|_F^2 \leq t^2$ is a minimizer. Moreover we have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle + t_0 \|X\|_* : \|X\|_F^2 \leq t^2\right\} = \min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle + t_0 \|X\|_* : \|X\|_F^2 = t^2\right\}\,. \tag{C.2.17}$$

Proof. The sub-differential of the nuclear norm [39] is given by

$$\partial \|X\|_* = \left\{UV^T + W : W \in \mathbb{R}^{A\times B}, U^TW = 0, WV = 0, \|W\|_2 \leq 1\right\}\,. \tag{C.2.18}$$

The normal cone for the set $C_1 = \left\{X : \|X\|_F^2 \leq t^2\right\}$ is given by

$$\mathcal{N}_{C_1}(\bar{X}) = \left\{V \in \mathbb{R}^{A\times B} : \left\langle V, X - \bar{X}\right\rangle \leq 0 \text{ for all } X \in C_1\right\} \equiv \left\{\theta\bar{X} : \theta \geq 0\right\}\,.$$

We consider the following problem

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle + t_0 \|X\|_* : \|X\|_F^2 \leq t^2\right\}\,.$$

and the optimality condition [150, Theorem 10.1, p. 422] results in

$$0 \in Q + t_0\partial \|X\|_* + \mathcal{N}_{C_1}(X)\,.$$

We follow the strategy from [39, Theorem 2.1]. One can decompose $-Q$ as

$$-Q = U_0\Sigma_0V_0^T + U_1\Sigma_1V_1^T\,.$$

where U_0, V_0 contain the singular vectors for singular values greater than t_0 and U_1, V_1 for less than equal to t_0. Then with $X = U_0\Sigma V_0^T$, the optimality condition becomes

$$0 = Q + t_0(U_0V_0^T + W) + \theta U_0\Sigma V_0^T\,, \tag{C.2.19}$$

and thus we obtain

$$U_0\Sigma_0V_0^T + U_1\Sigma_1V_1^T = t_0\left(U_0V_0^T + W\right) + \theta U_0\Sigma V_0^T\,.$$

With $W = t_0^{-1}U_1\Sigma_1V_1^T$ all the conditions in (C.2.18) are satisfied. For some unknown $\theta \geq 0$ we have

$$\theta\Sigma = \Sigma_0 - t_0I\,.$$

The objective $\langle Q, X\rangle + t_0 \|X\|_*$ is now monotonically decreasing with θ after substituting. Thus, we obtain the solution $X = \frac{t}{\|\Sigma_0 - t_0 I\|} U_0 \left(\Sigma_0 - t_0 I\right) V_0^T$ for $\|\Sigma_0 - t_0 I\| \neq 0$ else the solution is 0. The equivalence statement in (C.2.17) follows trivially because if $\|\Sigma_0 - t_0 I\| \neq 0$ we have $\|X\|_F^2 = t^2$ otherwise all the points satisfying $\|X\|_F^2 \leq t^2$ are minimizers. □

Here, we want to solve matrix factorization problem with nuclear norm regularization, where for certain constant $\lambda_2 > 0$ we want to solve

$$\min_{U \in \mathbb{R}^{M\times K}, Z \in \mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A - UZ\|_F^2 + \lambda_2 \left(\|U\|_* + \|Z\|_*\right) \right\} . \tag{C.2.20}$$

Denote $f_1 := \frac{1}{2}\|A - UZ\|_F^2$, $f_0 := \lambda_2 \left(\|U\|_* + \|Z\|_*\right)$ and $h = h_a$.

Proposition C.2.4.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = r\mathcal{D}_{\lambda_1\lambda}(-P^k)$, $Z^{k+1} = r\mathcal{D}_{\lambda_1\lambda}(-Q^k)$ where $r \geq 0$ and satisfies*

$$c_1 \left(\left\|\mathcal{D}_{\lambda_1\lambda}\left(-Q^k\right)\right\|_F^2 + \left\|\mathcal{D}_{\lambda_1\lambda}\left(-P^k\right)\right\|_F^2 \right) r^3 + c_2 r - 1 = 0 \, , \tag{C.2.21}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

The proof is similar to Proposition C.2.3.1, however Lemma C.2.4.1 must be used instead of Lemma C.2.3.1.

C.2.5 Extensions with non-convex sparsity constraints

We want to solve the matrix factorization problem with non-convex sparsity constraints [26]

$$\min_{U \in \mathbb{R}^{M\times K}, Z \in \mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A - UZ\|_F^2 : \|U\|_0 \leq s_1, \|Z\|_0 \leq s_2, \right\} . \tag{C.2.22}$$

The problem with additional non-negativity constraints, the so called Sparse NMF is considered in Section C.3.5. Now, denote $f_1 := \frac{1}{2}\|A - UZ\|_F^2$, $f_0 := I_{\|U\|_0 \leq s_1} + I_{\|Z\|_0 \leq s_2}$ and $h = h_a$. Note that the Assumption D(iii) is not valid here, hence CoCaIn BPG-MF theory does not hold and hints at possible extensions of CoCaIn BPG-MF, which is an interesting open question. Before, we proceed, we require the following concept. Let $y \in \mathbb{R}^N$ and without loss of generality we can assume that $|y_1| \geq |y_2| \geq \ldots \geq |y_d|$, then the hard-thresholding operator [111] is given by

$$\mathcal{H}_s\left(y\right) = \mathrm{argmin}_{x \in \mathbb{R}^N} \left\{ \|x - y\|^2 : \|x\|_0 \leq s \right\} = \begin{cases} y_i, & i \leq s, \\ 0, & \text{otherwise}, \end{cases} \tag{C.2.23}$$

where $s > 0$ and the operations are applied element-wise. We require the following technical lemma.

Lemma C.2.5.1. *Let $Q \in \mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t \geq 0$ and $\|Q\|_F \neq 0$ then*

$$\min_{X \in \mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s \right\} = -t\left\|\mathcal{H}_s(-Q)\right\| .$$

with the minimizer $X^* = \frac{t\mathcal{H}_s(-Q)}{\|\mathcal{H}_s(-Q)\|}$ if $\|\mathcal{H}_s(-Q)\| \neq 0$ else $X^* = 0$. Moreover we have the following equivalence

$$\min_{X \in \mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s \right\} = \min_{X \in \mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 = t^2, \|X\|_0 \leq s \right\} .$$

Proof. The proof is similar to [111, Proposition 11]. We have

$$\begin{aligned}\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle : \|X\|_F^2\le t^2, \|X\|_0\le s\right\} &= -\max_{X\in\mathbb{R}^{A\times B}}\left\{\langle -Q,X\rangle : \|X\|_F^2\le t^2, \|X\|_0\le s\right\}\,,\\ &= -\max_{X\in\mathbb{R}^{A\times B}}\left\{\langle \mathcal{H}_s(-Q),X\rangle : \|X\|_F^2\le t^2\right\}\,.\end{aligned}$$

The first equality is a simple rewriting of the objective. Then, the corresponding objective $\langle -Q, X\rangle$ can be maximized with $\sum_{i=1}^{A}\sum_{j=1}^{B} I_{(i,j)\in\Omega_0}(-Q_{ij}X_{ij})$ where Ω_0 is set of index pairs and $I_{(i,j)\in\Omega_0}$ is 1 if the index pair if $(i,j)\in\Omega_0$ and zero otherwise. Note that the objective $\langle -Q, X\rangle$ is maximized if Ω_0 contains all the index pairs corresponding to the elements of $-Q$ with highest absolute value which is captured by Hard-thresholding operator. Thus, the second equality follows and the solution follows due to Lemma D.2.0.1. The equivalence statement follows as $\|X^*\|_F^2 = t^2$ for $\|\mathcal{H}_s(-Q)\|\neq 0$ else the function value is zero and is attained by all the points in the set $\left\{X : \|X\|_F^2\le t^2\right\}$ are minimizers, hence the equivalence. □

Proposition C.2.5.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = r\mathcal{H}_{s_1}(-P^k)$, $Z^{k+1} = r\mathcal{H}_{s_2}(-Q^k)$ where $r\ge 0$ and satisfies*

$$c_1\left(\left\|\mathcal{H}_{s_1}\left(-Q^k\right)\right\|_F^2 + \left\|\mathcal{H}_{s_2}\left(-P^k\right)\right\|_F^2\right) r^3 + c_2 r - 1 = 0\,, \tag{C.2.24}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

The proof is similar to Proposition C.2.3.1, however Lemma C.2.5.1 must be used instead of Lemma C.2.3.1.

C.3 Closed form solutions: Part II for NMF variants

For simplicity we consider the following problem [100, 101]

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}}\left\{f(U,Z) := \frac{1}{2}\|A-UZ\|_F^2 + I_{U\ge 0} + I_{Z\ge 0}\right\}\,. \tag{C.3.1}$$

We set $\mathcal{R}_1(U)=0$, $\mathcal{R}_2(Z)=0$, $f_1 = f$ and $f = I_{U\ge 0} + I_{Z\ge 0}$ where I is the indicator operator. We start with the following technical lemma.

Lemma C.3.0.1. *Let $Q\in\mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t\ge 0$ and $\|Q\|_F\neq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle : \|X\|_F^2\le t^2, X\ge 0\right\} = -t\left\|\Pi_+(-Q)\right\|_F\,,$$

with the minimizer $X^* = t\frac{\Pi_+(-Q)}{\|\Pi_+(-Q)\|_F}$ if $\|\Pi_+(-Q)\|_F\neq 0$ else $X^*=0$. For $\|\Pi_+(-Q)\|_F\neq 0$, we have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle : \|X\|_F^2\le t^2, X\ge 0\right\} \equiv \min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle : \|X\|_F^2 = t^2, X\ge 0\right\}\,. \tag{C.3.2}$$

Proof. On rewriting we have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}}\left\{\langle Q,X\rangle : \|X\|_F^2\le t^2, X\ge 0\right\} \equiv -\max_{X\in\mathbb{R}^{A\times B}}\left\{\langle -Q,X\rangle : \|X\|_F^2\le t^2, X\ge 0\right\}\,.$$

The expression $\langle -Q, X\rangle$ is maximized at $X^* = c\Pi_+(-Q)$ for certain constant c. On substituting we have

$$\langle -Q, X^*\rangle = c\left\|\Pi_+(-Q)\right\|_F^2 .$$

Since, the dependence on c is linear and we additionally require $\|X\|_F^2 \le t^2$, we can set $c = \frac{t}{\|\Pi_+(-Q)\|_F}$ if $\|\Pi_+(-Q)\|_F \neq 0$ else $c = 0$. Hence, the minimizer to

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 \le t^2\right\}$$

is attained at $X^* = -t\frac{\Pi_+(-Q)}{\|\Pi_+(-Q)\|_F}$ for $\|\Pi_+(-Q)\|_F \neq 0$ else $X^* = 0$. The equivalence in the statement follows as $\|X^*\|_F^2 = t^2$. □

Denote $f_1 = f$, $f_0 = I_{U\ge 0} + I_{Z\ge 0}$ and $h = h_a$.

Proposition C.3.0.1. *In BPG-MF, when $f_1 = f$ in (C.3.1) the update step in each iteration are given by $U^{k+1} = \Pi_+(-P^k)$, $Z^{k+1} = \Pi_+(-Q^k)$ where $r \ge 0$ and satisfies*

$$c_1\left(\left\|\Pi_+(-Q^k)\right\|_F^2 + \left\|\Pi_+(-P^k)\right\|_F^2\right)r^3 + c_2 r - 1 = 0\,., \tag{C.3.3}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

Proof. The proof is similar to that of Proposition C.2.0.1, however with certain changes due to the involved non-negativity constraints for the objective. Consider the following subproblem

$$(U^{k+1}, Z^{k+1}) \in \underset{(U,Z)\in\mathbb{R}_+^{M\times K}\times\mathbb{R}_+^{K\times N}}{\operatorname{argmin}} \left\{\left\langle P^k, U\right\rangle + \left\langle Q^k, Z\right\rangle + c_1\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)^2 + c_2\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)\right\}.$$

Denote the objective in the above minimization problem as $\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})$. Now, we show that the following holds

$$\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right) \equiv \min_{t_1\ge 0, t_2\ge 0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F = t_1, \|Z\|_F = t_2} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\}, \tag{C.3.4}$$

$$\equiv \min_{t_1\ge 0, t_2\ge 0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F \le t_1, \|Z\|_F \le t_2} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\}, \tag{C.3.5}$$

where the first step is a simple rewriting of the objective and involved variables and the second equivalence proof is similar to that equivalence of (C.2.13) and (C.2.12) in Proposition C.2.3.1, which we describe now. The second step is non-trivial. In order to prove (C.3.5) we rewrite (C.3.4) as

$$\min_{t_1\ge 0, t_2\ge 0}\left\{\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 = t_1^2, U_1 \ge 0\right\} + \min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 = t_2^2, Z_1 \ge 0\right\}\right.$$
$$\left. + c_1\left(\frac{t_1^2 + t_2^2}{2}\right)^2 + c_2\left(\frac{t_1^2 + t_2^2}{2}\right)\right\}.$$

where the second step uses Lemma C.3.0.1 and strong convexity of h. Now, due to Lemma C.2.3.1, if $\left\|\Pi_+(-P^k)\right\|_F \neq 0$ we have

$$\min_{U_1\in\mathbb{R}^{M\times K}} \left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 = t_1^2\,, U_1 \geq 0\right\} \equiv \min_{U_1\in\mathbb{R}^{M\times K}} \left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 \leq t_1^2\,, U_1 \geq 0\right\}\,, \tag{C.3.6}$$

and similarly if $\left\|\Pi_+(-Q^k)\right\|_F \neq 0$ we have

$$\min_{Z_1\in\mathbb{R}^{K\times N}} \left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 = t_2^2\,, Z_1 \geq 0\right\} \equiv \min_{Z_1\in\mathbb{R}^{K\times N}} \left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 \leq t_2^2\,, Z_1 \geq 0\right\}\,. \tag{C.3.7}$$

Note that if $\left\|\Pi_+(-P^k)\right\|_F = 0$ and $\left\|P^k\right\|_F \neq 0$ then the objective

$$\min_{U_1\in\mathbb{R}^{M\times K}} \left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 = t_1^2\,, U_1 \geq 0\right\}$$

with minimum function value of a positive value $t_1 \min_{i\in[M],\,j\in[K]}\{(P^k)_{i,j}\}$ where we have $[A] = \{1, 2, \dots, A\}$ for a positive integer A. Similarly if $\left\|\Pi_+(-Q^k)\right\|_F = 0$ and $\left\|Q^k\right\|_F \neq 0$ the minimum function value for

$$\min_{Z_1\in\mathbb{R}^{K\times N}} \left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 = t_2^2\,, Z_1 \geq 0\right\}$$

is a positive value $t_2 \min_{i\in[K],\,j\in[N]}\{(Q^k)_{i,j}\}$. Thus for $\left\|P^k\right\|_F \neq 0$ with $\left\|\Pi_+(-P^k)\right\|_F = 0$ (or $\left\|Q^k\right\|_F \neq 0$ with $\left\|\Pi_+(-Q^k)\right\|_F = 0$) the final objective (C.3.4) is monotonically increasing in t_1 (or t_2) which will drive t_1 (or t_2) to 0 due to the constraint $t_1 \geq 0$ (or $t_2 \geq 0$). So, without loss of generality we can consider $\left\|\Pi_+(-Q^k)\right\|_F \neq 0$ and $\left\|\Pi_+(-Q^k)\right\|_F = 0$. Now, we obtain the solutions via the following strategy. Denote

$$U_1^*(t_1) \in \operatorname{argmin} \left\{\left\langle P^k, U_1\right\rangle : U_1 \in \mathbb{R}_+^{M\times K}\,, \|U_1\|_F^2 \leq t_1^2\right\}\,,$$

$$Z_1^*(t_2) \in \operatorname{argmin} \left\{\left\langle Q^k, Z_1\right\rangle : Z_1 \in \mathbb{R}_+^{K\times N}\,, \|Z_1\|_F^2 \leq t_2^2\right\}\,.$$

Then we obtain $(U^{k+1}, Z^{k+1}) = (U_1^*(t_1^*), Z_1^*(t_2^*))$, where t_1^* and t_2^* are obtained by solving the following two dimensional subproblem

$$\begin{aligned}(t_1^*, t_2^*) \in \operatorname*{argmin}_{t_1\geq 0, t_2\geq 0} \Bigg\{ &\min_{U_1\in\mathbb{R}_+^{M\times K}} \left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 \leq t_1^2\right\} + \min_{Z_1\in\mathbb{R}_+^{K\times N}} \left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 \leq t_2^2\right\} \\ &+ c_1\left(\frac{t_1+t_2}{2}\right)^2 + c_2\left(\frac{t_1+t_2}{2}\right)\Bigg\}\,.\end{aligned}$$

Note that inner minimization subproblems can be trivially solved once we obtain $U_1^*(t_1)$ and $Z_1^*(t_2)$. Due to Lemma C.3.0.1 we obtain the solution to the subproblem in each iteration as follows

$$U^{k+1} = \begin{cases} t_1^* \frac{\Pi_+(-P^k)}{\left\|\Pi_+(-P^k)\right\|_F}, & \text{for } \left\|\Pi_+(-P^k)\right\|_F \neq 0\,, \\ 0, & \textit{otherwise}\,. \end{cases}$$

$$Z^{k+1} = \begin{cases} t_2^* \frac{\Pi_+(-Q^k)}{\left\|\Pi_+(-Q^k)\right\|_F}, & \text{for } \left\|\Pi_+(-Q^k)\right\|_F \neq 0\,, \\ 0, & \textit{otherwise}\,. \end{cases}$$

We solve for t_1^* and t_2^* with the following two dimensional minimization problem

$$\underset{t_1 \geq 0, t_2 \geq 0}{\operatorname{argmin}} \left\{ -t_1 \left\|\Pi_+(-P^k)\right\|_F - t_2 \left\|\Pi_+(-Q^k)\right\|_F + c_1 \left(\frac{t_1^2 + t_2^2}{2}\right)^2 + c_2 \left(\frac{t_1^2 + t_2^2}{2}\right) \right\}\,.$$

Thus, the solutions t_1^* and t_2^* are the non-negative real roots of the following equations

$$-\left\|\Pi_+(-P^k)\right\|_F + c_1(t_1^2 + t_2^2)t_1 + c_2 t_1 = 0\,, \quad -\left\|\Pi_+(-Q^k)\right\|_F + c_1(t_1^2 + t_2^2)t_2 + c_2 t_2 = 0\,.$$

Further simplifications lead to $t_1 = r\left\|\Pi_+(-P^k)\right\|_F$ and $t_2 = r\left\|\Pi_+(-Q^k)\right\|_F$ for some $r \geq 0$. This results in the following cubic equation,

$$c_1 \left(\left\|\Pi_+(-Q^k)\right\|_F^2 + \left\|\Pi_+(-P^k)\right\|_F^2 \right) r^3 + c_2 r - 1 = 0\,,$$

where the solution is the non-negative real root. □

C.3.1 Extensions to L2-regularized NMF

Here, the goal is solve the following minimization problem

$$\min_{U \in \mathbb{R}^{M \times K}, Z \in \mathbb{R}^{K \times N}} \left\{ f(U, Z) := \frac{1}{2}\left\|A - UZ\right\|_F^2 + \frac{\lambda_0}{2}\left(\left\|U\right\|_F^2 + \left\|Z\right\|_F^2\right) + I_{U \geq 0} + I_{Z \geq 0} \right\}\,.$$

Denote $f_1 := \frac{1}{2}\left\|A - UZ\right\|_F^2 + \frac{\lambda_0}{2}\left(\left\|U\right\|_F^2 + \left\|Z\right\|_F^2\right)$, $f_0 := I_{U \geq 0} + I_{Z \geq 0}$ and $h = h_b$.

Proposition C.3.1.1. *In BPG-MF, with above defined f_1, f_0, h the update step in each iteration are given by $U^{k+1} = \Pi_+(-P^k)$, $Z^{k+1} = \Pi_+(-Q^k)$ where $r \geq 0$ and satisfies*

$$c_1 \left(\left\|\Pi_+(-Q^k)\right\|_F^2 + \left\|\Pi_+(-P^k)\right\|_F^2 \right) r^3 + (c_2 + \lambda_0) r - 1 = 0\,,$$

with $c_1 = 3$ and $c_2 = \left\|A\right\|_F$.

The proof is similar to Proposition C.3.0.1 with only change in c_2.

C.3.2 Extensions to L1-regularized NMF

Here, the goal is solve the following minimization problem

$$\min_{U\in\mathbb{R}^{M\times K},Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\left\|A-UZ\right\|_F^2 + \lambda_1\left(\left\|U\right\|_1 + \left\|Z\right\|_1\right) + I_{U\geq 0} + I_{Z\geq 0} \right\}.$$

We denote e_D to be a vector of dimension D with all its elements set to 1.

Lemma C.3.2.1. *Let $Q\in\mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t\geq 0$ and $\left\|Q\right\|_F\neq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{ \langle Q,X\rangle + t_0\left\|X\right\|_1 : \left\|X\right\|_F^2\leq t^2, X\geq 0 \right\} = -t\left\|\Pi_+(-\left(Q+t_0e_Ae_B{}^T\right))\right\|_F$$

with the minimizer $X^* = t\frac{\Pi_+(-\left(Q+t_0e_Ae_B{}^T\right))}{\left\|\Pi_+(-\left(Q+t_0e_Ae_B{}^T\right))\right\|_F}$ if the condition $\left\|\Pi_+(-\left(Q+t_0e_Ae_B{}^T\right))\right\|_F\neq 0$ holds.

Proof. By using $X\geq 0$ and the basic trace properties we have the following equivalence

$$\left\|X\right\|_1 = \sum_{i,j} X_{ij} = e_A{}^TXe_B = tr\left(e_A{}^TXe_B\right) = tr\left(e_Be_A{}^TX\right) = \left\langle e_Ae_B{}^T, X\right\rangle,$$

hence we have the following equivalence

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{ \langle Q,X\rangle + t_0\left\|X\right\|_1 : \left\|X\right\|_F^2\leq t^2, X\geq 0 \right\} \equiv \min_{X\in\mathbb{R}^{A\times B}} \left\{ \left\langle Q+t_0e_Ae_B{}^T,X\right\rangle : \left\|X\right\|_F^2\leq t^2, X\geq 0 \right\}$$

Now, the solution follows due to Lemma C.3.0.1. □

Denote $f_1 := \frac{1}{2}\left\|A-UZ\right\|_F^2$, $f_0 := \lambda_1\left(\left\|U\right\|_1+\left\|Z\right\|_1\right) + I_{U\geq 0} + I_{Z\geq 0}$ and $h = h_a$.

Proposition C.3.2.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = r\Pi_+(-\left(P^k+t_0e_Me_K^T\right))$, $Z^{k+1} = r\Pi_+(-\left(Q^k+t_0e_Ke_N^T\right))$ where $r\geq 0$ and satisfies*

$$c_1\left(\left\|\Pi_+(-\left(P^k+t_0e_Me_K^T\right))\right\|_F^2 + \left\|\Pi_+(-\left(Q^k+t_0e_Ke_N^T\right))\right\|_F^2\right)r^3 + c_2r - 1 = 0,$$

with $c_1 = 3$, $c_2 = \left\|A\right\|_F$ and $t_0 = \lambda\lambda_1$.

We skip the proof as it is similar to Proposition C.3.0.1.

C.3.3 Extensions to graph regularized non-negative matrix factorization

Graph regularized non-negative matrix factorization was proposed in [38]. Here, given $\mathcal{L}\in\mathbb{R}^{M\times M}$ we are interested to solve

$$\min_{U\in\mathbb{R}^{M\times K},Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) = \frac{1}{2}\left\|A-UZ\right\|_F^2 + \frac{\mu_0}{2}tr(U^T\mathcal{L}U) + \frac{\lambda_0}{2}\left(\left\|U\right\|_F^2+\left\|Z\right\|_F^2\right) + I_{U\geq 0} + I_{Z\geq 0} \right\}.$$

Recall that

$$h_c(U,Z) = 3h_1(U,Z) + \left(\left\|A\right\|_F + \mu_0\left\|\mathcal{L}\right\|_F\right)h_2(U,Z).$$

Denote $f_1 := \frac{1}{2}\left\|A-UZ\right\|_F^2 + \frac{\mu_0}{2}tr(U^T\mathcal{L}U)$, $f_0 := \frac{\lambda_0}{2}\left(\left\|U\right\|_F^2+\left\|Z\right\|_F^2\right) + I_{U\geq 0} + I_{Z\geq 0}$ and $h = h_c$.

Proposition C.3.3.1. *In BPG-MF, with the above defined f_0, f_1, h the update steps in each iteration are given by $U^{k+1} = r\Pi_+(-P^k)$, $Z^{k+1} = r\Pi_+(-Q^k)$ where $r \geq 0$ and satisfies*

$$c_1 \left(\left\| \Pi_+(-Q^k) \right\|_F^2 + \left\| \Pi_+(-P^k) \right\|_F^2 \right) r^3 + (c_2 + \mu_0 \|\mathcal{L}\|_F + \lambda_0) r - 1 = 0\,, \tag{C.3.8}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

The proof is similar to Proposition C.3.0.1 and only c_2 changes.

C.3.4 Extensions to symmetric NMF via non-symmetric relaxation.

In [177], the following optimization problem was proposed in the context of Symmetric NMF where the factors U and Z^T are equal. The symmetricity of the factors was lifted via a quadratic penalty terms resulting in the following problem

$$\min_{U \in \mathbb{R}^{M \times K}, Z \in \mathbb{R}^{K \times N}} \left\{ f(U,Z) := \frac{1}{2} \|A - UZ\|_F^2 + \frac{\lambda_0}{2} \|U - Z^T\|_F^2 + I_{U \geq 0} + I_{Z \geq 0} \right\}.$$

Now, we prove the L-smad property. We need the following technical lemma.

Lemma C.3.4.1. *Let $f_1(U,Z) = \frac{1}{2}\|A - UZ\|_F^2 + \frac{\lambda_0}{2}\|U - Z^T\|_F^2$ be as defined above, we have the following*

$$\nabla_U f_1(A, UZ) = \lambda_0 \left(U - Z^T \right) - (A - UZ)Z^T\,, \quad \nabla_Z f_1(A, UZ) = \lambda_0 \left(U - Z^T \right) + U^T(A - UZ)$$

and

$$\left\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2) \right\rangle = -2 \langle A - UZ, H_1 H_2 \rangle + \|UH_2 + H_1 Z\|_F^2 + \lambda_0 \left\| H_1 - {H_2}^T \right\|_F^2\,.$$

Proof. The first part of proof for function $\frac{1}{2}\|A - UZ\|_F^2$ follows from Proposition 4.5.0.1. For the other term, with the Forbenius dot product, we obtain

$$\frac{\lambda_0}{2} \|U + H_1 - Z^T - {H_2}^T\|_F^2 = \frac{\lambda_0}{2} \left(\|U - Z^T\|_F^2 + 2\langle U - Z^T, H_1 - {H_2}^T \rangle + \|H_1 - {H_2}^T\|_F^2 \right).$$

Combining with Lemma C.4.0.1, the statement follows from the collecting the first order and second order terms. □

Proposition C.3.4.1. *Let $f_1(U,Z) = \frac{1}{2}\|A - UZ\|_F^2 + \frac{\lambda_0}{2}\|U - Z\|_F^2$. Then, for a certain constant $L \geq 1$, the function f_1 satisfies L-smad property with respect to the following kernel generating distance,*

$$h_d(U,Z) = 3h_1(U,Z) + (\|A\|_F + 2\lambda_0)\, h_2(U,Z)\,.$$

Proof. The proof is similar to Proposition 4.5.0.1 and Lemma C.3.4.1 must be applied for the result. □

Denote $f_1 := \frac{1}{2}\|A - UZ\|_F^2 + \frac{\lambda_0}{2}\|U - Z\|_F^2$, $f_0 := I_{U \geq 0} + I_{Z \geq 0}$ and $h = h_d$.

Proposition C.3.4.2. *In BPG-MF, with the above defined update steps in each iteration are given by $U^{k+1} = r\Pi_+\left(-P^k\right)$, $Z^{k+1} = r\Pi_+\left(-Q^k\right)$ where $r \geq 0$ and satisfies*

$$c_1 \left(\left\| \Pi_+\left(-P^k\right) \right\|_F^2 + \left\| \Pi_+\left(-Q^k\right) \right\|_F^2 \right) r^3 + (c_2 + 2\lambda_0) r - 1 = 0\,, \tag{C.3.9}$$

with $c_1 = 3$ *and* $c_2 = \|A\|_F$.

The proof is similar to Proposition C.3.0.1 and only c_2 changes.

C.3.5 Extensions to NMF with non-convex sparsity constraints (Sparse NMF)

Consider the following problem from [26]

$$\min_{U\in\mathbb{R}^{M\times K},Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A-UZ\|_F^2 : U \geq 0, \|U\|_0 \leq s_1, Z\geq 0, \|Z\|_0 \leq s_2, \right\},$$

where s_1 and s_2 are two known positive integers. Denote $f_1 := \frac{1}{2}\|A-UZ\|_F^2$, $f_0 := I_{U\geq 0} + I_{\|U\|_0 \leq s_1} + I_{Z\geq 0} + I_{\|Z\|_0\leq s_2}$ and $h = h_a$. Note that the Assumption D(iii) is not valid here, hence CoCaIn BPG-MF theory does not hold and hints at possible extensions of CoCaIn BPG-MF, which is an interesting open question. We start with the following technical lemma.

Proposition C.3.5.1. *Let* $Q \in \mathbb{R}^{A\times B}$ *for some positive integers* A *and* B. *Let* $t \geq 0$ *and* $\|Q\|_F \neq 0$ *then*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s, X\geq 0\right\} = -t\|\mathcal{H}_s(\Pi_+(-Q))\|_F .$$

with the minimizer $X^* = t\frac{\mathcal{H}_s(\Pi_+(-Q))}{\|\mathcal{H}_s(\Pi_+(-Q))\|_F}$ *if* $\|\mathcal{H}_s(\Pi_+(-Q))\|_F \neq 0$ *else* $X^* = 0$. *If* $\|\mathcal{H}_s(\Pi_+(-Q))\|_F \neq 0$ *we have the following equivalence*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s, X\geq 0\right\} \equiv \min_{X\in\mathbb{R}^{A\times B}} \left\{ \langle Q, X\rangle : \|X\|_F^2 = t^2, \|X\|_0 \leq s, X\geq 0\right\}$$

Proof. We have

$$\begin{aligned}\min_X \left\{ \langle Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s, X\geq 0\right\} &= -\max_X \left\{ \langle -Q, X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s, X\geq 0\right\},\\ &= -\max_X \left\{ \langle \Pi_+(-Q), X\rangle : \|X\|_F^2 \leq t^2, \|X\|_0 \leq s\right\},\\ &= -\max_X \left\{ \langle \mathcal{H}_s(\Pi_+(-Q)), X\rangle : \|X\|_F^2 \leq t^2\right\}.\end{aligned}$$

The first equality is a simple rewriting of the objective. Then, the corresponding objective $\langle -Q, X\rangle$ can be maximized with $\sum_{i=1}^{A}\sum_{j=1}^{B} I_{(i,j)\in\Omega_0}(-Q_{ij}X_{ij})$ where Ω_0 is set of index pairs and $I_{(i,j)\in\Omega_0}$ is 1 if the index pair if $(i,j)\in\Omega_0$ and zero otherwise. It is easy to see that the objective $\langle -Q, X\rangle$ is maximized if Ω_0 contains all the index pairs corresponding to the elements of $-Q$ with highest absolute value which is captured by Hard-thresholding operator. However due to the non-negativity constraint if there is any $-Q_{ij}$ such that it is negative, then since X_{ij} will be driven to zero. So, before we use the Hard-thresholding operator, we need to use $\Pi_+(.) = \max\{0, .\}$ in second equality. The third equality follows as a consequence of hard sparsity constraint similar to Lemma C.2.5.1 and the solution follows due to Lemma D.2.0.1. The equivalence statement follows as $\|X^*\|_F^2 = t^2$. □

Proposition C.3.5.2. *In BPG-MF, with the above defined* f_1, f_0, h *the update steps in each iteration are* $U^{k+1} = r\mathcal{H}_{s_1}(\Pi_+(-P^k))$, $Z^{k+1} = r\mathcal{H}_{s_2}(\Pi_+(-Q^k))$ *where* $r \geq 0$ *and satisfies*

$$c_1\left(\left\|\mathcal{H}_{s_1}\left(\Pi_+(-Q^k)\right)\right\|_F^2 + \left\|\mathcal{H}_{s_2}\left(\Pi_+(-P^k)\right)\right\|_F^2\right)r^3 + c_2 r - 1 = 0,$$

with $c_1 = 3$ *and* $c_2 = \|A\|_F$.

The proof is similar to Proposition C.3.0.1.

C.4 Matrix completion problem

Matrix completion is an important non-convex optimization problem, which arises in practical real world applications, such as recommender systems [41, 68, 95]. Give a matrix A where only the values at the index set given by Ω are given. The goal is obtain the rest of the values. One of the popular strategy is to obtain the factors $U \in \mathbb{R}^{M\times K}$ and $Z \in \mathbb{R}^{K\times N}$ for a small positive integer K. This is cast into the following problem,

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\left\|P_\Omega\left(A - UZ\right)\right\|_F^2 + \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)\right\}, \tag{C.4.1}$$

where P_Ω is an masking operator over index set Ω which preserves the given matrix entries and sets others to zero.. We require the following technical lemma.

Lemma C.4.0.1. *Let* $f_1 := \frac{1}{2}\left\|P_\Omega\left(A - UZ\right)\right\|_F^2$ *be as defined above, we have the following*

$$\nabla_U f_1(A, UZ) = -P_\Omega(A - UZ)Z^T, \quad \nabla_Z f_1(A, UZ) = -U^T P_\Omega(A - UZ)$$

$$\left\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\right\rangle = \|P_\Omega(UH_2 + H_1 Z)\|_F^2 - 2\left\langle P_\Omega(A - UZ), H_1 H_2\right\rangle .$$

Proof. With the Forbenius dot product, we have

$$\|P_\Omega(A - UZ)\|_F^2 = \left\langle P_\Omega(A - UZ), P_\Omega(A - UZ)\right\rangle .$$

In the above expression by substituting U with $U + H_1$ and Z with $Z + H_2$, we obtain

$$\begin{aligned}&\left\langle P_\Omega(A - (U + H_1)(Z + H_2)), P_\Omega(A - (U + H_1)(Z + H_2))\right\rangle ,\\ &= \|P_\Omega(A - UZ)\|_F^2 + \|P_\Omega(UH_2 + H_1 Z)\|_F^2\\ &- 2\left\langle P_\Omega(A - UZ), P_\Omega(UH_2 + H_1 Z)\right\rangle - 2\left\langle P_\Omega(A - UZ), P_\Omega(H_1 H_2)\right\rangle\end{aligned}$$

where in the last term we ignored the terms higher than second order. Collecting all the first order terms we have

$$\begin{aligned}-2\left\langle P_\Omega(A - UZ), P_\Omega(UH_2 + H_1 Z)\right\rangle &= -2\left\langle P_\Omega(A - UZ), UH_2 + H_1 Z\right\rangle\\ &= -2\left\langle P_\Omega(A - UZ)Z^T, H_1\right\rangle - 2\left\langle U^T P_\Omega(A - UZ), H_2\right\rangle\end{aligned}$$

and similarly collecting all the second order terms we have

$$\|P_\Omega(UH_2 + H_1 Z)\|_F^2 - 2\left\langle P_\Omega(A - UZ), P_\Omega(H_1 H_2)\right\rangle = \|P_\Omega(UH_2 + H_1 Z)\|_F^2 - 2\left\langle P_\Omega(A - UZ), H_1 H_2\right\rangle$$

Thus the statement follows using the second order Taylor expansion. □

Proposition C.4.0.1. *Let* $f_1 := \frac{1}{2}\left\|P_\Omega\left(A - UZ\right)\right\|_F^2$ *and* h_1, h_2 *be as defined as in (4.5.2). Then, for a certain constant* $L \geq 1$*, the function* f_1 *satisfies* L*-smad property with respect to the following kernel generating distance,*

$$h_a(U, Z) = 3h_1(U, Z) + \|P_\Omega(A)\|_F\, h_2(U, Z) .$$

Proof. With Lemma C.4.0.1 we obtain

$$\begin{aligned}&\left\langle (H_1, H_2), \nabla^2 f_1(A, UZ)(H_1, H_2)\right\rangle\\&= \|P_\Omega(UH_2 + H_1 Z)\|_F^2 - 2\left\langle P_\Omega(A - UZ), H_1H_2\right\rangle\\&\leq \|H_1 Z + UH_2\|_F^2 - 2\left\langle P_\Omega(A - UZ), H_1H_2\right\rangle\\&\leq 2\|H_1Z\|_F^2 + 2\|UH_2\|_F^2 + 2\|P_\Omega(A)\|_F\|H_1H_2\|_F + 2\|P_\Omega(UZ)\|_F\|H_1H_2\|_F\,,\\&\leq 2\|H_1Z\|_F^2 + 2\|UH_2\|_F^2 + 2\|P_\Omega(A)\|_F\|H_1H_2\|_F + 2\|UZ\|_F\|H_1H_2\|_F\,.\end{aligned}$$

The rest of the proof is similar to Proposition 4.5.0.1. □

Proposition C.4.0.2. *Let* $f_1 := \frac{1}{2}\|P_\Omega(A - UZ)\|_F^2 + \frac{\lambda_0}{2}\left(\|U\|_F^2 + \|Z\|_F^2\right)$ *and* h_1, h_2 *be as defined as in* (4.5.2)*. Then, for a certain constant* $L \geq 1$*, the function* f_1 *satisfies* L*-smad property with respect to the following kernel generating distance,*

$$h_a(U, Z) = 3h_1(U, Z) + (\|P_\Omega(A)\|_F + \lambda_0)\, h_2(U, Z)\,.$$

The update steps are very similar as what we described earlier in Section C.2 and C.3.

C.5 Closed form solution with 5th-order polynomial

The goal of this section is to consider a setting, where the update step of BPG-MF involves a 5th order polynomial equation. In such a case, Newton based method solvers can be used to find the roots. We later show that we can obtain a cubic equation by slightly modifying the kernel generating distance. Let $\lambda_0 > 0$ and we consider the following problem

$$\min_{U \in \mathbb{R}^{M\times K}, Z \in \mathbb{R}^{K\times N}} \left\{ f(U, Z) := \frac{1}{2}\|A - UZ\|_F^2 + \frac{\lambda_0}{2}\|U\|_F^2 \right\}. \tag{C.5.1}$$

We set $\mathcal{R}_1(U) = \frac{\lambda_0}{2}\|U\|_F^2$, $\mathcal{R}_2(Z) = 0$, $f_1 = \frac{1}{2}\|A - UZ\|_F^2$, $f_0(U, Z) = \frac{\lambda_0}{2}\|U\|_F^2$ and $h = h_a$.

Proposition C.5.0.1. *In BPG-MF, with above defined* f_1, f_0, h *the update steps in each iteration are given by* $U^{k+1} = -\frac{P^k}{r_1 + \lambda_0}$, $Z^{k+1} = -\frac{Q^k}{r_1}$ *where* $r_1 \geq 0$ *and satisfies*

$$c_1\left(\left\|Q^k\right\|_F^2 (r_1 + \lambda_0)^2 + \left\|P^k\right\|_F^2 r_1^2\right) + c_2 r_1^2 (r_1 + \lambda_0)^2 - r_1^3 (r_1 + \lambda_0)^2 = 0\,, \tag{C.5.2}$$

with $c_1 = 3$ *and* $c_2 = \|A\|_F$.

Proof. The proof is similar to that of Proposition C.2.0.1. Consider the following subproblem

$$\begin{aligned}(U^{k+1}, Z^{k+1}) \in \operatorname*{argmin}_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}} &\left\{ \frac{\lambda_0}{2}\|U\|_F^2 + \left\langle P^k, U\right\rangle + \left\langle Q^k, Z\right\rangle \right.\\&\left. + c_1\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)^2 + c_2\left(\frac{\|U\|_F^2 + \|Z\|_F^2}{2}\right)\right\},\end{aligned}$$

Denote the objective in the above minimization problem as $\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})$. Now, we show that the following holds

$$\begin{aligned}\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}} \left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right) &\equiv \min_{t_1\geq 0, t_2\geq 0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F=t_1, \|Z\|_F=t_2}\left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\},\\ &\equiv \min_{t_1\geq 0, t_2\geq 0}\left\{\min_{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}, \|U\|_F\leq t_1, \|Z\|_F\leq t_2}\left(\mathcal{O}(\mathcal{U}^{\|}, \mathcal{Z}^{\|})\right)\right\}.\end{aligned}$$

where the first step is a simple rewriting of the objective and the second step follows as there is no change in the constraint set and due to Lemma D.2.0.1, which is given precisely in Proposition C.2.0.1 where the equivalence argument used for (C.2.4) and (C.2.3) holds here. Note that in the first step, we used $\|U\|_F = t_1$ this results in deviation of value of c_2 to $c_2 + \lambda_0$, corresponding to U (see below). We solve for (U^{k+1}, Z^{k+1}) via the following strategy. Denote

$$U_1^*(t_1) \in \operatorname{argmin}\left\{\left\langle P^k, U_1\right\rangle : U_1 \in \mathbb{R}^{M\times K}, \|U_1\|_F^2 \leq t_1^2\right\},$$

$$Z_1^*(t_2) \in \operatorname{argmin}\left\{\left\langle Q^k, Z_1\right\rangle : Z_1 \in \mathbb{R}^{K\times N}, \|Z_1\|_F^2 \leq t_2^2\right\}.$$

Then we obtain $(U^{k+1}, Z^{k+1}) = (U_1^*(t_1^*), Z_1^*(t_2^*))$, where t_1^* and t_2^* are obtained by solving the following two dimensional subproblem

$$\begin{aligned}(t_1^*, t_2^*) \in \operatorname*{argmin}_{t_1\geq 0, t_2\geq 0}\Bigg\{&\min_{U_1\in\mathbb{R}^{M\times K}}\left\{\left\langle P^k, U_1\right\rangle : \|U_1\|_F^2 \leq t_1^2\right\} + \min_{Z_1\in\mathbb{R}^{K\times N}}\left\{\left\langle Q^k, Z_1\right\rangle : \|Z_1\|_F^2 \leq t_2^2\right\}\\ &+c_1\left(\frac{t_1^2+t_2^2}{2}\right)^2 + c_2\frac{t_2^2}{2} + (c_2+\lambda_0)\frac{t_1^2}{2}\Bigg\}.\end{aligned}$$

Note that inner minimization subproblems can be trivially solved once we obtain $U_1^*(t_1)$ and $Z_1^*(t_2)$ via Lemma D.2.0.1. Then the solution to the subproblem in each iteration as follows:

$$U^{k+1} = \begin{cases} t_1^* \frac{-P^k}{\|P^k\|_F}, & \text{for } \left\|P^k\right\|_F \neq 0,\\ 0 & \textit{otherwise}.\end{cases}$$

$$Z^{k+1} = \begin{cases} t_2^* \frac{-Q^k}{\|Q^k\|_F}, & \text{for } \left\|Q^k\right\|_F \neq 0,\\ 0 & \textit{otherwise}.\end{cases}$$

We solve for t_1^* and t_2^* with the following two dimensional minimization problem

$$\operatorname*{argmin}_{t_1\geq 0, t_2\geq 0}\left\{-t_1\left\|P^k\right\|_F - t_2\left\|Q^k\right\|_F + c_1\left(\frac{t_1^2+t_2^2}{2}\right)^2 + c_2\frac{t_2^2}{2} + (c_2+\lambda_0)\frac{t_1^2}{2}\right\}.$$

Thus, the solutions t_1^* and t_2^* are the non-negative real roots of the following equations

$$-\left\|P^k\right\|_F + c_1(t_1^2+t_2^2)t_1 + (c_2+\lambda_0)t_1 = 0, \tag{C.5.3}$$

$$-\left\|Q^k\right\|_F + c_1(t_1^2+t_2^2)t_2 + c_2 t_2 = 0. \tag{C.5.4}$$

Further simplifications with $t_1 = \frac{\|P^k\|_F}{r_1+\lambda_0}$ and $t_2 = \frac{\|Q^k\|_F}{r_1}$ denoting $r_1 = c_1(t_1^2+t_2^2)+c_2$, then we have

$$r_1 = c_1\left(\left(\frac{\|P^k\|_F}{r_1+\lambda_0}\right)^2 + \left(\frac{\|Q^k\|_F}{r_1}\right)^2\right) + c_2$$

This will result in following 5^{th} order equation,

$$c_1\left(\left\|P^k\right\|_F^2 r_1^2 + \left\|Q^k\right\|_F^2 (r_1+\lambda_0)^2\right) + c_2 r_1^2(r_1+\lambda_0)^2 - r_1^3(r_1+\lambda_0)^2 = 0\,.$$

□

C.5.1 Conversion to cubic equation

We set $\mathcal{R}_1(U) = \frac{\lambda_0}{2}\|U\|_F^2$, $\mathcal{R}_2(Z) = 0$ and $f_1 = \frac{1}{2}\|A - UZ\|_F^2$. Denote $f_0(U,Z) = \frac{\lambda_0}{2}\|U\|_F^2$, $h(U,Z) = h_a(U,Z) + \frac{\lambda_0}{2}\|Z\|_F^2$. Note that such a f_1 satisfies L-smad property with respect to h satisfies L-smad trivially since only a quadratic term is added to h_a.

Proposition C.5.1.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = -r\,P^k$, $Z^{k+1} = -r\,Q^k$ where r is the non-negative real root of*

$$c_1\left(\left\|Q^k\right\|_F^2 + \left\|P^k\right\|_F^2\right) r^3 + (c_2+\lambda_0)r - 1 = 0\,, \tag{C.5.5}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

Proof. The resulting subproblem is

$$(U^{k+1}, Z^{k+1}) \in \underset{(U,Z)\in\mathbb{R}^{M\times K}\times\mathbb{R}^{K\times N}}{\operatorname{argmin}} \left\{\left\langle P^k, U\right\rangle + \left\langle Q^k, Z\right\rangle \right.$$
$$\left. + c_1\left(\frac{\|U\|_F^2+\|Z\|_F^2}{2}\right)^2 + (c_2+\lambda_0)\left(\frac{\|U\|_F^2+\|Z\|_F^2}{2}\right)\right\}.$$

The rest of the proof is similar to Proposition C.2.0.1. □

C.5.2 Extensions to mixed regularization terms

Let $\lambda_0 > 0$ and we consider the following problem

$$\min_{U\in\mathbb{R}^{M\times K}, Z\in\mathbb{R}^{K\times N}} \left\{ f(U,Z) := \frac{1}{2}\|A-UZ\|_F^2 + \frac{\lambda_0}{2}\|U\|_F^2 + \lambda_1\|Z\|_1 \right\}. \tag{C.5.6}$$

Note that the regularizer is a mixture of L1 and L2 regularization. The usual strategy with $h = h_a$ would result in a fifth order polynomial. In order to generate a cubic equation, we use the same strategy as given Section C.5.1. We set $h(U,Z) = h_a(U,Z) + \frac{\lambda_0}{2}\|Z\|_F^2$, $f_1 = \frac{1}{2}\|A-UZ\|_F^2$ and $f_0(U,Z) = \frac{\lambda_0}{2}\|U\|_F^2 + \lambda_1\|Z\|_1$.

Proposition C.5.2.1. *In BPG-MF, with the above defined f_1, f_0, h the update steps in each iteration are given by $U^{k+1} = -r\,P^k$, $Z^{k+1} = r\mathcal{S}_{\lambda\lambda_1}\left(-Q^k\right)$ where r is the non-negative real root of*

$$c_1\left(\left\|P^k\right\|_F^2 + \left\|\mathcal{S}_{\lambda\lambda_1}\left(-Q^k\right)\right\|_F^2\right)r^3 + (c_2 + \lambda_0)r - 1 = 0\,, \tag{C.5.7}$$

with $c_1 = 3$ and $c_2 = \|A\|_F$.

The proof is similar to Proposition C.2.0.1 and Proposition C.2.3.1.

Appendix D

Appendix for deep matrix factorization - Chapter 7

D.1 Proof of Theorem 7.3.2.1

Proof. The kernel generating distances h in Section 4.6 are continuously differentiable (also proper, lsc, convex with full domain). The function f_1 in (4.6.1) is continuously differentiable and for L1 or L2 regularization, the function f_0 is proper, lsc, convex, and bounded from below by the zero function. As both f_0 and f_1 are non-negative, the objective is bounded from below. The function f is obviously semi-algebraic, which assures the KL property. Supercoercivity of $h+\lambda f_0$ is true, as h is a polynomial of degree greater than 1 with positive coefficients in terms of $\|W\|_F$, and f_0 is either L1 or squared L2 regularizer. Strong convexity of h and the L-smad property are verified in Section 4.6. Lipschitz continuity of ∇f_1 and ∇h on bounded sets follows from boundedness of second order derivative of f_1 and h, as $\|X\|_F$ and $\|Y\|_F$ are constant (see Section A.3). Thus, Assumptions A,C,D are verified. Moreover, BPG satisfies the descent property with respect to objective value, i.e., f is monotonically non-increasing (cf. [28, Lemma 4.1]). The regularization terms guarantee that the objective f is coercive, which implies that the level sets are bounded. The descent property ensures that all BPG iterates lie in the set $\{W : f(W) \leq f(W^0)\}$. Combining this with the boundedness of the level-sets shows that the entire sequence generated by BPG is bounded. It remains to apply [28, Theorem 4.1] to conclude the statement. □

D.2 Closed form update steps

Lemma D.2.0.1. *Let $Q \in \mathbb{R}^{A\times B}$ for some positive integers A and B. Let $t \geq 0$ and $\|Q\|_F \neq 0$ then*

$$\min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 = t^2\right\} \equiv \min_{X\in\mathbb{R}^{A\times B}} \left\{\langle Q, X\rangle : \|X\|_F^2 \leq t^2\right\} = -t\,\|Q\|_F\ ,$$

with the minimizer at $X^* = -tQ/\|Q\|_F$.

Consider the following non-convex optimization problem

$$\min_{W_i\in\mathcal{W}_i\,\forall i\in\{1,\ldots,K\}} \left\{f(W_1,\ldots,W_N) := \frac{1}{2}\|W_1W_2\ldots W_NX - Y\|_F^2\right\}\ , \tag{D.2.1}$$

Recall that $f_1 = \frac{1}{2}\|W_1W_2\ldots W_NX - Y\|_F^2$, $f_0 := 0$ and h as explained in Section 7.3.1.

D.2.1 Proof of Proposition 7.3.1.1

We use the same proof strategy as Proposition C.2.0.1. Consider the following subproblem, involved in the update step

$$(W_1^{k+1}, \dots, W_N^{k+1}) \in \underset{(W_1,\dots,W_N)\in C}{\operatorname{argmin}} \left\{ \left(\sum_{i=1}^{N} \left\langle P_i^k, W_i \right\rangle \right) + c_1(N) \left(\frac{\|W\|_F^2}{N} \right)^N + c_2(N) \left(\frac{\|W\|_F^2}{N} \right)^{\frac{N}{2}} + \rho \left(\frac{\|W\|_F^2}{N} \right) \right\} .$$

In order to solve the above minimization problem, we introduce additional optimization variables $t_1, \dots, t_N \geq 0$ and the constraint $\|W_i\|_F = t_i$ for all i. This splits the optimization problem, where the constraints of the inner problem with respect to $W_1, \dots, W_N$ can be relaxed to $\|W_i\|_F \leq t_i$ without changing the minimal value thanks to Lemma D.2.0.1 . We arrive at

$$\min_{t_i \geq 0, \forall i \in \{1,\dots,N\}} \left\{ \sum_{i=1}^{N} \min_{W_i \in \mathcal{W}_i} \left\{ \left\langle P_i^k, W_i \right\rangle : \|W_i\|_F^2 \leq t_i^2 \right\} \right.$$
$$\left. + c_1(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^N + c_2(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^{\frac{N}{2}} + \rho \left(\frac{\sum_{i=1}^N t_i^2}{N} \right) \right\} .$$

Then the solution to the subproblem for the i-th block due to Lemma D.2.0.1, in each iteration is as follows

$$W_i^{k+1} = \begin{cases} t_i^* \frac{-P_i^k}{\|P_i^k\|_F}, & \text{for } \|P_i^k\|_F \neq 0 \,, \\ 0 & \textit{otherwise} \,. \end{cases}$$

We solve for t_i^* with the following minimization problem

$$\underset{t_i \geq 0, \forall i \in \{1,\dots,N\}}{\operatorname{argmin}} \left\{ -\sum_{i=1}^{N} t_i \left\| P_i^k \right\|_F + c_1(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^N + c_2(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^{\frac{N}{2}} + \rho \left(\frac{\sum_{i=1}^N t_i^2}{N} \right) \right\} .$$

Thus, the solutions t_i^* are the non-negative real roots of the following equations

$$-\left\| P_i^k \right\|_F + 2c_1(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^{N-1} t_i + c_2(N) \left(\frac{\sum_{i=1}^N t_i^2}{N} \right)^{\frac{N}{2}-1} t_i + \frac{2\rho}{N} t_i = 0 \,, \quad \forall i \in \{1,\dots,N\} \tag{D.2.2}$$

Substitute the following

$$t_i = r \frac{\sqrt{N} \, \|P_i^k\|_F}{\sqrt{\sum_{i=1}^N \|P_i^k\|_F^2}} \,,$$

which implies that $\frac{\sum_{i=1}^N t_i^2}{N} = r^2$ for certain $r > 0$. Now, we find r via substituting t_i in (D.2.2), which results in

$$2c_1(N) r^{2N-1} + c_2(N) r^{N-1} + \frac{2\rho}{N} r - \frac{\sqrt{\sum_{i=1}^N \|P_i^k\|_F^2}}{\sqrt{N}} = 0 \,. \tag{D.2.3}$$

The proof is similar for $N > 2$ and N being odd. □

D.2.2 L2-regularization

Consider the following non-convex optimization problem

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in \{1,\ldots,K\}} \left\{ f(W_1,\ldots,W_N) := \frac{1}{2}\left\|W_1W_2\ldots W_NX - Y\right\|_F^2 + \frac{\lambda_0}{2}\left(\sum_{i=1}^{N}\left\|W_i\right\|_F^2\right)\right\} . \tag{D.2.4}$$

Denote $f_1 := \frac{1}{2}\left\|W_1W_2\ldots W_NX - Y\right\|_F^2$, $f_0 := \frac{\lambda_0}{2}\left(\sum_{i=1}^{N}\left\|W_i\right\|_F^2\right)$ and h as explained in Section 7.3.1.

Proposition D.2.2.1. *In BPG, with above defined f_1, f_0, h, using the notation $P_i^k = P^k{}_i\left({W_1}^k,\ldots,{W_N}^k\right) = \lambda\nabla_{W_i}f_1\left({W_1}^k,\ldots,{W_N}^k\right) - \nabla_{W_i}h({W_1}^k,\ldots,{W_N}^k)$. the update steps in each iteration are given by $W_i^{k+1} = -r\,\frac{\sqrt{N}\,P_i^k}{\|P\|_F}$ for all $i \in \{1,\ldots,N\}$ where r is the non-negative real root of for $N = 2$*

$$2c_1(2)r^3 + (c_2(2) + \lambda\lambda_0)r - \frac{\sqrt{\sum_{i=1}^{2}\left\|P_i^k\right\|_F^2}}{\sqrt{2}} = 0\,, \tag{D.2.5}$$

If $N > 2$ and even, we have

$$2c_1(N)r^{2N-1} + c_2(N)r^{N-1} + \left(\frac{2\rho}{N} + \lambda\lambda_0\right)r - \frac{\sqrt{\sum_{i=1}^{N}\left\|P_i^k\right\|_F^2}}{\sqrt{N}} = 0\,, \tag{D.2.6}$$

and if $N > 2$ and odd, then

$$2c_1(N)r^{2N-1} + c_3(N)\left(\frac{Nr^2+1}{N+1}\right)^{\frac{N-1}{2}} r + \left(\frac{2\rho}{N} + \lambda\lambda_0\right)r - \frac{\sqrt{\sum_{i=1}^{N}\left\|P_i^k\right\|_F^2}}{\sqrt{N}} = 0\,. \tag{D.2.7}$$

Proof. The proof is exactly the same as Proposition 7.3.1.1 and the only change is in the value ρ for $N > 2$ and c_2 for $N = 2$. For $N = 2$, the results coincide results from Chapter 6. □

D.2.3 Closed form updates for L1 Regularization

Recall that the soft-thresholding operator is defined as follows $\mathcal{S}_\theta(x) = \max\{|x| - \theta, 0\}\text{sgn}(x)$, where the operations are performed coordinate-wise. We consider below an extension of (4.6.1),

$$\min_{W_i \in \mathcal{W}_i \, \forall i \in \{1,\ldots,K\}} \left\{ f(W_1,\ldots,W_N) := \frac{1}{2}\left\|W_1W_2\ldots W_NX - Y\right\|_F^2 + \sum_{i=1}^{N}\mu_i\left\|W_i\right\|_1\right\} , \tag{D.2.8}$$

where $\mu_i > 0$ for all $i \in \{1,\ldots,N\}$ and $\|W_i\|_1$ is the standard L1-norm, which denotes the sum of absolute of values of the all the elements in W_i.

Denote $f_1 := \frac{1}{2}\left\|W_1W_2\ldots W_NX - Y\right\|_F^2$, $f_0 := \sum_{i=1}^{N}\mu_i\left\|W_i\right\|_1$ and h as explained in Section 7.3.1.

Proposition D.2.3.1. *In BPG, with above defined f_1, f_0, h, with the notation $P_i^k = P_i^k\left({W_1}^k,\ldots,{W_N}^k\right) = \lambda\nabla_{W_i}f_1\left({W_1}^k,\ldots,{W_N}^k\right) - \nabla_{W_i}h({W_1}^k,\ldots,{W_N}^k)$, the update steps in each iteration are given by $W_i^{k+1} =$*

$r\frac{\sqrt{N}\mathcal{S}_{\lambda\mu_i}(-P_i^k)}{\sqrt{\sum_{i=1}^N\|\mathcal{S}_{\lambda\mu_i}(-P_i^k)\|_F^2}}$ *for all* $i \in \{1,\ldots,N\}$ *where for* $N = 2$*,* r *is the non-negative real root of*

$$2c_1(2)r^3 + c_2(2)r - \frac{\sqrt{\sum_{i=1}^2 \left\|\mathcal{S}_{\lambda\mu_i}(-P_i^k)\right\|_F^2}}{\sqrt{2}} = 0\,. \tag{D.2.9}$$

If $N > 2$ *and even, we have*

$$2c_1(N)r^{2N-1} + c_2(N)r^{N-1} + \frac{2\rho}{N}r - \frac{\sqrt{\sum_{i=1}^N \left\|\mathcal{S}_{\lambda\mu_i}(-P_i^k)\right\|_F^2}}{\sqrt{N}} = 0\,, \tag{D.2.10}$$

and if $N > 2$ *and odd, then*

$$2c_1(N)r^{2N-1} + c_3(N)\left(\frac{Nr^2+1}{N+1}\right)^{\frac{N-1}{2}} r + \frac{2\rho}{N}r - \frac{\sqrt{\sum_{i=1}^N \left\|\mathcal{S}_{\lambda\mu_i}(-P_i^k)\right\|_F^2}}{\sqrt{N}} = 0\,. \tag{D.2.11}$$

Proof. We use the same proof strategy as Proposition C.2.0.1.The subproblem is

$$\begin{aligned} W^{k+1} \in \underset{(W_1,\ldots,W_N)\in C}{\operatorname{argmin}} & \left\{ \sum_{i=1}^N \left(\lambda\mu_i \left\|W_i\right\|_1 + \left\langle P_i^k, W_i\right\rangle\right)\right. \\ & \left. + c_1(N)\left(\frac{\|W\|_F^2}{N}\right)^N + c_2(N)\left(\frac{\|W\|_F^2}{N}\right)^{\frac{N}{2}} + \rho\left(\frac{\|W\|_F^2}{N}\right)\right\}\,. \end{aligned}$$

The rest of the proof is only a minor modification to the proof of Proposition 7.3.1.1 and Lemma C.2.3.1 is used instead of Lemma D.2.0.1.

□

D.3 Closed form inertia

D.3.1 Proof of Proposition 7.4.1.1

We use

$$h(W_1,\ldots,W_N) = H_a(W_1,\ldots,W_N) + \rho H_4(W_1,\ldots,W_N)\,,$$

where

$$H_a(W_1,\ldots,W_N) = c_1(N)H_1(W_1,\ldots,W_N) + c_2(N)H_2(W_1,\ldots,W_N)\,.$$

Now for any $x \in \overline{C}, y \in C$, we have $D_{h_1+h_2}(x,y) = D_{h_1}(x,y) + D_{h_2}(x,y)$ for any $h_1, h_2 \in \mathcal{G}(C)$. Thus,

$$D_h(x,y) = c_1(N)D_{H_1}(x,y) + c_2(N)D_{H_2}(x,y) + \rho D_{H_4}(x,y)\,.$$

We solve $D_h\left(x^k,y^k\right) \le \kappa D_h\left(x^{k-1},x^k\right)$ using the results from Lemma A.3.0.3 ,A.3.0.5, to obtain

$$D_h\left(x^k,y^k\right) \le \gamma_k^2\left(c_1(N)\mathcal{B}_k + c_2(N)\mathcal{C}_k + \rho\left\|\Delta_k\right\|^2\right) \le \kappa D_h\left(x^{k-1},x^k\right)\,.$$

The proof for $N > 2$ and N being odd is similar.

□

D.3.2 Closed form inertia for matrix factorization

Lemma D.3.2.1. *Given* $h_1(W_1, W_2) := \left(\frac{\|W_1\|_F^2+\|W_2\|_F^2}{2}\right)^2$*, then we have the following*

$$\left\langle (H_1, H_2), \nabla^2 h_1(W_1, W_2)(H_1, H_2)\right\rangle \leq 3\left(\|H_1\|_F^2 + \|H_2\|_F^2\right)\left(\|W_1\|_F^2 + \|W_2\|_F^2\right) .$$

Given $h_2 := \left(\frac{\|W_1\|_F^2+\|W_2\|_F^2}{2}\right)$*, then we have the following*

$$\left\langle (H_1, H_2), \nabla^2 h_2(W_1, W_2)(H_1, H_2)\right\rangle = \|H_1\|_F^2 + \|H_2\|_F^2 .$$

Then, with $h_a(W_1, W_2) = 3h_1(W_1, W_2) + \|Y\|_F\, h_2(W_1, W_2)$ *we have the following*

$$\begin{aligned}&\left\langle (H_1, H_2), \nabla^2 h_a(W_1, W_2)(H_1, H_2)\right\rangle \\ &\leq 9\left(\|H_1\|_F^2 + \|H_2\|_F^2\right)\left(\|W_1\|_F^2 + \|W_2\|_F^2\right) + \|Y\|_F\left(\|H_1\|_F^2 + \|H_2\|_F^2\right) .\end{aligned}$$

Proof. The result regarding h_1 is from Lemma A.3.0.2 with $N = 2$. The results for h_2 follows trivially (see for example Section 4.5). The statement for h_a holds trivially. □

In the context of matrix factorization problem, where $N = 2$, $X = 1$, $\|X\|_F = 1$, we obtain the following result on the extrapolation parameter.

Lemma D.3.2.2. *Denote* $x^k = (W_1^k, \ldots, W_N^k)$*. For* $\kappa > 0$, $y^k := x^k + \gamma_k(x^k - x^{k-1})$ *and* $x^k \neq x^{k-1}$*, the parameter* $\gamma_k \in [0, 1]$ *such that*

$$0 \leq \gamma_k \leq \sqrt{\frac{\kappa}{(\xi_1^k + \xi_2^k)} D_h(x^{k-1}, x^k)}\,,$$

satisfies the condition (7.4.1)*, where* $\xi_1^k = 42\left\|x^k - x^{k-1}\right\|^4$ *and* $\xi_2^k = 15\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{30}\right)\left\|x^k - x^{k-1}\right\|^2$.

Proof. From Lemma A.3.0.1 we obtain

$$\begin{aligned}&\int_0^1 (1-t)\int_0^1 \left\langle \nabla^2 h\left(x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right)(x^k - y^k), x^k - y^k\right\rangle dt_1 dt \\ &\leq \int_0^1 (1-t)\int_0^1 9\left\|x^k - y^k\right\|^2\left\|x^k + (t_1 + (1-t_1)t)(y^k - x^k)\right\|^2 + \|Y\|_F\left\|x^k - y^k\right\|^2 dt_1 dt \\ &\leq \int_0^1\int_0^1 18\,(1-t)\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{18}\right)\left\|x^k - y^k\right\|^2 dt_1 dt \\ &+ \int_0^1\int_0^1 18\,(1-t)\,(t_1 + (1-t_1)t)^2\left\|x^k - y^k\right\|^4 dt_1 dt \\ &= 9\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{18}\right)\left\|x^k - y^k\right\|^2 + \int_0^1 18\,(1-t)\,(2t^2 + \frac{1}{3})\left\|x^k - y^k\right\|^4 dt \\ &= 9\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{18}\right)\left\|x^k - y^k\right\|^2 + 6\left\|x^k - y^k\right\|^4 \\ &= 9\gamma_k^2\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{18}\right)\left\|x^k - x^{k-1}\right\|^2 + 6\gamma_k^4\left\|x^k - x^{k-1}\right\|^4 ,\end{aligned}$$

where in the first inequality we used Lemma D.3.2.1 and the second inequality is due to the following

$$\left\|x^k + (t_1 + (1 - t_1)t)(y^k - x^k)\right\|^2 \leq 2\left\|x^k\right\|^2 + 2(t_1 + (1 - t_1)t)^2\left\|x^k - y^k\right\|^2 .$$

Denote $\xi_2^k = 9\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F}{18}\right)\left\|x^k - x^{k-1}\right\|^2$ and $\xi_1^k = 6\left\|x^k - x^{k-1}\right\|^4$ we have

$$\xi_1^k\gamma_k^4 + \xi_2^k\gamma_k^2 \leq \kappa D_h(x^{k-1}, x^k) ,$$

and the result follows due to the condition $0 \leq \gamma \leq 1$. □

Note that for a general X, we need to set $\xi_2^k := 15\left(\left\|x^k\right\|^2 + \frac{\|Y\|_F\|X\|_F}{30}\right)\left\|x^k - x^{k-1}\right\|^2$.

Appendix E

Appendix for deep neural networks - Chapter 8

E.1 Proof of Proposition 8.3.0.1

Proof. We use the same proof strategy as Proposition C.2.0.1. Consider the following subproblem, involved in the update step

$$(W_1^{k+1},\dots,W_N^{k+1}) \in \underset{(W_1,\dots,W_N)\in C}{\operatorname{argmin}} \left\{ \left(\sum_{i=1}^{N} \left\langle P_i^k, W_i \right\rangle \right) + \sum_{u=1}^{2N} \mathcal{C}_u \left(\frac{\sum_{p=1}^{N} \|W_p\|^2}{N} \right)^u \right\} .$$

In order to solve the above minimization problem, we introduce additional optimization variables $t_1,\dots,t_N \geq 0$ and the constraint $\|W_i\|_F = t_i$ for all i. This splits the optimization problem, where the constraints of the inner problem with respect to $W_1,\dots,W_N$ can be relaxed to $\|W_i\|_F \leq t_i$ without changing the minimal value thanks to Lemma D.2.0.1 . We arrive at

$$\min_{t_i\geq 0,\forall i\in\{1,\dots,N\}} \left\{ \sum_{i=1}^{N} \min_{W_i\in\mathcal{W}_i} \left\{ \left\langle P_i^k, W_i \right\rangle : \|W_i\|_F^2 \leq t_i^2 \right\} + \sum_{u=1}^{2N} \mathcal{C}_u \left(\frac{\sum_{p=1}^{N} t_p^2}{N} \right)^u \right\} .$$

Then the solution to the subproblem for the i-th block due to Lemma D.2.0.1, in each iteration is as follows

$$W_i^{k+1} = \begin{cases} t_i^* \frac{-P_i^k}{\|P_i^k\|_F}, & \text{for } \left\|P_i^k\right\|_F \neq 0\,, \\ 0 & \textit{otherwise}\,. \end{cases}$$

We solve for t_i^* with the following minimization problem

$$\underset{t_i\geq 0,\forall i\in\{1,\dots,N\}}{\operatorname{argmin}} \left\{ -\sum_{i=1}^{N} t_i \left\|P_i^k\right\|_F + \sum_{u=1}^{2N} \mathcal{C}_u \left(\frac{\sum_{p=1}^{N} t_p^2}{N} \right)^u \right\} .$$

Thus, the solutions t_i^* are the non-negative real roots of the following equations

$$-\left\|P_i^k\right\|_F + \sum_{u=1}^{2N} 2\mathcal{C}_u \left(\frac{u}{N}\right) \left(\frac{\sum_{p=1}^{N} t_p^2}{N} \right)^{u-1} t_i = 0\,, \quad \forall i \in \{1,\dots,N\}\,. \tag{E.1.1}$$

Substitute the following

$$t_i = r\frac{\sqrt{N}\,\left\|P_i^k\right\|_F}{\sqrt{\sum_{i=1}^N \left\|P_i^k\right\|_F^2}},$$

which implies that $\frac{\sum_{i=1}^N t_i^2}{N} = r^2$ for certain $r > 0$. Now, we find r via substituting t_i in (E.1.1), which results in

$$\sum_{u=1}^{2N} 2\mathcal{C}_u\left(\frac{u}{N}\right) r^{2u-1} - \frac{\sqrt{\sum_{i=1}^N \left\|P_i^k\right\|_F^2}}{\sqrt{N}} = 0. \tag{E.1.2}$$

□

Appendix F

Appendix for Model BPG - Chapter 9

F.1 Proof of Example 9.3.0.1

The model error is given by

$$\begin{aligned}
|f(x) - f(x;\bar{x})| &\leq |g(x) - g(\bar{x}) - \langle \nabla g(\bar{x}), x - \bar{x}\rangle| \ , \\
&\leq |\langle \nabla g(\bar{x} + s(x-\bar{x})) - \nabla g(\bar{x}), x - \bar{x}\rangle| \ , \\
&\leq \|\nabla g(\bar{x} + s(x-\bar{x})) - \nabla g(\bar{x})\| \|x - \bar{x}\| \ .
\end{aligned}$$

where in the second inequality we use mean value theorem with $s \in [0,1]$, the third inequality is a simple application of Cauchy-Schwarz rule. On further application of the fundamental theorem of calculus, we have

$$\begin{aligned}
\|\nabla g(\bar{x} + s(x-\bar{x})) - \nabla g(\bar{x})\| &= \| \int_0^1 \nabla^2 g(\bar{x} + s(x-\bar{x}))(x-\bar{x}) ds\| \ , \\
&\leq \int_0^1 \|\nabla^2 g(\bar{x} + s(x-\bar{x}))\| \|x-\bar{x}\| ds \, .
\end{aligned}$$

Using the fact that $\nabla^2 g(x) = 4\|x\|^2 I + 8xx^T$, and $\|\nabla^2 g(x)\| \leq 12\|x\|^2$ we obtain

$$\begin{aligned}
\|\nabla g(\bar{x} + s(x-\bar{x})) - \nabla g(\bar{x})\| &\leq 12 \int_0^1 \|\bar{x} + s(x-\bar{x})\|^2 \|x-\bar{x}\| ds \, , \\
&\leq 12 \int_0^1 \left(2\|\bar{x}\|^2 + 2s^2\|(x-\bar{x})\|^2\right) \|x-\bar{x}\| ds \, , \\
&\leq 24\|\bar{x}\|^2\|x-\bar{x}\| + 8\|x-\bar{x}\|^3 \, ,
\end{aligned}$$

where in the second step we used the inequality $\|A+b\|^2 \leq 2\|A\|^2 + 2\|b\|^2$ for any $A, b \in \mathbb{R}^N$. For any model center $\bar{x} \in \mathbb{R}^N$, the growth function is then given by $\varsigma_{\bar{x}}(t) = 24\|\bar{x}\|^2 t^2 + 8t^4$.

F.2 Model function preserves first order information

Lemma F.2.0.1. *Let Assumption F, G hold true. For any $x \in \operatorname{dom} f$, the following condition holds true:*

$$\partial_y f(y;x)|_{y=x} = \widehat{\partial} f(x) \, .$$

Proof. We follow the proof strategy of [140, Lemma 14]. Let $\tilde{x} \in \operatorname{dom} f$ and let $v \in \widehat{\partial} f(\tilde{x})$, then, by definition we have

$$f(x) \geq f(\tilde{x}) + \langle v, x - \tilde{x} \rangle + o(\|x - \tilde{x}\|) \quad \forall\, x \in \operatorname{dom} f.$$

Using the Definition 9.3.0.2, with $f(\tilde{x}; \tilde{x}) = f(\tilde{x})$ we have the following

$$f(x; \tilde{x}) + \varsigma_{\tilde{x}}(\|x - \tilde{x}\|) \geq f(\tilde{x}; \tilde{x}) + \langle v, x - \tilde{x} \rangle + o(\|x - \tilde{x}\|)\,.$$

For any $t > 0$, note that $\varsigma_{\tilde{x}}(t) = o(t)$ as $\varsigma_{\tilde{x}}$ is a growth function, using which we obtain

$$f(x; \tilde{x}) \geq f(\tilde{x}; \tilde{x}) + \langle v, x - \tilde{x} \rangle + o(\|x - \tilde{x}\|)\,.$$

This implies that $v \in \widehat{\partial} f(\tilde{x}; \tilde{x})$ and by regularity of $f(\,\cdot\,; \tilde{x})$ we also obtain that $v \in \partial f(\tilde{x}; \tilde{x})$. For the second part of the proof, let $v \in \partial f(\tilde{x}; \tilde{x})$ with $\tilde{x} \in \operatorname{dom} f$, thus satisfying:

$$f(\bar{x}; \tilde{x}) \geq f(\tilde{x}; \tilde{x}) + \langle v, \bar{x} - \tilde{x} \rangle + o(\|\bar{x} - \tilde{x}\|)\,, \quad \forall\, \bar{x} \in \operatorname{dom} f\,.$$

Using the definition of model function (Definition 9.3.0.2), we obtain

$$f(\bar{x}) + \varsigma_{\tilde{x}}(\|\bar{x} - \tilde{x}\|) \geq f(\tilde{x}; \tilde{x}) + \langle v, \bar{x} - \tilde{x} \rangle + o(\|\bar{x} - \tilde{x}\|)\,, \quad \forall\, \bar{x} \in \operatorname{dom} f\,,$$

which on using the fact that $\varsigma_{\tilde{x}}(t) = o(t)$ results in

$$f(\bar{x}) \geq f(\tilde{x}) + \langle v, \bar{x} - \tilde{x} \rangle + o(\|\bar{x} - \tilde{x}\|)\,, \quad \forall\, \bar{x} \in \operatorname{dom} f\,.$$

□

F.3 Proof of Proposition 9.3.0.1

By global optimality of x_{k+1} as in (9.3.4), we have

$$f(x_{k+1}; x_k) + \frac{1}{\tau_k} D_h(x_{k+1}, x_k) \leq f(x_k; x_k) = f(x_k)\,. \tag{F.3.1}$$

We have the following inequality from MAP property

$$f(x_{k+1}) \leq f(x_{k+1}; x_k) + \bar{L} D_h(x_{k+1}, x_k)\,. \tag{F.3.2}$$

Thus, the result follows by combining (F.3.1) and (F.3.2). □

Appendix G

Appendix for Inertial Model BPG - Chapter 10

G.1 Proof of Lemma 10.4.1.1

Proof. Fix $k \geq 1$. With $\bar{x} \in \operatorname{dom} f \cap \operatorname{int} \operatorname{dom} h$, from the convexity of $f\left(\,\cdot\,;\bar{x}\right) - (\alpha/2)\left\|\cdot\right\|_2^2$, we obtain from the subgradient inequality [150, Example 8.8 and Proposition 8.12] that

$$f\left(x_k;y_k\right) - \frac{\alpha(y_k)}{2}\|x_k\|_2^2 \geq f\left(x_{k+1};y_k\right) - \frac{\alpha(y_k)}{2}\|x_{k+1}\|_2^2 + \left\langle \xi^{k+1} - \alpha(y_k)x_{k+1}, x_k - x_{k+1} \right\rangle,$$

where $\xi^{k+1} \in \partial f\left(x_{k+1};x_k\right)$. With $f(x_k;x_k) = f(x_k)$, by rearranging the inequality we obtain

$$f\left(x_k;y_k\right) \geq f\left(x_{k+1};y_k\right) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 + \left\langle \xi^{k+1}, x_k - x_{k+1} \right\rangle. \tag{G.1.1}$$

Now using the following inequality from MAP property

$$f(x_k;y_k) - f(x_k) \leq \underline{L}_k D_h(x_k,y_k)\,,$$

and thus we have

$$f(x_k) + \underline{L}_k D_h(x_k,y_k) \geq f\left(x_{k+1};y_k\right) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 + \left\langle \xi^{k+1}, x_k - x_{k+1} \right\rangle.$$

Now employing the following inequality from MAP property

$$f(x_k) \leq f(x_k;y_{k-1}) + \bar{L}_{k-1} D_h(x_k,y_{k-1})\,,$$

we have

$$\begin{aligned} f(x_k;y_{k-1}) + \bar{L}_{k-1} D_h(x_k,y_{k-1}) \geq{}& f\left(x_{k+1};y_k\right) + \bar{L}_k D_h(x_{k+1},y_k) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 \\ &+ \left\langle \xi^{k+1}, x_k - x_{k+1} \right\rangle - \underline{L}_k D_h(x_k,y_k) - \bar{L}_k D_h(x_{k+1},y_k)\,. \end{aligned}$$

From the optimality condition of step (10.3.4), we have that

$$\xi^{k+1} + \frac{1}{\tau_k}\left(\nabla h\left(x_{k+1}\right) - \nabla h\left(y_k\right)\right) = 0\,,$$

which yields that

$$\begin{aligned}
f(x_k; y_{k-1}) + \bar{L}_{k-1} D_h(x_k, y_{k-1}) \geq{} & f\left(x_{k+1}; y_k\right) + \bar{L}_k D_h(x_{k+1}, y_k) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 \\
& + \frac{1}{\tau_k}\left\langle \nabla h\left(y_k\right) - \nabla h\left(x_{k+1}\right), x_k - x_{k+1}\right\rangle \\
& - \underline{L}_k D_h(x_k, y_k) - \bar{L}_k D_h(x_{k+1}, y_k)\,, \\
\geq{} & f\left(x_{k+1}; y_k\right) + \bar{L}_k D_h(x_{k+1}, y_k) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 \\
& + \frac{1}{\tau_k}\left(D_h\left(x_k, x_{k+1}\right) + D_h\left(x_{k+1}, y_k\right) - D_h\left(x_k, y_k\right)\right) \\
& - \underline{L}_k D_h(x_k, y_k) - \bar{L}_k D_h(x_{k+1}, y_k)\,,
\end{aligned}$$

where the last equality follows from the three point identity of Bregman distances. Using the fact that $\tau_k^{-1} \geq \bar{L}_k$, implies that

$$\begin{aligned}
f(x_k; y_{k-1}) + \bar{L}_{k-1} D_h(x_k, y_{k-1}) \geq{} & f\left(x_{k+1}; y_k\right) + \bar{L}_k D_h(x_{k+1}, y_k) + \frac{\alpha(y_k)}{2}\|x_{k+1} - x_k\|_2^2 \\
& + \frac{1}{\tau_k} D_h\left(x_k, x_{k+1}\right) - \left(\frac{1}{\tau_k} + \underline{L}_k\right) D_h\left(x_k, y_k\right),
\end{aligned}$$

which completes the proof. □

G.2 Proof of Proposition 10.4.1.1

Proof. Multiplying (10.4.1) with τ_k and by the definition of the Lyapunov function $G_{\bar{L}}^h$ and the fact that $\tau_k \leq \tau_{k-1}$ we have

$$\begin{aligned}
& G_{\bar{L}}^h\left(x_k, x_{k-1}, x_{k-1}, \gamma_{k-1}, \tau_{k-1}, \bar{L}_{k-1}\right) \\
& \geq G_{\bar{L}}^h\left(x_{k+1}, x_k, x_{k-1}, \gamma_k, \tau_k, \bar{L}_k\right) + \frac{\alpha(y_k)\tau_k}{2}\|x_{k+1} - x_k\|_2^2 + (1-\delta)\, D_h\left(x_k, x_{k+1}\right) \\
& + \delta D_h\left(x_{k-1}, x_k\right) - \left(1 + \underline{L}_k \tau_k\right) D_h\left(x_k, y_k\right).
\end{aligned}$$

With $1 - \delta > 0$ and the σ-strong convexity of h we obtain

$$\frac{\alpha(y_k)\tau_k}{2}\|x_{k+1} - x_k\|_2^2 + (1-\delta)\, D_h\left(x_k, x_{k+1}\right) \geq \left(\frac{\alpha(y_k)\tau_k}{2} + (1-\delta)\frac{\sigma}{2}\right)\|x_{k+1} - x_k\|_2^2 \geq 0,$$

where the last inequality holds, since $\tau_k^{-1} \geq \bar{L}_k$ and $\bar{L}_k \geq \frac{-\alpha(y_k)}{(1-\delta)\sigma}$. Next, we observe that

$$D_h\left(x_k, y_k\right) \leq \frac{\delta - \varepsilon}{\left(1 + \underline{L}_k \tau_{k-1}\right)} D_h\left(x_{k-1}, x_k\right) \leq \frac{\delta - \varepsilon}{\left(1 + \underline{L}_k \tau_k\right)} D_h\left(x_{k-1}, x_k\right),$$

where the first inequality is due to the step (10.3.2) of the algorithm and the second inequality is due to fact that $\tau_k \leq \tau_{k-1}$. By rearranging we obtain,

$$\delta D_h\left(x_{k-1}, x_k\right)-\left(1+\underline{L}_k \tau_k\right) D_h\left(x_k, y_k\right) \geq \varepsilon D_h\left(x_{k-1}, x_k\right)$$

thus completing the proof. □

G.3 Proof of Lemma 10.4.2.1

Proof. Combining the sum rule for the limiting subdifferential in [150, Prop. 10.5], we obtain

$$\begin{aligned}
&\partial G^h_{\bar{L}}(x_{k+1}, x_k, x_{k-1}, \gamma_k)\\
&= \Big(\partial_{x_{k+1}} f(x_{k+1}; y_k) + \bar{L}\big(\nabla h(x_{k+1}) - \nabla h(y_k)\big) + \delta_1\left(\nabla h(x_k) - \nabla h(x_{k+1})\right), \qquad \text{(G.3.1)}\\
&(1+\gamma_k)\partial_{y_k} f(x_{k+1}; y_k) - (1+\gamma_k)\bar{L}\nabla^2 h(y_k)(x_{k+1} - y_k) + \delta_1(\nabla h(x_k) - \nabla h(x_{k+1}))\,,\\
&(-\gamma_k)\partial_{y_k} f(x_{k+1}; y_k) - (-\gamma_k)\bar{L}\nabla^2 h(y_k)(x_{k+1} - y_k)\,,\\
&(x_k - x_{k-1})^T\partial_{y_k} f(x_{k+1}; y_k) - \bar{L}(x_k - x_{k-1})^T\nabla^2 h(y_k)(x_{k+1} - y_k)\Big)\,.
\end{aligned}$$

Using Fermat's rule, optimality of x_{k+1} in (10.3.4) and [150, Prop. 10.5] imply the existence of $\xi^{k+1}_{x_{k+1}} \in \partial_{x_{k+1}} f(x_{k+1}; x_k)$ such that (10.3.4) holds. The first block coordinate in (G.3.1) satisfies

$$\begin{aligned}
&\xi^{k+1}_{x_{k+1}} + \bar{L}\big(\nabla h(x_{k+1}) - \nabla h(y_k)\big) + \delta_1\left(\nabla h(x_k) - \nabla h(x_{k+1})\right)\\
&= \left(\bar{L} - \frac{1}{\tau_k}\right)\big(\nabla h(x_{k+1}) - \nabla h(x_k)\big) + \delta_1\left(\nabla h(x_k) - \nabla h(x_{k+1})\right)\,.
\end{aligned}$$

In the subsequent calculation, we use the fact that the bounded second order derivatives of bounded subsets of $\operatorname{int}\operatorname{dom} h$ and also for some $C_1 > 0$ the following condition holds true

$$\inf_{v \in \partial_{y_k} f(x_{k+1}; y_k)} \|v\|_2 \leq C_1 \|x_{k+1} - y_k\|_2\,.$$

For any $w_1 \in \partial_{y_k} f(x_{k+1}; y_k)$, we have

$$\begin{aligned}
\left|(x_k - x_{k-1})^T w_1\right| &\leq \frac{\|w_1\|_2^2 + \|(x_k - x_{k-1})\|_2^2}{2}\,,\\
&\leq \frac{c^2\|(x_{k+1} - y_k)\|_2^2 + \|(x_k - x_{k-1})\|_2^2}{2}\,,\\
&\leq c^2\|(x_{k+1} - x_k)\|_2^2 + \left(\frac{2c^2\gamma_k^2 + 1}{2}\right)\|(x_k - x_{k-1})\|_2^2\,.
\end{aligned}$$

Thus, there exists $B_1, B_2 > 0$ such that for any $w_1 \in \partial_{y_k} f(x_{k+1}; y_k)$ we have

$$\left|(x_k - x_{k-1})^T w_1\right| \leq B_1\|(x_{k+1} - x_k)\|_2^2 + B_2\|(x_k - x_{k-1})\|_2^2\,.$$

There exists $\zeta_{k+1} \in \partial G^h_{\bar{L}}(x_{k+1}, x_k, x_{k-1})$ such that

$$\|\zeta_{k+1}\|_2 \leq D_1\|x_{k+1} - x_k\|_2 + D_2\|x_k - x_{k-1}\|_2 + B_1\|(x_{k+1} - x_k)\|_2^2 + B_2\|(x_k - x_{k-1})\|_2^2\,,$$

holds true for some $D_1, D_2, B_1, B_2 > 0$ where in the last step, we used the boundedness of $\nabla^2 h$ by L_h, and Assumption I. Using a similar strategy as in (9.5.3.1), the stated result follows. □

G.4 Proof of Proposition 10.4.3.2

Proof. (*i*) We show the inclusion $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) \subset \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ and $\omega_f^{\operatorname{int}\operatorname{dom} h}(x_0) \subset \omega^{\operatorname{int}\operatorname{dom} h}(x_0)$ is clear by definition. Let $x^\star \in \omega^{\operatorname{int}\operatorname{dom} h}(x_0)$, then we obtain the following

$$
\begin{aligned}
f(x^\star) + \left(\underline{L} + \frac{1}{\tau_k}\right) D_h(x^\star, y_k) \overset{(9.3.3)}{\geq} f(x^\star; y_k) + \frac{1}{\tau_k} D_h(x^\star, y_k) &\overset{(10.3.4)}{\geq} f(x_{k+1}; y_k) + \frac{1}{\tau_k} D_h(x_{k+1}, y_k) \\
&\overset{(9.3.3)}{\geq} f(x_{k+1}) - \left(\bar{L} - \frac{1}{\tau_k}\right) D_h(x_{k+1}, y_k) \overset{\varepsilon_k > 0}{\geq} f(x_{k+1}) \,.
\end{aligned}
$$

Obviously, by Assumption I(iii) combined with the fact that $y_k \underset{K}{\to} x^\star$, we have $D_h(x^\star, y_k) \to 0$ as $k \underset{K}{\to} \infty$, which, together with the lower semicontinuity of f, implies

$$
f(x^\star) \geq \liminf_{k \underset{K}{\to} \infty} f(x_{k+1}) \geq f(x^\star) \,,
$$

thus $x^\star \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$.

(*ii*) If $x \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$ and $x_k \underset{K}{\to} x$ for $K \subset \mathbb{N}$, then we have that $D_h(x_{k+1}, x_k) \to 0$ as $k \underset{K}{\to} \infty$ and $f(x_k) \underset{K}{\to} f(x)$. As $D_h(x_{k+1}, x_k) \to 0$ as $k \to \infty$, we have $x_{k-1} \underset{K}{\to} x$. The first part of the proof implies $f(x_{k-1}) \underset{K}{\to} f(x)$. We also have $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} f(x)$ which we prove below, which implies that $(x, x, x, \gamma) \in \omega_{G^h_{\bar{L}}}^{(\operatorname{int}\operatorname{dom} h)^3 \times [0,1]}(x_0)$. We now describe why $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} f(x)$. Note that by definition of the $G^h_{\bar{L}}$ we have the following

$$
\begin{aligned}
G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) &= f(x_k; y_{k-1}) + \bar{L} D_h(x_k, y_{k-1}) + \delta_1 D_h\left(x_{k-1}, x_k\right) \,, \\
&= f(x_k) + \left(f(x_k; y_{k-1}) - f(x_k)\right) + \bar{L} D_h(x_k, y_{k-1}) + \delta_1 D_h\left(x_{k-1}, x_k\right) \,,
\end{aligned}
$$

and with MAP property we have

$$
f(x_k) + \delta_1 D_h\left(x_{k-1}, x_k\right) \leq G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \leq f(x_k) + (\bar{L} + \underline{L}) D_h(x_k, y_{k-1}) + \delta_1 D_h\left(x_{k-1}, x_k\right) \,. \tag{G.4.1}
$$

Thus, we have that $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} f(x)$ as $D_h(x_k, x_{k-1}) \underset{K}{\to} 0$ and $D_h(x_k, y_{k-1}) \underset{K}{\to} 0$.

Conversely, suppose $(x, x, x, \gamma) \in \omega_{G^h_{\bar{L}}}^{(\operatorname{int}\operatorname{dom} h)^3 \times [0,1]}(x_0)$ and $x_k \underset{K}{\to} x$ for $K \subset \mathbb{N}$. This, together with $D_h(x_k, x_{k-1}) \to 0$ as $k \underset{K}{\to} \infty$, induces $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} f(x)$, hence $f(x_k) \underset{K}{\to} f(x)$ due to the following. Note that we have

$$
\begin{aligned}
f(x_k) &= G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) + \left(f(x_k) - f(x_k; y_{k-1})\right) - \bar{L} D_h(x_k, y_{k-1}) - \delta_1 D_h\left(x_{k-1}, x_k\right) \,, \\
&\geq G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) - (\bar{L} + \underline{L}) D_h(x_k, y_{k-1}) - \delta_1 D_h\left(x_{k-1}, x_k\right) \,.
\end{aligned}
$$

Finally we have

$$\begin{aligned}&G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) - (\bar{L} + \underline{L})D_h(x_k, y_{k-1}) - \delta_1 D_h\left(x_{k-1}, x_k\right)\\ &\leq f(x_k) \leq G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) - \delta_1 D_h\left(x_{k-1}, x_k\right) .\end{aligned}$$

Thus, with $D_h(x_k, y_{k-1}) \underset{k\in K}{\to} 0$ and $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} f(x)$, we deduce that $f(x_k) \underset{K}{\to} f(x)$. And therefore $x \in \omega_f^{\operatorname{int}\operatorname{dom} h}(x_0)$.
(iii) By Proposition 10.4.1.3, the sequence $(G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}))_{k\in\mathbb{N}}$ converges to some $-\infty < \underline{G} < \infty$. Note that $D_h(x_k, x_{k-1}) \underset{K}{\to} 0$ by simple application of (10.4.4). For $(x^\star, x^\star, x^\star, \gamma) \in \omega_{G^h_{\bar{L}}}^{(\operatorname{int}\operatorname{dom} h)^3\times[0,1]}(x_0)$ there exists $K \subset \mathbb{N}$ such that $x_k \underset{K}{\to} x^\star$, $x_{k-1} \underset{K}{\to} x^\star$, $x_{k-2} \underset{K}{\to} x^\star$ and $G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) \underset{K}{\to} G^h_{\bar{L}}(x^\star, x^\star, x^\star, \gamma) = f(x^\star)$, i.e., the value of the limit point is independent of the choice of the subsequence. The result follows directly and by using (i).

□

G.5 Proof of Theorem 10.4.4.1

Proof. Note that the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model CoCaIn BPG is a bounded sequence (using a similar argument as in Remark 9.3.0.2). The proof relies on Theorem 9.4.0.1 provided in Section 9.4 in the appendix, for which we need to verify the conditions (i)–(v). Similar to Lemma 9.5.5.1, $G^h_{\bar{L}}$ satisfies Kurdyka–Łojasiewicz property at each point of $\operatorname{dom}\partial G^h_{\bar{L}}$.

Note that as $\omega^{\operatorname{int}\operatorname{dom} h}(x_0) = \omega(x_0)$ holds true, there exists a sufficiently small $\varepsilon > 0$ such that $\tilde{B} := \{x : \operatorname{dist}(x, \omega(x_0)) \leq \varepsilon\} \subset \operatorname{int}\operatorname{dom} h$. As $\omega(x_0)$ is compact due to Proposition 9.5.4.1(i), the set $\tilde{B}$ is also compact. Moreover, the convex hull of the set $\tilde{B}$ denoted by $B := \operatorname{conv}\tilde{B}$ is also compact, as the convex hull of a compact set is also compact in finite dimensional setting. A simple calculation reveals that the set B lies in the set $\operatorname{int}\operatorname{dom} h$. Thus, due to Proposition 10.4.3.1 along with Proposition 9.5.4.1(ii), without loss of generality, we assume that the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Model BPG lies in the set B. By definition of σ_B as per Assumption K we have

$$D_h(x_{k+1}, x_k) \geq \frac{\sigma_B}{2}\|x_{k+1} - x_k\|^2 , \tag{G.5.1}$$

through which we obtain

$$G^h_{\bar{L}}(x_{k+1}, x_k, x_{k-1}, \gamma_k) \leq G^h_{\bar{L}}(x_k, x_{k-1}, x_{k-2}, \gamma_{k-1}) - \frac{\varepsilon\sigma_B}{2}\|x_{k-1} - x_k\|_2^2 ,$$

which is (i) with $d_n = \frac{\varepsilon\sigma_B}{2}\|x_{k-1} - x_k\|_2^2$ and $a_n = 1$. Using (10.4.6) from the proof of Theorem 9.5.4.1, we deduce existence of $w_{k+1} \in \partial G^h_{\bar{L}}(x_{k+1}, x_k, x_{k-1}, \gamma_k)$ such that we have

$$\|w_{k+1}\|_2 \leq D_1\|x_{k+1} - x_k\|_2 + D_2\|x_k - x_{k-1}\|_2 + B_1\|(x_{k+1} - x_k)\|_2^2 + B_2\|(x_k - x_{k-1})\|_2^2$$

for some $D_1, D_2, B_1, B_2 > 0$ which is (ii) with $b = \frac{1}{D_1+D_2}$, $\theta_1 = \frac{D_1}{D_1+D_2}$ and $\theta_2 = \frac{D_2}{D_1+D_2}$, since the coefficients for both Euclidean distances are bounded from above. Denote $\varepsilon_{k+1} := \frac{B_1}{D_1+D_2}\|(x_{k+1} - x_k)\|_2^2 + \frac{B_2}{D_1+D_2}\|(x_k -$

$x_{k-1})\|_2^2$. Note that from Proposition 10.4.1.3(ii) we have

$$\frac{\sigma_B}{2}\sum_{k=1}^{\infty}\|x_{k+1}-x_k\|_2^2 \leq \sum_{k=1}^{\infty} D_h(x_{k+1},x_k) < \infty\,, \tag{G.5.2}$$

which implies ε_{k+1} is ℓ_1-summable.

The continuity condition (iii) is deduced from a converging subsequence, whose existence is guaranteed by boundedness of $(x_k)_{k\in\mathbb{N}}$, and Proposition 10.4.3.2. The distance condition (iv) holds trivially as $\varepsilon > 0$ and $\mu > 0$. The parameter condition (v), holds because $b_n = 1$ in this setting, hence $(b_n)_{n\in\mathbb{N}} \notin \ell_1$ and also, we have

$$\sup_{n\in\mathbb{N}} \frac{1}{b_n a_n} = 1 < \infty\,, \quad \inf_n a_n =: 1 > 0\,.$$

The last statement of the theorem follows using the same technique as Theorem 9.5.6.3. □

Bibliography

[1] P. Ablin, D. Fagot, H. Wendt, A. Gramfort, and C. Févotte. A quasi-Newton algorithm on the orthogonal manifold for NMF with transform learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 700–704, 2019. 82

[2] M. Ahookhosh, A. Themelis, and P. Patrinos. A Bregman forward-backward linesearch algorithm for nonconvex composite optimization: superlinear convergence to nonisolated local minima. *SIAM Journal on Optimization*, 31(1):653–685, 2021. 123

[3] S. Arora, N. Cohen, W. Hu, and Y. Luo. Implicit regularization in deep matrix factorization. In *Advances in Neural Information Processing Systems*, pages 7413–7424, 2019. 92, 93

[4] S. Arora, R. Ge, R. Kannan, and A. Moitra. Computing a nonnegative matrix factorization–provably. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 145–162. ACM, 2012. 83

[5] H. Asi and J. C. Duchi. The importance of better models in stochastic optimization. *Proceedings of the National Academy of Sciences*, 116(46):22924–22930, 2019. 140

[6] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming*, 116(1-2):5–16, 2009. 69, 133, 134, 147

[7] 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*, 137(1-2):91–129, 2013. 17, 30, 33, 147

[8] A. Auslender and M. Teboulle. Interior gradient and proximal methods for convex and conic optimization. *SIAM Journal on Optimization*, 16(3):697–725, 2006. 39

[9] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012. 3

[10] H. H. Bauschke, J. Bolte, and M. Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2017. 9, 38, 40, 42, 65, 115, 117, 118, 143, 144, 145, 146

[11] H. H. Bauschke and J. M. Borwein. Legendre functions and the method of random Bregman projections. *Journal of Convex Analysis*, 4(1):27–67, 1997. 39, 40

[12] H.H. Bauschke, J.M. Borwein, and P.L. Combettes. Bregman monotone optimization algorithms. *SIAM Journal on control and optimization*, 42(2):596–636, 2003. 40

[13] A. Beck. *First-order methods in optimization.* SIAM, 2017. 2, 3, 9, 17, 18, 19, 20

[14] A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003. 5, 38, 41, 116

[15] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. 60, 150

[16] M. Benning, M. M. Betcke, M. J. Ehrhardt, and C. B. Schönlieb. Choose your path wisely: gradient descent in a Bregman distance framework. *arXiv preprint arXiv:1712.04045*, 2017. 38

[17] R. V. D. Berg, T. N. Kipf, and M. Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, 2017. 92

[18] M. Bertero, P. Boccacci, G. Desiderà, and G. Vicidomini. Image deblurring with Poisson data: from cells to galaxies. *Inverse Problems*, 25(12):123006, 2009. 8, 143

[19] D. P. Bertsekas. *Convex optimization theory.* Athena Scientific Belmont, 2009. 11

[20] B. Birnbaum, N. R. Devanur, and L. Xiao. Distributed algorithms via gradient descent for Fisher markets. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 127–136. ACM, 2011. 38

[21] J. Bochnak, M. Coste, and M-F. Roy. *Real algebraic geometry.* Springer, 1998. 31, 32

[22] J. Bolte, A. Daniilidis, and A. Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17:1205–1223, 2006. 30, 32, 69

[23] J. Bolte, A. Daniilidis, A.S. Lewis, and M. Shiota. Clarke subgradients of stratifiable functions. *SIAM Journal on Optimization*, 18(2):556–572, 2007. 30, 31, 32, 33

[24] J. Bolte, A. Daniilidis, O. Ley, and L. Mazet. Characterizations of Łojasiewicz inequalities: subgradient flows, talweg, convexity. *Transactions of the American Mathematical Society*, 362(6):3319–3363, 2010. 32

[25] J. Bolte and E. Pauwels. Majorization-minimization procedures and convergence of SQP methods for semi-algebraic and tame programs. *Mathematics of Operations Research*, 41(2):442–465, 2016. 123

[26] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014. 7, 8, 30, 31, 32, 33, 69, 82, 83, 84, 86, 96, 105, 128, 133, 134, 147, 201, 208

[27] J. Bolte, S. Sabach, and M. Teboulle. Nonconvex Lagrangian-based optimization: monitoring schemes and global convergence. *Mathematics of Operations Research*, 43(4):1210–1232, 2018. 116

[28] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd. First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151, 2018. xxv, 4, 5, 6, 9, 31, 33, 37, 38, 39, 41, 42, 43, 46, 63, 64, 65, 69, 74, 75, 76, 82, 89, 94, 95, 104, 114, 115, 116, 117, 118, 119, 121, 122, 123, 126, 131, 135, 136, 140, 141, 147, 166, 175, 215

[29] R. I. BoŢ, E. R. Csetnek, and S. C. László. An inertial forward–backward algorithm for the minimization of the sum of two nonconvex functions. *EURO Journal on Computational Optimization*, 4(1):3–25, 2016. 63

[30] L. Bottou. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nımes*, 91(8):12, 1991. 105

[31] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 105

[32] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018. 105, 160

[33] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 12, 13, 15

[34] L. M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3):200–217, 1967. 36, 38

[35] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences*, 101(12):4164–4169, 2004. xxiv, 7, 43, 88, 89

[36] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005. 7

[37] S. Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015. 38

[38] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011. 83, 84, 196, 206

[39] J. F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010. 83, 84, 199, 200

[40] E. J. Candes, X. Li, and M. Soltanolkotabi. Phase retrieval via Wirtinger flow: theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015. 6, 74, 140

[41] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009. 83, 209

[42] E. J. Candes, M. B. Wakin, and S. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008. 73

[43] C. Cartis, N. I. M. Gould, and P. L. Toint. On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming. *SIAM Journal on Optimization*, 21(4):1721–1739, 2011. 9

[44] Y. Censor and T. Elfving. A multiprojection algorithm using Bregman projections in a product space. *Numerical Algorithms*, 8(2):221–239, 1994. 38

[45] Y. Censor and A. Lent. An iterative row-action method for interval convex programming. *Journal of Optimization Theory and Applications*, 34(3):321–353, 1981. 38

[46] Y. Censor and S. A. Zenios. Proximal minimization algorithm with D-functions. *Journal of Optimization Theory and Applications*, 73(3):451–464, 1992. 38, 40

[47] H. Chang, S. Marchesini, Y. Lou, and T. Zeng. Variational phase retrieval with globally convergent preconditioned proximal algorithm. *SIAM Journal on Imaging Sciences*, 11(1):56–93, 2018. 6, 74, 75

[48] S. Chaudhuri, R. Velmurugan, and R. M. Rameshan. *Blind image deconvolution*. Springer, 2016. 7, 43

[49] Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pages 192–204, 2015. 45, 92

[50] E. Chouzenoux, J. C. Pesquet, and A. Repetti. A block coordinate variable metric forward–backward algorithm. *Journal of Global Optimization*, 66(3):457–485, 2016. 82, 105

[51] A. Cichocki, R. Zdunek, and S. Amari. Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization. In *International Conference on Independent Component Analysis and Signal Separation*, pages 169–176. Springer, 2007. 82, 105

[52] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H.H. Bauschke, R.S. Burachik, P.L. Combettes, V. Elser, D.R. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer, 2011. 116

[53] V. Corona, M. Benning, M. J. Ehrhardt, L. F. Gladden, R. Mair, A. Reci, A. J. Sederman, S. Reichelt, and C. B. Schönlieb. Enhancing joint reconstruction and segmentation with non-convex Bregman iteration. *Inverse Problems*, 35(5):055001, 2019. 38

[54] M. Coste. An introduction to semialgebraic geometry, 2000. 32

[55] D. Davis, D. Drusvyatskiy, and K. J. MacPhee. Stochastic model-based minimization under high-order growth. *ArXiv preprint arXiv:1807.00255*, 2018. 9, 38, 85, 89, 93, 96, 112, 115, 116, 118, 119, 161

[56] D. Davis, D. Drusvyatskiy, and C. Paquette. The nonsmooth landscape of phase retrieval. *IMA Journal of Numerical Analysis*, 40(4):2652–2695, 2020. 10

[57] L. Van den Dries and C. Miller. Geometric categories and o-minimal structures. *Duke Math. J*, 84(2):497–540, 1996. 31

[58] R. A. Dragomir, A. d'Aspremont, and J. Bolte. Quartic first-order methods for low-rank minimization. *Journal of Optimization Theory and Applications*, pages 1–23, 2021. 38, 83, 85, 86, 92, 96

[59] D. Drusvyatskiy. The proximal point method revisited. *arXiv preprint arXiv:1712.06038*, 2017. 137

[60] D. Drusvyatskiy, A. D Ioffe, and A. S. Lewis. Nonsmooth optimization using Taylor-like models: error bounds, convergence, and termination criteria. *Mathematical Programming*, pages 1–27, 2019. 116, 117, 118, 130, 147

[61] D. Drusvyatskiy and A. S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *Mathematics of Operations Research*, 2018. 137, 138

[62] D. Drusvyatskiy and C. Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, 178(1-2):503–558, 2019. 9, 116, 137

[63] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011. 96, 105, 160

[64] 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*, 8(3):471–529, 2019. 6, 10, 74

[65] J. Eckstein. Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18(1):202–226, 1993. 38, 40

[66] G. Z. Eskandani, M. Raeisi, and T. M. Rassias. A hybrid extragradient method for solving pseudomonotone equilibrium problems using Bregman distance. *Journal of Fixed Point Theory and Applications*, 20(3):132, 2018. 38

[67] F. Esposito, N. Gillis, and N. D. Buono. Orthogonal joint sparse NMF for microarray data analysis. *Journal of Mathematical Biology*, pages 1–25, 2019. 83

[68] H. Fang, Z. Zhang, Y. Shao, and C. J. Hsieh. Improved bounded matrix completion for large-scale recommender systems. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1654–1660. AAAI Press, 2017. 83, 209

[69] P. Frankel, G. Garrigos, and J. Peypouquet. Splitting methods with variable metric for Kurdyka–Łojasiewicz functions and general convergence rates. 165(3):874–900, September 2014. 133

[70] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*. Springer series in statistics New York, 2001. 73, 142

[71] J. Geiping and M. Moeller. Composite optimization by nonconvex majorization-minimization. *SIAM Journal on Imaging Sciences*, 11(4):2494–2528, 2018. 38

[72] G. Gidel, F. Bach, and S. Lacoste-Julien. Implicit regularization of discrete gradient dynamics in deep linear neural networks. *arXiv preprint arXiv:1904.13262*, 2019. 93

[73] N. Gillis. The why and how of nonnegative matrix factorization. *Regularization, Optimization, Kernels, and Support Vector Machines*, 12(257), 2014. 82, 83

[74] N. Gillis and S. A. Vavasis. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(4):698–714, 2014. 83

[75] G. H. Golub and C. F.V. Loan. *Matrix computations*, volume 3. John Hopkins University Press, 2012. 82, 105

[76] P. Gong, C. Zhang, L. Zhaosong, J. Z. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28, pages 37–45. PMLR, 2013. 74

[77] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. 7, 8, 45, 47, 52, 92, 103, 105

[78] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtarik. SGD: General analysis and improved rates. *ArXiv preprint arXiv:1901.09401*, 2019. 89

[79] S. Gunasekar, J. Lee, D. Soudry, and N. Srebro. Characterizing implicit bias in terms of optimization geometry. In *International Conference on Machine Learning*, pages 1832–1841. PMLR, 2018. 93

[80] S. Gunasekar, B. E. Woodworth, S. Bhojanapalli, B. Neyshabur, and N. Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems*, pages 6151–6159, 2017. 93

[81] W. M. Haddad and V. Chellaboina. *Nonlinear dynamical systems and control: a Lyapunov-based approach*. Princeton university press, 2011. 123

[82] B. D. Haeffele and R. Vidal. Structured low-rank matrix factorization: global optimality, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 7, 43

[83] F. M. Harper and J. A. Konstan. The movielens datasets: history and context. *Transactions on Interactive Intelligent Systems (TIIS)*, 5(4):19, 2016. xxiv, 86, 88, 90, 97

[84] L. T. K. Hien and N. Gillis. Algorithms for nonnegative matrix factorization with the Kullback-Leibler divergence. *arXiv preprint arXiv:2010.01935*, 2020. 38

[85] L. T. K. Hien, N. Gillis, and P. Patrinos. Inertial block mirror descent method for non-convex non-smooth optimization. *ArXiv preprint arXiv:1903.01818*, 2019. 96

[86] J.-B. Hiriart-Urruty and C. Lemarechal. *Fundamentals of Convex Analysis*. Springer Science & Business Media, 2012. 77

[87] C. J. Hsieh and I. S. Dhillon. Fast coordinate descent methods with variable selection for non-negative matrix factorization. In *International Conference on Knowledge Discovery and Data Mining (ICKDDM)*, pages 1064–1072. ACM, 2011. 83

[88] C. J. Hsieh and P. Olsen. Nuclear norm minimization via active subspace selection. In *International Conference on Machine Learning*, pages 575–583, 2014. 83, 84

[89] P. Jawanpuria and B. Mishra. A unified framework for structured low-rank matrix learning. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 2254–2263. PMLR, 2018. 83, 85, 90

[90] D. Harrison Jr and D. L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81–102, 1978. 108

[91] A. Juditsky, A. Nemirovski, et al. First order methods for nonsmooth convex large-scale optimization, ii: utilizing problems structure. *Optimization for Machine Learning*, pages 149–183, 2011. 38

[92] K. Kawaguchi. Deep learning without poor local minima. In *Advances in neural information processing systems*, pages 586–594, 2016. 45, 92

[93] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 96, 105, 160

[94] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009. 85, 89, 92

[95] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009. 83, 84, 86, 97, 209

[96] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 8, 47, 52, 103, 105

[97] K. Kurdyka. On gradients of functions definable in o-minimal structures. *Université de Grenoble. Annales de l'Institut Fourier*, 48(3):769–783, 1998. 30, 69

[98] E. Laude, P. Ochs, and D. Cremers. Bregman proximal mappings and Bregman-Moreau envelopes under relative prox-regularity. *Journal of Optimization Theory and Applications*, 184(3):724–761, 2020. 38, 121, 123

[99] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, U. A. Muller, E. Sackinger, et al. Comparison of learning algorithms for handwritten digit recognition. In *International conference on artificial neural networks*, volume 60, pages 53–60. Perth, Australia, 1995. 52

[100] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788, 1999. 202

[101] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001. 83, 202

[102] A. S. Lewis and S. J. Wright. A proximal method for composite minimization. *Mathematical Programming*, 158(1-2):501–546, 2016. 137, 138

[103] Q. Li, Z. Zhu, G. Tang, and M. B. Wakin. Provable Bregman-divergence based methods for nonconvex and non-lipschitz problems. *arXiv preprint arXiv:1904.09712*, 2019. 38, 43, 44, 96

[104] W. Li and D.-Y. Yeung. Relation regularized matrix factorization. In *International Joint Conference on Artifical Intelligence (IJCAI)*, pages 1126–1131, 2009. 196

[105] X. Li and X. Wu. Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4520–4524. IEEE, 2015. 8, 47, 103, 105

[106] S. Łojasiewicz. Une propriété topologique des sous-ensembles analytiques réels. In *Les Équations aux Dérivées Partielles (Paris, 1962)*, pages 87–89. Éditions du Centre National de la Recherche Scientifique, Paris, 1963. 69

[107] C. Lu, M. Yang, F. Luo, F. X. Wu, M. Li, Y. Pan, Y. Li, and J. Wang. Prediction of lncRNA–disease associations based on inductive matrix completion. *Bioinformatics*, 34(19):3357–3364, 2018. 83

[108] H. Lu. "Relative-Continuity" for non-Lipschitz non-smooth convex optimization using stochastic (or deterministic) mirror descent. *INFORMS Journal on Optimization*, 1(4):288–303, 2019. 38

[109] H. Lu, R. M. Freund, and Y. Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018. 38, 39, 118, 119

[110] D. R. Luke. Phase retrieval, What's new? *SIAG/OPT Views and News*, 25(1):1–6, 2017. 6, 74, 140

[111] R. Luss and M. Teboulle. Conditional gradient algorithms for rank-one matrix approximations with a sparsity constraint. *SIAM Review*, 55(1):65–98, 2013. 194, 198, 201, 202

[112] A. Mnih and R. R. Salakhutdinov. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, pages 1257–1264, 2008. 7, 43

[113] A. Moitra. An almost optimal algorithm for computing nonnegative rank. *SIAM Journal on Computing*, 45(1):156–173, 2016. 83

[114] C. Molinari, J. Liang, and J. Fadili. Convergence rates of Forward–Douglas–Rachford splitting method. *Journal of Optimization Theory and Applications*, 182(2):606–639, 2019. 123

[115] F. Monti, M. M. Bronstein, and X. Bresson. Geometric matrix completion with recurrent multi-graph neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3700–3710, 2017. 92

[116] B. S. Mordukhovich. *Variational analysis and applications*. Springer, 2018. 21, 126

[117] J. J. Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société mathématique de France*, 93:273–299, 1965. 63

[118] M. C. Mukkamala, J. Fadili, and P. Ochs. Global convergence of model function based Bregman proximal minimization algorithms. *arXiv preprint arXiv:2012.13161*, 2020. 10

[119] M. C. Mukkamala and M. Hein. Variants of rmsprop and adagrad with logarithmic regret bounds. In *International Conference on Machine Learning (ICML)*, pages 2545–2553, 2017. 89, 96, 105, 160

[120] M. C. Mukkamala and P. Ochs. Beyond alternating updates for matrix factorization with inertial Bregman proximal gradient algorithms. In *Advances in Neural Information Processing Systems*, pages 4266–4276, 2019. 10

[121] M. C. Mukkamala, P. Ochs, T. Pock, and S. Sabach. Convex-Concave backtracking for inertial Bregman proximal gradient algorithms in nonconvex optimization. *SIAM Journal on Mathematics of Data Science*, 2(3):658–682, 2020. 10

[122] M. C. Mukkamala, F. Westerkamp, E. Laude, D. Cremers, and P. Ochs. Bregman proximal framework for deep linear neural networks. *arXiv preprint arXiv:1910.03638*, 2019. 10

[123] M. C. Mukkamala, F. Westerkamp, E. Laude, D. Cremers, and P. Ochs. Bregman proximal gradient algorithms for deep matrix factorization. In *Scale Space and Variational Methods in Computer Vision: 8th International Conference, SSVM 2021, Virtual Event, May 16–20, 2021, Proceedings*, page 204. Springer, 2021. 10

[124] Y. Nesterov. Introductory lectures on convex optimization: a basic course, 2004. 3, 20, 114

[125] Y. Nesterov. Modified Gauss–Newton scheme with worst case guarantees for global performance. *Optimisation methods and software*, 22(3):469–483, 2007. 137

[126] Y. E. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Doklady Akademii Nauk SSSR*, 269(3):543–547, 1983. 19, 60, 150

[127] B. Neyshabur, R. Tomioka, R. Salakhutdinov, and N. Srebro. Geometry of optimization and implicit regularization in deep learning. *arXiv preprint arXiv:1705.03071*, 2017. 93

[128] L. M. Nguyen, P. H. Nguyen, M. V. Dijk, P. Richtárik, K. Scheinberg, and M. Takáč. SGD and Hogwild! convergence without the bounded gradients assumption. *arXiv preprint arXiv:1802.03801*, 2018. 89

[129] Q. Nguyen and M. Hein. The loss surface of deep and wide neural networks. volume 70 of *Proceedings of Machine Learning Research*, pages 2603–2612, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. 48

[130] Q. V. Nguyen. Forward–Backward splitting with Bregman distances. *Vietnam Journal of Mathematics*, 45(3):519–539, 2017. 38

[131] M. Nikolova. Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares. *Multiscale Modeling & Simulation*, 4(3):960–991, 2005. 73, 77, 143

[132] D. Noll. Convergence of non-smooth descent methods using the Kurdyka–Łojasiewicz inequality. *Journal of Optimization Theory and Applications*, 160(2):553–572, September 2013. 117

[133] D. Noll, O. Prot, and P. Apkarian. A proximity control algorithm to minimize nonsmooth and nonconvex functions. *Pacific Journal of Optimization*, 4(3):571–604, 2008. 117

[134] P. Ochs. *Long term motion analysis for object level grouping and nonsmooth optimization methods.* PhD thesis, Albert-Ludwigs-Universität Freiburg, Mar 2015. 22, 31, 32, 33, 63, 65

[135] P. Ochs. Local convergence of the heavy-ball method and ipiano for non-convex optimization. *Journal of Optimization Theory and Applications*, 177(1):153–180, 2018. 65

[136] P. Ochs. Unifying abstract inexact convergence theorems and block coordinate variable metric ipiano. *SIAM Journal on Optimization*, 29(1):541–570, 2019. 63, 69, 121, 122, 150

[137] P. Ochs, Y. Chen, T. Brox, and T. Pock. iPiano: inertial proximal algorithm for nonconvex optimization. *SIAM Journal on Imaging Sciences*, 7(2):1388–1419, 2014. 2, 8, 62, 67, 72, 96, 116, 123, 135

[138] P. Ochs, A. Dosovitskiy, T. Pock, and T. Brox. An iterated ℓ_1 algorithm for non-smooth non-convex optimization in computer vision. In *CVPR*, 2013. xxv, 116, 139, 141, 143, 146, 147, 158

[139] P. Ochs, J. Fadili, and T. Brox. Non-smooth non-convex Bregman minimization: unification and new algorithms. *Journal of Optimization Theory and Applications*, 181(1):244–278, 2019. xxiv, 9, 38, 40, 76, 78, 79, 116, 117, 118, 121, 126, 128, 130, 143, 147, 154

[140] P. Ochs and Y. Malitsky. Model function based conditional gradient method with Armijo-like line search. In *International Conference on Machine Learning*, pages 4891–4900, 2019. 117, 118, 224

[141] J. S. Pang and M. Tao. Decomposition methods for computing directional stationary solutions of a class of nonsmooth nonconvex optimization problems. *SIAM Journal on Optimization*, 28(2):1640–1669, 2018. 38

[142] E. Pauwels. The value function approach to convergence analysis in composite optimization. *Operations Research Letters*, 44(6):790–795, 2016. 116, 123

[143] T. Pock and A. Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *International Conference on Computer Vision*, pages 1762–1769, 2011. 141, 143

[144] T. Pock and S. Sabach. Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems. *SIAM Journal on Imaging Sciences*, 9(4):1756–1787, 2016. 7, 8, 31, 33, 63, 82, 86, 96, 105

[145] B. T. Polyak. Some methods of speeding up the convergence of iterative methods. *Akademija Nauk SSSR. Žurnal Vyčislitel′noĭ Matematiki i Matematičeskoĭ Fiziki*, 4:791–803, 1964. 62, 72, 150, 158

[146] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. P. Reyes, M. L. Shyu, S. C. Chen, and S. S. Iyengar. A survey on deep learning: algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)*, 51(5):1–36, 2018. 47

[147] M. Powell. On search directions for minimization algorithms. *Mathematical programming*, 4(1):193–201, 1973. 85

[148] R. T. Rockafellar. *Convex analysis*, volume 36. Princeton university press, 1970. 13, 39

[149] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1970. 39

[150] R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis*, volume 317 of *Fundamental Principles of Mathematical Sciences*. Springer-Verlag, Berlin, 1998. 11, 21, 22, 23, 25, 26, 27, 28, 29, 30, 68, 117, 120, 121, 126, 136, 138, 186, 200, 225, 227

[151] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016. 105

[152] S. Sabach. *Iterative methods for solving optimization problems*. Technion-Israel Institute of Technology, Faculty of Mathematics, 2012. 27

[153] D. Scieur, A. d'Aspremont, and F. Bach. Regularized nonlinear acceleration. *Mathematical Programming*, 179(1):47–83, 2020. 150

[154] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019. 8, 103, 105

[155] S. Sra and I. S. Dhillon. Generalized nonnegative matrix approximations with Bregman divergences. In *Advances in Neural Information Processing Systems*, pages 283–290, 2006. 7, 43

[156] N. Srebro, J. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems*, pages 1329–1336, 2005. 7, 43

[157] J.-L. Starck, F. Murtagh, and J. Fadili. *Sparse image and signal processing: wavelets, curvelets, morphological diversity*. Cambridge University Press, 2010. 7, 43

[158] L. Stella, A. Themelis, and P. Patrinos. Forward–backward quasi-Newton methods for nonsmooth optimization problems. *Computational Optimization and Applications*, 67(3):443–487, 2017. 123

[159] W. Su, S. Boyd, and E. Candes. A differential equation for modeling Nesterov's accelerated gradient method: theory and insights. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2510–2518. Curran Associates, Inc., 2014. 67

[160] M. Teboulle. Entropic proximal mappings with application to nonlinear programming. *Mathematics of Operations Research*, 17(3):670–690, 1992. 38, 40, 63

[161] M. Teboulle. A simplified view of first order methods for optimization. *Mathematical Programming*, 170(1):67–96, 2018. 38, 40, 63, 65

[162] M. Teboulle and Y. Vaisbourd. Novel proximal gradient methods for nonnegative matrix factorization with sparsity constraints. *SIAM Journal on Imaging Sciences*, 13(1):381–421, 2020. 38, 96

[163] K. Thung, P. T. Yap, E. Adeli, S. W. Lee, D. Shen, and Alzheimer's Disease Neuroimaging Initiative. Conversion and time-to-conversion predictions of mild cognitive impairment using low-rank affinity pursuit denoising and matrix completion. *Medical image analysis*, 45:68–82, 2018. 83

[164] G. Wang, G. B. Giannakis, and Y. C. Eldar. Solving systems of random quadratic equations via truncated amplitude flow. *IEEE Transactions on Information Theory*, 64(2):773–794, 2018. 6, 74, 140

[165] X. Wang, X. He, M. Wang, F. Feng, and T. S. Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019. 92

[166] B. Wen, X. Chen, and T. K. Pong. Linear convergence of proximal gradient algorithm with extrapolation for a class of nonconvex nonsmooth minimization problems. *SIAM Journal on Optimization*, 27(1):124–145, 2017. 61, 63, 67

[167] F. Wen, L. Chu, P. Liu, and R. C. Qiu. A survey on nonconvex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning. *IEEE Access*, 6:69883–69906, 2018. 77

[168] T. Wu, S. Liu, J. Zhang, and Y. Xiang. Twitter spam detection based on deep learning. In *Proceedings of the Australasian Computer Science Week Multiconference*, pages 1–8. ACM, 2017. 52

[169] Y. Wu, B. Poczos, and A. Singh. Towards understanding the generalization bias of two layer convolutional linear classifiers with gradient descent. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1070–1078. PMLR, 2019. 45, 92

[170] Y. Xu, Z. Li, J. Yang, and D. Zhang. A survey of dictionary learning algorithms for face recognition. *IEEE access*, 5:8502–8514, 2017. 7, 43

[171] Y. Xu and W. Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013. 82, 105

[172] Lei Yang, Ting Kei Pong, and Xiaojun Chen. A nonmonotone alternating updating method for a class of matrix factorization problems. *SIAM Journal on Optimization*, 28(4):3402–3430, 2018. 82

[173] Q. Yao and J. Kwok. Scalable robust matrix factorization with nonconvex loss. In *Advances in Neural Information Processing Systems*, pages 5061–5070, 2018. 83, 89

[174] A. W. Yu, W. Ma, Y. Yu, J. Carbonell, and S. Sra. Efficient structured matrix rank minimization. In *Advances in Neural Information Processing Systems*, pages 1350–1358, 2014. 83

[175] C. Yun, S. Sra, and A. Jadbabaie. Global optimality conditions for deep neural networks. In *International Conference on Learning Representations*, 2018. 7, 45, 92

[176] R. Zanella, P. Boccacci, L. Zanni, and M. Bertero. Efficient gradient projection methods for edge-preserving removal of Poisson noise. *Inverse Problems*, 25(4), 2009. 8

[177] Z. Zhu, X. Li, K. Liu, and Q. Li. Dropping symmetry for fast symmetric nonnegative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 5154–5164, 2018. 83, 84, 207