Genuense Athenaeum

Cristian Jesús Vega Cereño

# Implicit Regularization: Insights from Iterative Regularization and Overparameterization

PhD thesis



DIMA - Dipartimento di eccellenza 2023-27

May 18, 2024

University of Genova

PhD Program in Mathematics and Applications
Curriculum Mathematical Methods for Data Analysis

# Implicit Regularization: Insights from Iterative Regularization and Overparameterization

by

Cristian Jesús Vega Cereño

Thesis submitted for the degree of Doctor of Philosophy (Cycle XXXVI)

May 18, 2024

Silvia Villa — Supervisor
Lorenzo Rosasco — Supervisor
Stefano Vigni — Head of the PhD program

MaLGa Center, DIMA - Dipartimento di eccellenza 2023-27, Università di Genova, Via Dodecaneso 35, Genoa, Italy

May 18, 2024

Cristian Jesús Vega Cereño

For my grandmother Nelly Concha.

# Abstract

The goal of many data-driven problems is to achieve a good prediction by estimating a quantity of interest based on a finite set of (possibly noisy) measurements, exploiting training data, and some property of the model that may be known or not a-priori. The most common methods to reach this objective are explicit and implicit regularization. The first technique consists of minimizing the sum of a loss function plus a regularizer, which is explicitly added to the objective function and entails some a priori knowledge or some desired property of the solutions that we want to select. The second technique, implicit regularization, which is the main focus of this thesis, consists of minimizing a regularizer subject to the constraints established by the loss function minimizers. In this thesis, we propose two different approaches to solving the implicit problem.

In the first part of the thesis, we follow a more traditional approach. We propose and study two new iterative regularization methods for inverse problems that are based on a primal-dual algorithm, where the bias and the loss is fixed. Our analysis, in the noise-free case, provides convergence rates for the Lagrangian and the feasibility gap. In the noisy case, it provides stability bounds and early stopping rules with theoretical guarantees. The main novelty of our work is the exploitation of some a priori knowledge about the solution set: we show that the linear equations determined by the data can be used more than once along the iterations. We discuss various approaches to reusing linear equations that are at the same time consistent with our assumptions and flexible in their implementation. Finally, we illustrate our theoretical findings with numerical simulations for robust sparse recovery and image reconstruction. We confirm the efficiency of the proposed regularization approaches by comparing the results with state-of-the-art methods.

In the second part of this thesis, inspired by the recent success of re- and over- parameterization trained with gradient descent in machine learning, we flip our perspective by fixing the loss and the algorithm (gradient flow) and reparameterize the linear model. Then, we aim to find the implicit bias introduced by the chosen optimization method and reparameterization. But there is still an open question of how to find systematically what the inductive bias hidden behind the model for a particular optimization scheme is. The goal of this thesis is to take a step in this direction by studying a unified framework encompassing various reparametrizations presented in the state of the art, called time-warped mirror flow. However, a theoretical analysis of the existence and convergence of the trajectory is missing in the state of the art. Here, we fill this gap by providing a comprehensive study. First, we prove the existence and uniqueness of the solution. Next, we establish the convergence of both the trajectory and the corresponding values of the loss function. Finally, for any convex loss function, we prove that the trajectory converges towards a minimizer of the loss function, and we provide an implicit bias. For a specific case of loss functions, including least squares, the implicit bias can be made explicit. Furthermore, we explore the flexibility of our formulation by applying the previous results to different examples related to weight normalization. Finally, we give a criterion to determine, for a

given function that depends only on the norm, a suitable weight normalization parameterization.

**Keywords.** Primal-dual splitting algorithms, Iterative regularization, Early stopping, Landweber method, Stability and convergence analysis, Overparameterization, Implicit Regularization, Time-warping mirror flow, Fully connected normalized linear networks, Weight normalization.

**AMS Mathematics Subject Classification (2020):** 34A55, 90C25, 65K10, 49M29.

# Acknowledgements

First of all, my heartfelt appreciation goes to my advisor, Silvia Villa, for her continuous support, patience, and exceptional guidance as both my thesis director and a human being. Similarly, I express my gratitude to Lorenzo Rosasco, the co-director of this thesis, to whom I'm grateful for his valuable comments on this work and his extensive knowledge. Their invaluable guidance and support were pivotal throughout the development of this work. Their insightful direction and suggestions helped to shape the findings in this thesis. Their guidance in my writing, communication, and other essential skills has been important to my growth as a mathematician.

I extend special thanks to Cesare Molinari, whose interaction was a profoundly enriching experience. Continuing collaboration with him would be both an honor and a pleasure. His comments made significant enhancements to some of the results I had initially obtained. Similarly, I extend my gratitude to Vasilis Apidopoulos, who became my mentor during this doctoral journey. I also express my gratitude to the two reviewers, Juan Peypouquet and Nicolas Flammarion, for having gladly consented to read this thesis.

The development of this doctorate would not have been possible without the financial support of Marie Skłodowska-Curie Grant No. 861137 of the European Union's Horizon 2020 research and innovation program. I also want to thank the mathematics department at the University of Genova, especially MaLGa. Likewise, I thank my former university (Técnica Federico Santa María), especially Luis Briceño and Julio Deride, who sparked my interest in optimization.

The time I spent in Genoa (and Paris) was made all the more enjoyable thanks to Luis, Angelos, Eliana, Elena, and Fernando. I also must express my profound appreciation for the enormous professional support and wonderful company during these three years from my colleagues Cheik Traore, Jonathan Chirinos, Mouna Gharbi, and Marco Rando.

I especially thank my fiance, Valeska Campos, for being the biggest source of unconditional support during these years of study, research, and writing of this thesis. I extend special thanks to my friends Omar and Jeremıas for making this doctorate time a jolly one. I also must thank my family for their constant help and support during these three years. I would like to emphasize my grandmother, Nelly Concha, who was, is, and until the end of her life will be one of the most important people and loved ones of all time, but regrettably, she could not accompany me in this achievement. You were a big support in my life, and I appreciate all your unconditional love. I want you to know that this achievement and all others that come are due to you.

Finally, I take this opportunity to express my gratitude to each and everyone who directly or indirectly supported me until the completion of my degree, as this achievement would not have been possible without the guidance and blessings of many. Thank you.

# Table of Contents

# List of Figures

# List of Tables

# CHAPTER 1

# General introduction

## 1.1 Motivation

Many data-driven problems involve estimating an input-output relation based on a finite set of (possibly noisy) measurements, i.e., $f(x_i) \quad y_i$. Assuming that $f$ is linear, expressed as $f(x) = h \quad j\, x i$, the problem simplifies to determine $\quad$ from the following linear equation:

$$X \quad = y;$$

where $X$ represents a matrix with rows corresponding to input data points, and $y$ can be seen as the vector of measurements of some unknown $\quad$ that we want to recover.

In general, the solution might not exist or might not be unique, or might not depend continuously on the data $y$. To address the issue of existence, a loss or data-fitting function $L$, often the least squares, is utilized. To solve non-uniqueness and select a particular solution, a regularization term $R$ is introduced, also called the bias of the solution. This regularization enforces prior knowledge or a desired property of the solution. A standard approach to recover $\quad$ is to assume that it is a minimizer of the following constrained optimization problem:

$$\min_{2R^p} fR(\ ) : \quad 2 \operatorname{argmin} Lg: \tag{1.1.1}$$

Typically, in (1.1.1), the regularizer $R$ and the loss function $L$ are fixed. Subsequently, a suitable algorithm that leverages the properties of both the loss and the regularizer is designed to efficiently find a solution. In this thesis, we propose two different approaches to solve the implicit problem.

We also assume that $L$ is the least squares and the data is noisy, meaning that there exists $y$ such that:

$$ky \quad yk \quad ;$$

where $\quad$ is the level of noise. A common method to solve the above problem and avoid instabilities is Tikhonov regularization [52, 128] which consists in minimizing the sum of a penalized loss term plus a regularizer. A trade-off parameter is then introduced to balance the fidelity term and the regularizer. In practice, this implies that the optimization problem has to be solved many times for different values of the parameter. Then, the best performing iterate is chosen according to some a priori data driven criterion and considered the regularized solution.

In the first part of our thesis, we solve (1.1.1) using a classical approach, to deal the case where $R$ is a convex regularizer, which is neither smooth nor strongly convex. We explore an alternative, efficient approach known as early stopping [12, 20, 27, 30, 82, 112, 141, 146]. This method runs an iterative algorithm solving the exact problem but on the inexact data and early stopping to prevent convergence to the noisy solution. In this setting, the number of iterations plays the role of the regularization parameter. Compared to Tikhonov regularization, this procedure is very efficient since only one optimization problem is solved, and not even until convergence. Recently, in [40, 84] an algorithm was introduced that combined primal-dual methods with early stopping. On the other side, in [24], an extra activation step is added, improving the feasibility of the iterates, obtaining empirical speed up. Then, the idea is to combine these two approaches to design an efficient algorithm to solve (1.1.1).

We introduce and analyze two new iterative regularization methods based on a primal-dual algorithm with activations. Our analysis yields in the noise-free scenario ( = 0) convergence rates for the Lagrangian and the feasibility gap of ($P$), and stability bounds along with early stopping criteria with theoretical guarantees in the noisy scenario. The main novelty of our work is the exploitation of some a priori knowledge about the solution set: we show that the linear equations determined by the data can be used more than once along the iterations. We propose various strategies for reusing linear equations that align with our theoretical framework while remaining flexible for practical implementation. Finally, we illustrate our theoretical findings with numerical simulations for robust sparse recovery and image reconstruction. We confirm the efficiency of the proposed regularization approaches by comparing the results with state-of-the-art methods.

In the second part of this thesis, we flip this perspective by fixing the loss and the algorithm, which, to simplify the analysis, will be gradient flow. We also reparameterize the linear model as follows:

$$\overline{?} \quad = q( \ )$$

which is called re- or over-parameterization. Then, we want to answer the following question:

**For a given loss, algorithm, and reparameterization, what implicit bias is introduced by the chosen optimization method and reparameterization?**

In simpler terms, does there exist a regularization function $R$ such that the output of gradient flow applied to the loss function $L$ for a specific reparameterization $q$ is the solution of problem (1.1.1)?.

This question was partially answered in [4, 77], which found that gradient flow on the reparameterization and mirror flow on are equivalent, for a suitable mirror map that only depends on reparametrization. While these articles cover a wide range of reparameterizations, their assumptions are too restrictive to include certain interesting reparameterizations, such as weight normalization and the multiplication of a matrix by a vector.

To address this limitation, we study a unified framework proposed in [7], called time-warp mirror flow. This consists of the vanilla mirror flow with a scalar preconditioning that allows us to encompass many reparameterizations presented in the state of the art. However, the theoretical analysis of the existence and convergence of the trajectory is missing. Then, we pose the following question:

**Can we expect well-posedness, convergence to a stationary point, convergence rates, and implicit bias of the sequence $(t)$ generated by time-warped mirror flow?**

In this thesis, we provide a set of assumptions to prove the well-posedness of the trajectory of the time-warping mirror flow. Our analysis includes convergence of both the trajectory and the corresponding values of the loss function. In the case when the loss function is the composition of a linear operator with a strictly convex function, we provide an explicit expression of the implicit bias. Finally, we illustrate our theoretical findings by applying the obtained result to weight normalization and matrix vector overparameterization.

## 1.2 Main contributions

We briefly mention the main contributions of our work.

- **In Chapter 4**, we design and analyze two new iterative regularization methods for convex regularizers, which are not necessarily smooth nor strongly convex. The new iterative regularization methods are based on primal-dual algorithms [40, 46, 133] combined with the idea of reusing the linear equations determined by the data at every iteration [24]. The first method that we propose is a primal-dual algorithm with additional activations of the linear equations. The second method is a dual-primal algorithm, where a subset containing the dual solutions is activated at each step.

  Our analysis, in the noisy case, provides stability bounds and early stopping rules with theoretical guarantees. In the noise-free case, we provide convergence rates for the Lagrangian and the feasibility gap of the problem $(P)$.

  We propose different variants of our algorithm, using different extra activation steps, including a gradient descent step over the least square with a fixed or adaptive step size. We also compare their numerical performance with state-of-the-art methods, obtaining a considerable improvement in run-time.

- **In Chapter 6**, we provide an implicit bias for reparametrizations such that applying gradient flow over is equivalent to apply time-warped mirror flow over . The analysis consists of several steps. First, we establish conditions for the well-posedness of the time warped mirror flow. Second, we demonstrate that for any convex function, the sequence $(t)$ generated by time-warping mirror flow converges to a stationary point that minimizes the loss function while avoiding the extra stationary points that the reparameterization produces. For the specific case of a strictly convex function composed with a linear operator, an explicit expression for the implicit bias is provided.

  We apply our results to matrix vector parameterization and to various weight normalization reparameterizations, generalizing many results in the state of the art. Furthermore, we give a criterion to determine, for a given function that depends only on the norm, a suitable weight normalization parameterization. Finally, we explore the flexibility of our formulation by applying the previous results to different examples related to weight normalization.

### Journal publications

- C. Vega, C. Molinari, S. Villa, and L. Rosasco, "Fast iterative regularization by reusing data", Journal of Inverse and Ill-posed Problems, 2023.

**Articles in preparation**

- C. Vega, C. Molinari, S. Villa, and L. Rosasco, "Learning from data via overparameterization", In preparation, 2024.

## 1.3 Outline

This section offers a concise overview of the thesis, which is organized into six chapters and supplemented by one appendix. Chapters 4 and 6 comprise the core articles developed during the doctoral research, one of them published in a journal [132]. Chapters 3 and 5 provide technical introductions that precede the discussions in Chapters 4 and 6, respectively.

Chapter 2 lays out the notations and mathematical framework essential for this thesis, organized into four sections. The first section is dedicated to Hilbert spaces, establishing the fundamental concepts used throughout the thesis. The second section delves into matrices, presenting the results necessary for our analysis. The third section explores convex optimization concepts, which are mainly, but not exclusively, applied in Chapter 4. In the last section, we introduce essential notions of calculus, ordinary differential equations, and the algorithms used in this thesis necessary for the discussion in Chapter 5.

Chapter 3 presents implicit and iterative regularization approaches. Initially, it outlines the four fundamental components of implicit regularization: the model, the loss, the bias, and the algorithm. Subsequently, the chapter provides a review of iterative algorithms.

Chapter 4 is organized as follows. In Section 4.2 we present the main problem and propose two algorithms to solve it numerically. In Section 4.3 we derive stability and feasibility gap bounds and related early stopping rules. In Section 4.4 we verify the performance of the algorithm on two numerical applications: robust sparse recovery problem and image reconstruction by total variation. Finally, we provide some conclusions.

Chapter 5 formulates our problem as an implicit regularization problem. Next, from the point of view of optimization, we explain how overparameterization induces some implicit bias and we give some examples. Finally, we present the main results in the state of the art for vanilla and time-warped mirror flow.

Chapter 6 is organized as follows. In Section 6.2, we derive the convergence of our algorithm to a stationary point and characterize this solution. In Section 6.3, we derive conditions to cast gradient flow on weight normalization reparameterization, which decomposes a vector into its direction and its norm, as a mirror flow over a radial function. Finally, we provide some conclusions and open problems for future research directions.

Additionally, the thesis includes an appendix: which offers the supplementary material of Chapter 6, omitted from the main text due to its extension. This section is mainly focused on examples of overparameterization.

# CHAPTER 2

# Notation and Preliminaries

We begin by introducing some notation used throughout this thesis.

## 2.1 Basic notions

Let $\mathsf{H}$, $\mathsf{G}$ be two real vector spaces. Although in most of this thesis $\mathsf{H} = \mathsf{R}^p$ and $\mathsf{G} = \mathsf{R}^d$, which are finite dimensional, this section is written for general Hilbert spaces. For further results, the reader is referred to [15].

**Definition 2.1.1.** Now, we present some examples of norms in $\mathsf{R}^p$, which will be used in this thesis.

- **General $\ell_q$-norm:** Let $\quad 2 \mathsf{R}^p$. For $q \quad 1$, the $\ell_q$-norm is defined as:

$$k \quad k_q = \left( \sum_{i=1} j \quad {}_i j^q \right)^{\frac{1}{q}}$$

  For $q = 1; 2;$ and $1 ;$ we have that

$$k \quad k_1 = \sum_{i=1} j \quad {}_i j; \quad k \quad k_2 = \sqrt{\sum_{i=1} {}_i^2} \text{ and } k \quad k_1 = \max_{1 \quad i \quad p} j \quad {}_i j:$$

- **$\ell_{1;2}$-norm**: Let $\quad$ be a square matrix in $\mathsf{R}^{p \quad p}$. The $\ell_{1;2}$-norm is defined as:

$$k \quad k_{1;2} : \mathsf{R}^{p \quad p} \quad \mathsf{R}^{p \quad p} \; ! \; \mathsf{R}; \qquad ! \sum_{i=1} \sum_{j=1} k( {}_{ij}^1; {}_{ij}^2 ) k_2;$$

  where from now on ${}_{ij}$ is $i$-th component of the $j$-th column of $\quad$. This norm will be used in the numerical examples.

**Remark 2.1.2.** The following two fundamental inequalities are continuously employed throughout this thesis. For every $( \; ; z) \; 2 \; \mathsf{H} \quad \mathsf{H}$ and $\quad > 0$, we have that:

$$h \quad j \, z i \quad k \quad k k z k \quad \frac{k z k^2}{2} + \frac{k \quad k^2}{2} :$$

The first and second inequalities are known as the Cauchy-Schwarz and Young inequalities, respectively.

The open ball of radius $r > 0$ centered at $\in$ H is the set of all points in H whose distance from is strictly less than $r$. This can be formally written as:

$$B( ; r) = fz \in \mathsf{H} \mid kz \quad k < rg:$$

Analogously, the closed ball of radius $r > 0$ centered at $\in$ H is the set of all points in H whose distance from is less than or equal to $r$ and is denoted by $B( ; r)$. This can be formally written as:

$$B( ; r) = fz \in \mathsf{H} \mid kz \quad k \quad rg:$$

**Definition 2.1.3.** Given $\in ]0; 1[$; an operator $T : \mathsf{H} \to \mathsf{H}$ is

- -averaged non-expansive, if for all $( ; z) \in \mathsf{H} \quad \mathsf{H}$;

$$kT \quad Tzk^2 \quad k \quad zk^2 \quad \frac{1}{}k(\mathrm{Id} \quad T) \quad (\mathrm{Id} \quad T)zk^2:$$

- Firmly nonexpansive if it is $\frac{1}{2}$ average.
- Non-expansive if, for all $( ; z) \in \mathsf{H} \quad \mathsf{H}$;

$$kT \quad Tzk^2 \quad k \quad zk^2:$$

- Quasi-non-expansive if

$$kT \quad zk^2 \quad k \quad zk^2;$$

for all $\in \mathsf{H}$ and all $z \in \mathrm{Fix}\, T$, where $\mathrm{Fix}\, T = f \in \mathsf{H} \mid T = g$ is the set of fixed points of $T$.

The next lemma allows us to bound a positive sequence (see [111, Lemma A.1] for further references).

**Lemma 2.1.4.** Assume that the sequence $f \ _N g$ is a non-negative and satisfies the recursion

$$\frac{2}{N} \quad S_N + \sum_{k=1}^{N} \ _k \ _k; \tag{2.1.1}$$

for all $N \quad 1$, where $fS_N g$ is an increasing sequence, $S_0 \quad \frac{2}{0}$, and $_k \quad 0$ for all $k \quad 0$. Then for all $N \quad 1$

$$_N \quad \frac{1}{2}\sum_{k=1}^{N} \ _k + \left(S_N + \frac{1}{2}\sum_{k=1}^{N} \ _k\right)^{\frac{1}{2}} : \tag{2.1.2}$$

The following lemma allows us to prove convergence of a given sequence.

**Lemma 2.1.5** (Opial's Lemma)**.** Let $C$ be a nonempty subset of a finite dimensional Hilbert space H and let $^k \ _{k\in\mathsf{N}}$ be a sequence in H: Assume

- For every $\in C$ there exists $\lim_{k \to +1} k \ _k \quad k$; and
- Every limit point of $^k \ _{k\in\mathsf{N}}$ belongs to $C$.

Then $^k \ _{k\in\mathsf{N}}$ converges as $k \to +1$ to some $_1 \in C$:

## 2.2 Matrices

This section introduces various fundamental definitions and results in matrix theory and linear algebra.

We use lowercase and uppercase letters, for matrices and vectors, respectively. For any positive integer $p$, the set of all integers from 1 to $p$ is denoted by $[p]$. The Hadamard product of two vectors and $z$ in $\mathbb{R}^p$ is represented as $z$, where, for every $i$ in $[p]$, $( z)_i = {}_i z_i$. For a positive scalar $L$, the vector ${}^L := {}_{i\ i=1}^{L\ p}$. The vector $\tilde{}$ in $\mathbb{R}^p$ has all components equal to . The diagonal matrix $\mathrm{Diag}( ) \in \mathbb{R}^{p\ p}$, for a vector $\in \mathbb{R}^p$, has the components of on its main diagonal and zeros elsewhere.

The operator norm for a matrix $X \in \mathbb{R}^{d\ p}$ is defined by

$$kXk = \sup_{\substack{z \in \mathbb{R}^p \\ kzk\ 1}} kXzk:$$

The Frobenius norm of $X$, represented as $kXk_F$, is defined as

$$kXk_F^2 := \sum_{i=1}^{\not} kx_ik^2;$$

where $x_i$ is the $i$-th row of $X$. This norm extends the Euclidean norm to matrices.

The adjoint (or transpose) of a matrix $X \in \mathbb{R}^{d\ p}$, denoted by $X^>$, is the unique matrix satisfying $hX\ \ j\ zi = \ \ j\ X^>z$ , for every $\in \mathbb{R}^p$ and $z \in \mathbb{R}^d$. Two vectors $( \ ;z) \in \mathbb{R}^p \quad \mathbb{R}^p$ are orthogonal if and only if $h\ \ j\ zi = 0$. For a subset $V \quad \mathbb{R}^p$, its orthogonal complement is defined as:

$$V^? = f \in \mathbb{R}^p \ j \text{ for all } z \in V;\ \ h\ \ j\ zi = 0g:$$

The range of $X \in \mathbb{R}^{d\ p}$ is denoted by $\mathrm{ran}(X)$ and defined as $fu \in \mathbb{R}^d \ j\ 9$ such that $X = ug:$ Similarly, the kernel of $X$ is denoted by $\ker(X)$ and defined as $f \in \mathbb{R}^d \ j\ X = 0g:$ In the finite-dimensional setting, it follows that both $\ker(X)$ and $\mathrm{ran}(X)$ are closed subspaces, and we have the relationship $\mathrm{ran}(X) = \ker(X)^?:$ An orthogonal matrix, or orthonormal matrix, is a real square matrix $X \in \mathbb{R}^{p\ p}$ whose columns and rows are orthonormal vectors.

A matrix $M \in \mathbb{R}^{p\ p}$ is positive definite if ${}^>M > 0$ for every non-zero vector $\in \mathbb{R}^p$. Similarly, a matrix is positive semidefinite if ${}^>M \quad 0$ for every vector $\in \mathbb{R}^p$. The set of all $p \quad p$ positive semidefinite and positive definite matrices are represented by $S_+^k$ and $S_{++}^k$, respectively.

**Definition 2.2.1.** Given a matrix $X \in \mathbb{R}^{p\ d}$, its Singular Value Decomposition (SVD) is a factorization of the form $X = U\ V^>$, where:

- $U \in \mathbb{R}^{p\ p}$ and $V \in \mathbb{R}^{d\ d}$ are orthogonal matrices.
- $\in \mathbb{R}^{p\ d}$ is a diagonal matrix with non-negative real numbers on the diagonal.

These diagonal entries of are known as the singular values of $X$, and the columns of $U$ and $V$ are called the left-singular vectors and right-singular vectors of $X$, respectively.

**Definition 2.2.2.** Given a matrix $X \in \mathbb{R}^{p \times d}$, if the singular values of the $X$ matrix are denoted by $\sigma_i$ for every $i \in [\min\{p, d\}]$, then the nuclear-norm $\|\cdot\|_*$ is defined by:

$$\|X\|_* = \sum_{i=1}^{\min\{p,d\}} \sigma_i.$$

The next formula provides an efficient method for calculating the inverse of a matrix plus a rank-one matrix, specifically for $(M + z z^\top)$ where $M$ is an invertible $p \times p$ matrix and $z, \alpha \in \mathbb{R}^p$.

**Lemma 2.2.3.** If $M$ is an invertible $p \times p$ matrix and $z$ and $\alpha$ are two $p$-dimensional vectors such that $\alpha^\top M^{-1} z \neq -1$, the Sherman–Morrison formula gives

$$(M + z \alpha^\top)^{-1} = M^{-1} - \frac{M^{-1} z \alpha^\top M^{-1}}{1 + \alpha^\top M^{-1} z}. \tag{2.2.1}$$

The following definition will be used in the numerical examples.

**Definition 2.2.4.** The discrete gradient operator $D : \mathbb{R}^{p \times p} \to (\mathbb{R}^2)^{p^2}$, for every matrix $u \in \mathbb{R}^{p \times p}$ is defined by:

$$(Du)_{ij} = ((D_x u)_{ij}, (D_y u)_{ij}),$$

where

$$(D_y u)_{ij} = \begin{cases} u_{i+1,j} - u_{i,j}, & \text{if } 1 \leq i \leq N-1; \\ 0, & \text{if } i = N; \end{cases}$$

$$(D_x u)_{ij} = \begin{cases} u_{i,j+1} - u_{i,j}, & \text{if } 1 \leq j \leq N-1; \\ 0, & \text{if } j = N. \end{cases}$$

## 2.3 Convex optimization

This section introduces various fundamental definitions and results in convex optimization. For further results on convex analysis and operator theory, the reader is referred to [15, 107].

**Definition 2.3.1.** Let $f : \mathcal{H} \to [-\infty, +\infty]$. The domain of $f$ is

$$\mathrm{dom}(f) = \{x \in \mathcal{H} \mid f(x) < +\infty\},$$

the graph of $f$ is

$$\mathrm{gra}(f) = \{(x, y) \in \mathcal{H} \times \mathbb{R} \mid f(x) = y\},$$

the epigraph of $f$ is

$$\mathrm{epi}(f) = \{(x, y) \in \mathcal{H} \times \mathbb{R} \mid f(x) \leq y\}.$$

The function $f$ is proper if $-\infty \notin f(\mathcal{H})$ and $\mathrm{dom}(f) \neq \emptyset$.

**Definition 2.3.2.** The argmin of a proper function $f : \mathcal{H} \to ]-\infty, +\infty]$ is

$$\mathrm{argmin}(f) := \{x \in \mathcal{H} \mid f(x) \leq f(y) \text{ for all } y \in \mathcal{H}\}.$$

**Definition 2.3.3.** A subset $C \subseteq \mathcal{H}$ is convex if and only if

$$\lambda x_1 + (1 - \lambda) x_2 \in C$$

for every $x_1, x_2 \in C$ and $\lambda \in (0, 1)$.

**Definition 2.3.4.** A function $f : H \to [-\infty, +\infty]$ is convex if

$$f(\lambda x_1 + (1-\lambda) x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$$

for every $x_1, x_2 \in \mathrm{dom}(f)$ and $\lambda \in (0,1)$.

**Definition 2.3.5.** A function $f : H \to \mathbb{R} \cup \{+\infty\}$ is lower semicontinuous (l.s.c.) if and only if $\mathrm{epi}(f)$ is closed in $H \times \mathbb{R}$.

We denote by $\Gamma_0(H)$ the set of convex, lower semicontinuous, and proper functions on $H$.

**Definition 2.3.6.** Let $f : H \to \mathbb{R} \cup \{+\infty\}$ be a proper and convex function. A point $u \in H$ is a subgradient of $f$ at $x$ if, for all $z \in H$,

$$f(z) \geq f(x) + \langle u \mid z - x \rangle.$$

The set of all subgradients of $f$ at $x$ is the subdifferential of $f$ at $x$ and is denoted by $\partial f(x)$. If $\partial f(x) \neq \emptyset$, we say that $f$ is subdifferentiable at $x$. The domain of the subdifferential is

$$\mathrm{dom}(\partial f) = \{x \in H \mid \partial f(x) \neq \emptyset\}.$$

Note that, by definition,

$$\mathrm{dom}(\partial f) \subset \mathrm{dom}(f).$$

**Proposition 2.3.7.** Let $f : H \to \mathbb{R} \cup \{+\infty\}$ be a convex function. If $u_1 \in \partial f(x_1)$ and $u_2 \in \partial f(x_2)$, then $\langle u_1 - u_2 \mid x_1 - x_2 \rangle \geq 0$.

**Example 2.3.8.** Let $f : \mathbb{R} \to \mathbb{R} \cup \{+\infty\} : x \mapsto |x|$. Then, $\partial f(x) = \{1\}$ if $x > 0$, $\partial f(x) = \{-1\}$ if $x < 0$, and $\partial f(0) = [-1, 1]$.

**Example 2.3.9.** Let $f : H \to \mathbb{R} \cup \{+\infty\} : x \mapsto \|x\|$. Then, $\partial f(x) = \frac{x}{\|x\|}$ if $x \neq 0$, and $\partial f(0) = \overline{B}(0,1)$, where $\|\cdot\|$ is the norm induced by the scalar product.

**Example 2.3.10.** Let $C$ a subset of $H$. Define the indicator function of $C$ as

$$\iota_C(x) = \begin{cases} 0, & \text{if } x \in C; \\ +\infty, & \text{if } x \notin C. \end{cases}$$

If $C \subset H$ is a nonempty, convex, and closed subset, then $\iota_C \in \Gamma_0(H)$. Moreover, $\partial \iota_C = N_C$, where, for every $x \in H$,

$$N_C(x) = \begin{cases} \{u \in H \mid \langle z - x \mid u \rangle \leq 0, \text{ for all } z \in C\} & \text{if } x \in C; \\ \emptyset; & \text{if } x \notin C. \end{cases}$$

The following definition allows us to define a Hilbert space by the product of a finite number of Hilbert spaces.

**Definition 2.3.11.** The Hilbert direct sum of a family of real Hilbert spaces $(H_i)_{i \in [d]}$ is the real Hilbert space:

$$\bigoplus_{i \in [d]} H_i = \left( x = (x_i)_{i \in [d]} \in \prod_{i \in [d]} H_i \right).$$

For every $i \in [d]$, let $f_i : H_i \to \mathbb{R} \cup \{+\infty\}$. Then,

$$\bigoplus_{i \in [d]} f_i : \bigoplus_{i \in [d]} H_i \to \mathbb{R} \cup \{+\infty\} : (x_i)_{i \in [d]} \mapsto \sum_{i=1}^{d} f_i(x_i).$$

**Proposition 2.3.12.** Consider a finite family of functions $\{f_i\}_{i\in[d]}$ such that, for every $i \in [d]$, $f_i : \mathsf{H}_i \to \mathbb{R} \cup \{+\infty\}$ is proper and convex. Then,

$$\partial \sum_{i\in[d]} f_i = \bigoplus_{i\in[d]} \partial f_i.$$

The following lemma allows us to characterize the minimizer of a function.

**Theorem 2.3.13.** Let $f : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}$ be a proper and convex function. Then

$$\operatorname{argmin} f = \{x \in \mathbb{R}^p \mid 0 \in \partial f(x)\}.$$

For a given set $C$, we denoted by $\overline{C}$ the smallest closed subset containing $C$. The relative interior of $C$ is

$$\operatorname{ri}(C) = \{x \in C \mid \mathbb{R}_{++}(C - x) = \overline{\operatorname{span}}(C - x)\},$$

where

$$\mathbb{R}_{++}C = \{\lambda y \mid (\lambda > 0) \wedge (y \in C)\}$$

and $\operatorname{span}(C)$ is the smallest linear subspace of $\mathsf{H}$ containing $C$. From now on, every time we use ri we assume that we are in a finite dimensional space.

**Proposition 2.3.14.** Let $f : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}$, and $g : \mathsf{G} \to \mathbb{R} \cup \{+\infty\}$ be two proper, l.s.c., and convex functions. Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator. Suppose that $\operatorname{dom} g \cap X \operatorname{dom} f \neq \emptyset$. Then, $\partial f + X^* \partial g X \subseteq \partial(f + g X)$. Additionally, if we suppose that $0 \in \operatorname{ri}(\operatorname{dom} g - X \operatorname{dom} f)$, then $\partial f + X^* \partial g X = \partial(f + g X)$.

**Definition 2.3.15.** Let $f : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}$ be in $\Gamma_0(\mathsf{H})$. The conjugate (or Fenchel conjugate, or Legendre transform, or Legendre–Fenchel transform) of $f$ is

$$f^* : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}, u \mapsto f^*(u) := \sup_{x\in\mathsf{H}} \langle x \mid u\rangle - f(x).$$

**Proposition 2.3.16.** Consider a finite family of functions $\{f_i\}_{i\in[d]}$ such that, for every $i \in [d]$, $f_i : \mathsf{H}_i \to \mathbb{R} \cup \{+\infty\}$ is proper and convex. Then,

$$\partial \Big( \sum_{i\in[d]} f_i \Big) = \sum_{i\in[d]} f_i.$$

**Example 2.3.17.** Let $C$ be a nonempty, closed, and convex set. Then,

$$(\iota_C)^* = \sigma_C : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}, u \mapsto \sup_{x\in C} \langle x \mid u\rangle.$$

Note that if $C = \{y\}$, then $\sigma_C(u) = \langle u \mid y\rangle$.

**Example 2.3.18.** Set $B_1 = \{x \in \mathsf{H} \mid \max_{i\in[p]} |\langle x \mid e_i\rangle| \leq 1\}$. Then, $(\|\cdot\|_1)^* = \|\cdot\|_{B_1}$.

**Proposition 2.3.19.** Let $f \in \Gamma_0(\mathsf{H})$. Then $(\partial f)^{-1} = \partial f^*$, where the inverse of $\partial f$, is defined through its graph

$$\operatorname{gra}(\partial f)^{-1} = \{(u, x) \in \mathsf{H} \times \mathsf{H} \mid u \in \partial f(x)\}$$

**Definition 2.3.20.** Let $f : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}$, and $g : \mathsf{G} \to \mathbb{R} \cup \{+\infty\}$ be two proper, l.s.c., and convex functions. Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator. The primal problem is given by:

$$\inf_{x \in \mathsf{H}} f(x) + g(Xx).$$

Its optimal value is denoted by $\mu$ and the set of primal solutions is $S$. The dual problem is given by:

$$\inf_{u \in \mathsf{H}} f^*(-X^*u) + g^*(u).$$

Its optimal value is denoted by $\nu$ and the set of dual solutions is $S^*$.

**Theorem 2.3.21.** Let $f : \mathsf{H} \to \mathbb{R} \cup \{+\infty\}$, and $g : \mathsf{G} \to \mathbb{R} \cup \{+\infty\}$ be two proper, l.s.c., and convex functions. Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator. Then, $\mu + \nu \geq 0$. Additionally, if

$$0 \in \operatorname{ri}(\operatorname{dom} g - X \operatorname{dom} f),$$

then $\mu + \nu = 0$.

The next theorem characterize the primal-dual solutions.

**Theorem 2.3.22.** Let $x \in \mathsf{H}$ and $u \in \mathsf{G}$. Then the following are equivalent:

1. $-X^*u \in \partial f(x)$ and $u \in \partial g(Xx)$;
2. $f(x) + g(Xx) + f^*(-X^*u) + g^*(u) = 0$; and
3. $x \in S$ and $u \in S^*$ and $\mu + \nu = 0$.

**Definition 2.3.23.** The proximal operator of $f \in \Gamma_0(\mathsf{H})$ is defined by

$$\operatorname{prox}_f : \mathsf{H} \to \mathsf{H}, \quad x \mapsto \operatorname*{argmin}_{z \in \mathsf{H}} f(z) + \frac{1}{2}\|x - z\|^2.$$

**Proposition 2.3.24.** Let $f \in \Gamma_0(\mathsf{H})$, and let $x$ and $p$ be in $\mathsf{H}$. Then $p = \operatorname{prox}_f(x)$ if and only if $x - p \in \partial f(p)$.

For every self-adjoint positive definite matrix $M$, we define the proximity operator of $f$ relative to the metric induced by $\|x\|_M^2 := \langle x \mid Mx \rangle$ as $\operatorname{prox}_f^M = (\mathrm{Id} + M^{-1}\partial f)^{-1}$. If $M = \mathrm{Id}$ for some real number $\gamma > 0$, it is customary to write $\operatorname{prox}_{\gamma f}$ rather than $\operatorname{prox}_f$.

**Proposition 2.3.25.** Let $f \in \Gamma_0(\mathsf{H})$ and let $\gamma > 0$. Then $\operatorname{prox}_f(x) = x - \gamma \operatorname{prox}_{\frac{1}{\gamma} f^*}(\tfrac{x}{\gamma})$.

**Proposition 2.3.26.** Consider a finite family of functions $(f_i)_{i \in [d]}$ such that, for every $i \in [d]$, $f_i : \mathsf{H}_i \to \mathbb{R} \cup \{+\infty\}$ is in $\Gamma_0(\mathsf{H}_i)$. Set $x = (x_i)_{i \in [d]}$ and $f = \bigoplus_{i \in [d]} f_i$. Then, $\operatorname{prox}_f(x) = (\operatorname{prox}_{f_i}(x_i))_{i \in [d]}$.

**Example 2.3.27.** Let $\gamma > 0$ and $x \in \mathsf{H}$. Then,

$$\operatorname{prox}_{\gamma \|\cdot\|_1}(x) = (\operatorname{soft}(x_i))_{i \in [p]},$$

where

$$\operatorname{soft}(x) = \begin{cases} x + \gamma; & \text{if } x < -\gamma; \\ 0; & \text{if } x \in [-\gamma, \gamma]; \\ x - \gamma; & \text{if } x > \gamma; \end{cases}$$

and

$$\operatorname{prox}_{\gamma \|\cdot\|_1}(x) = \frac{x}{\max\{|x|/\gamma, 1\}}.$$

**Definition 2.3.28.** The projection operator onto a non-empty closed convex set $C \subseteq \mathsf{H}$ is defined by

$$P_C : \mathsf{H} \to \mathsf{H}; \quad x \mapsto \operatorname*{argmin}_{z \in C} \frac{1}{2}\|x - z\|^2.$$

Let $f = \iota_C \in \Gamma_0(\mathsf{H})$, then $\operatorname{prox}_{\iota_C} = P_C$. The distance to $C$ is defined as $\operatorname{dist}^2(x; C) = \min_{z \in C} \|x - z\|^2 = \|x - P_C x\|^2$.

**Example 2.3.29.** Let $x$ be a non-zero vector in $\mathsf{H}$, let $y \in \mathbb{R}$, and set

$$C = \{z \in \mathsf{H} \mid \langle z \mid x \rangle = y\}.$$

Then

$$P_C x = x + \frac{y - \langle x \mid x \rangle}{\|x\|^2} x.$$

**Example 2.3.30.** Let $x$ be a non-zero vector in $\mathsf{H}$, let $(y_1, y_2) \in \mathbb{R}^2$, such that $y_2 \geq y_1$, and set $C = \{z \in \mathsf{H} \mid y_1 \leq \langle z \mid x \rangle \leq y_2\}$. Then

$$P_C x = \begin{cases} x + \frac{y_1 - \langle x \mid x \rangle}{\|x\|^2} x, & \text{if } \langle x \mid x \rangle < y_1; \\ x, & \text{if } y_1 \leq \langle x \mid x \rangle \leq y_2; \\ x + \frac{y_2 - \langle x \mid x \rangle}{\|x\|^2} x, & \text{if } \langle x \mid x \rangle > y_2. \end{cases}$$

**Theorem 2.3.31.** Let $C$ be a nonempty closed and convex subset of $\mathsf{H}$. Then, for every $x \in \mathsf{H}$, $P_C x \in C$, and for every $z \in C$,

$$\langle z - P_C x \mid x - P_C x \rangle \leq 0.$$

Additionally, if $C$ is a closed affine subspace, then $P_C$ is an affine operator, and for every $z \in C$, $\langle z - P_C x \mid x - P_C x \rangle = 0$.

In the following two definitions and two propositions, we assume that $\mathsf{H}$ and $\mathsf{G}$ are finite dimensional.

**Definition 2.3.32.** Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator and let $y \in \mathsf{G}$. Then $w$ is a least square solution to the linear system $Xz = y$ if

$$w \in \operatorname*{argmin}_{z \in \mathsf{H}} \|Xz - y\|^2.$$

**Proposition 2.3.33.** Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator and let $y \in \mathsf{G}$. Then the equation $Xz = y$ has at least one least squares solution. Moreover, for every $w \in \mathsf{H}$, the following are equivalent:

1. $w$ is the least square solution.
2. $Xw = P_{\operatorname{ran}(X)} y$.
3. $X^* Xw = X^* y$.

**Definition 2.3.34.** Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator. Let $C_y = \{w \in \mathsf{H} \mid X^* Xw = X^* y\}$. The generalized (or Moore-Penrose) inverse of $X$ is $X^\dagger : \mathsf{G} \to \mathsf{H} : y \mapsto P_{C_y}(0)$.

**Proposition 2.3.35.** Let $X : \mathsf{H} \to \mathsf{G}$ a nonzero, bounded, and linear operator, such that $\operatorname{ran}(X)$ is closed. Then $X^\dagger$ is a bounded linear operator, $P_{\operatorname{ran}(X)} = XX^\dagger$, and $P_{\ker(X)} = \mathrm{Id} - X^*(X^*)^\dagger$.

## 2.4 Differential Calculus and Ordinary differential equations

In this section, we recall some basic facts of Differential Calculus and Ordinary differential equations.

**Definition 2.4.1.** Let $f : H \to \mathbb{R} \cup \{+\infty\}$. The directional derivative of $f : H \to \mathbb{R} \cup \{+\infty\}$ at $x \in \text{int}(\text{dom}(f))$ in the direction $d \in H$ is:

$$f'(x;d) = \lim_{t \to 0^+} \frac{f(x+td) - f(x)}{t};$$

whenever the limit exists. The function $f$ is Gâteaux differentiable at the point $x$ if $f'(x;d)$ exists for all $d \in H$ and the function $d \mapsto f'(x;d)$ is linear and continuous. In this situation, the Gâteaux derivative (or gradient) of $f$ at $x$ is $\nabla f(x) = f'(x;\cdot)$.

As before, we can define a directional derivative of $\nabla f$ at $x \in \text{int}(\text{dom}(f))$ in the direction $d \in H$ as:

$$(\nabla f)'(x;d) = \lim_{t \to 0^+} \frac{\nabla f(x+td) - \nabla f(x)}{t}$$

whenever the limit exists. The function $f$ is twice Gâteaux differentiable at the point $x$ if $(\nabla f)'(x;d)$ exists for all $d \in H$ and the function $d \mapsto (\nabla f)'(x;d)$ is linear and continuous. In this situation, the second Gâteaux derivative (or hessian) of $f$ at $x$ is denoted by $\nabla^2 f(x) = (\nabla f)'(x;\cdot)$.

**Proposition 2.4.2.** Let $f : H \to \mathbb{R} \cup \{+\infty\}$ be proper and convex function, and let $x \in \text{dom}(f)$. If the function $f$ is Gâteaux differentiable at the point $x$, then $\partial f(x) = \{\nabla f(x)\}$.

**Lemma 2.4.3** (Descent Lemma)**.** If $f : H \to \mathbb{R} \cup \{+\infty\}$ is Gâteaux-differentiable and $\nabla f$ is Lipschitz-continuous with constant $L$, then:

$$f(z) \leq f(x) + \langle \nabla f(x) \mid z - x \rangle + \frac{L}{2}\|z-x\|^2;$$

for every $(z, x) \in \text{dom}(f) \times \text{int}(\text{dom}(f))$.

**Definition 2.4.4.** Let $f : H \to \mathbb{R} \cup \{+\infty\}$ be strictly convex, proper, and Gâteaux differentiable on $\text{int dom } f \neq \emptyset$. The Bregman divergence $D_f$ associated with $f$ is defined as:

$$D_f : H \times H \to \mathbb{R} \cup \{+\infty\}$$
$$(z, x) \mapsto \begin{cases} f(z) - f(x) - \langle \nabla f(x) \mid z - x \rangle, & \text{if } x \in \text{int}(\text{dom}(f)); \\ +\infty, & \text{otherwise.} \end{cases} \tag{2.4.1}$$

Note that, for every $(z, x) \in \text{dom}(f) \times \text{int}(\text{dom}(f))$, the Bregman distance $D_f(z, x)$ is positive and $D_f(z, x) = 0 \Leftrightarrow x = z$.

The next Lemma provides a sufficient condition to determine if a vector field is a gradient.

**Lemma 2.4.5.** Let $f = (f_1, \ldots, f_p) : U \subset \mathbb{R}^p \to \mathbb{R}^p$ let be a continuously differentiable vector field on an open set $U$. If $f$ is a gradient on $U$, then the partials derivatives of the components of $f$ are related by the equations:

$$\frac{\partial f_i}{\partial x_j}(x) = \frac{\partial f_j}{\partial x_i}(x);$$

for all $i, j \in [p]$ and $x \in U$.

**Definition 2.4.6.** Let $U \subseteq \mathbb{R}^p$ be a non-empty open set. Let a function $f : U \to \mathbb{R} \cup \{+\infty\}$ such that at each point of the set $U$ all partial derivatives are continuous. Then we say that $f$ is of the class $C^1$ on $U$. The set of all these functions is denoted by $C^1(U)$.

Analogously, a function $f : U \to \mathbb{R} \cup \{+\infty\}$ is said to be of the class $C^2$ on $U$ if it has continuous second partial derivatives at each point in the set $U$. The set of all such functions is denoted by $C^2(U)$.

**Definition 2.4.7.** Let $f : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ in $\Gamma_0(\mathbb{R}^p)$ such that $U := \operatorname{int}(\operatorname{dom}(f)) \neq \emptyset$. We say that $f$ is essentially smooth if it satisfies the following two conditions

1. $f$ is differentiable on $U$;
2. For every sequence $\{x_k\}_{k=1}^{+\infty} \subseteq U$ converging to a point in $\partial U := \overline{U} \setminus \operatorname{int}(U)$, we get that
$$\lim_{k \to +\infty} \|\nabla f(x_k)\| = +\infty.$$

In addition, $f$ is a Legendre type (or simply Legendre) function if it is essentially smooth and strictly convex on $U$.

For this part, we follow [11]. Now, we introduce the notion of a solution of an ordinary differential equation.

**Definition 2.4.8.** Let $U \subseteq \mathbb{R} \times \mathbb{R}^p$ be an open set. Let $f : U \to \mathbb{R}$ be a continuous function. Consider the (ordinary) differential equation:

$$\dot{x}(t) = f(t, x(t)). \tag{2.4.2}$$

A function $x : (t_-, t_+) \to \mathbb{R}^p$ of class $C^1$ (with $t_- < -1$ and $t_+ < +\infty$) is said to be a solution of (2.4.2) if:

1. $(t, x(t)) \in U$, for every $t \in (t_-, t_+)$;
2. $\dot{x}(t) = f(t, x(t))$, for every $t \in (t_-, t_+)$.

The ordinary differential equation (2.4.2) is said to be autonomous if $f$ does not depend on $t$ and $U \subseteq \mathbb{R}^p$, which is the setting that we consider from now on. Next we introduce the notion of initial value problem.

**Definition 2.4.9.** Given $(t_0, x_0) \in (t_-, t_+) \times U$, the initial value problem

$$\dot{x}(t) = f(x); \quad x(t_0) = x_0; \tag{2.4.3}$$

consists of finding an interval $(t_-, t_+)$ containing $t_0$ and a solution $x : (t_-, t_+) \to U$ of (2.4.2) such that $x(t_0) = x_0$. The condition $x(t_0) = x_0$ is called the initial condition of problem (2.4.3).

**Proposition 2.4.10.** Let $U \subseteq \mathbb{R}^p$ be an open set. Let $f : U \to \mathbb{R}$ be a continuous function. Given $(t_0, x_0) \in (t_-, t_+) \times U$, a continuous function $x : (t_-, t_+) \to \mathbb{R}^p$ in a open interval $(t_-, t_+)$ containing $t_0$ is a solution of the initial value problem (2.4.3) if and only if

$$x(t) = x_0 + \int_{t_0}^{t} f(x(s))ds;$$

for every $t \in (t_-, t_+)$.

**Proposition 2.4.11.** Let $: [t_0; + 1) \; ! \; \mathbb{R}$ with $(t_0) = {}_0$ be a continuously differentiable function, whose derivative satisfies

$$-(t) \quad u(t) \; (t);$$

for an integrable function $u : [t_0; + 1) \; ! \; ( \; 1 ; 0]$. Then we have

$$(t) \quad {}_0 \exp \int_{t_0}^{t} u(s)ds:$$

**Definition 2.4.12.** Let $U \quad \mathbb{R}^p$ be an open set. A function $f : U \; ! \; \mathbb{R}^d$ is said to be locally Lipschitz if for every compact set $K \quad U$ there exists $L > 0$ such that

$$(8 \; ; z \; 2 \; K) \quad kf( \; ) \quad f(z)k \quad Lk \quad zk:$$

**Theorem 2.4.13.** Let $U \quad \mathbb{R}^p$ be an open set. Let $f : U \; ! \; \mathbb{R}^p$. If $f$ is continuous and locally Lipschitz function in $U$, then for every $(t_0; \; {}_0) \; 2 \; \mathbb{R} \quad U$ there exists an open interval containing $t_0$ such that the solution of the initial value problem (2.4.3) is unique.

**Definition 2.4.14.** Let $U \quad \mathbb{R}^p$ be an open set. Let $f : U \; ! \; \mathbb{R}^p$ a continuous and locally Lipschitz in an open set $U \quad \mathbb{R}^p$. The maximal interval of a solution $: I \; ! \; \mathbb{R}^p$ of the equation $-(t) = f( \; )$ is the largest open interval $(t \; ; t_+)$ where there exists a solution.

**Definition 2.4.15.** If $= \; (t)$ is a solution of (2.4.2) with maximal interval $(t \; ; t_+)$, then the set $f \; (t) : t \; 2 \; (t \; ; t_+)g \quad U$ is called trajectory of the equation. This solution is said to be global if its maximal interval is $\mathbb{R}$.

**Proposition 2.4.16.** Any solution of equation (2.4.2) whose trajectory is contained in a compact subset of $U$ is global.

**Definition 2.4.17.** A point ${}_1 \; 2 \; U$ with $f( \; {}_1) = 0$ is called stationary point of (2.4.2).

**Theorem 2.4.18** (Cauchy, Lipschitz, Picard)**.** Let $f : \mathbb{R}^p \; ! \; \mathbb{R}$ be a Lipschitz continuous function in $\mathbb{R}^p$. Then for every ${}_0 \; 2 \; \mathbb{R}^p$ there exists a unique solution $: [0; + 1) \; ! \; \mathbb{R}^p$ in $C^1(\mathbb{R}^p)$ of the initial value problem (2.4.3).

The following definitions are collected from [77]. From now on, the map

$$2 \; \mathbb{R}^k \; 7! \quad = q( \; ) \; 2 \; \mathbb{R}^p;$$

is called reparameterization.

**Definition 2.4.19.** The Jacobian of $q = (q_1; : : : ; q_p)$ in $\quad 2 \; \mathbb{R}^k$ is denoted by $J_q( \; )$ and is defined as

$$J_q := \frac{@q}{@ \; 1}; : : : ; \frac{@q}{@ \; k} \quad := \begin{bmatrix} \frac{@q_1}{@ \; 1} & : : : & \frac{@q_1}{@ \; k} \\ \vdots & \ddots & \vdots \\ \frac{@q_p}{@ \; 1} & : : : & \frac{@q_p}{@ \; k} \end{bmatrix} :$$

**Definition 2.4.20.** A regular parameterization $q : \mathbb{R}^k \; ! \; \mathbb{R}^p$ is a $C^1$ parameterization such that $J_q( \; )$ is of rank $p$ for all $\quad 2 \; \mathbb{R}^p$.

**Definition 2.4.21.** A $C^2$ parameterization $q : \mathbb{R}^k \; ! \; \mathbb{R}^p$ is commuting if and only if for any $i; j \; 2 \; [p]$, we have that,

$$r^2 q_j( \; ) \; r \; q_i( \; ) \quad r^2 q_i( \; ) \; r \; q_j( \; ) = 0;$$

for all $\quad 2 \; \mathbb{R}^k$.

**Definition 2.4.22.** For any $C^1$ function $f : \mathbb{R}^k \to \mathbb{R} \cup \{+\infty\}$, we denote by $\varphi_f^t(x_0) = \gamma(t)$; where $\gamma(t)$ is the solution at time $t$ (when it exists) of

$$\begin{cases} \dot{\gamma}(t) = -\nabla f(\gamma(t)); & t > 0; \\ \gamma(0) = x_0 \in \mathbb{R}^k. \end{cases} \tag{2.4.4}$$

We say $\varphi_f^t(x_0)$ is well-defined at time $t$ when the above differential equation has a solution at time $t$.

**Definition 2.4.23.** Given a $C^2$ parameterization $q : \mathbb{R}^k \to \mathbb{R}^p$, for any $x_0 \in \mathbb{R}^p$ and $t \in \mathbb{R}^p$, we define

$$\Phi(x_0; t) := \varphi_{q_1}^{t_1} \circ \cdots \circ \varphi_{q_p}^{t_p}(x_0);$$

when it is well-defined, i.e., the corresponding differential equations have a solution. For any $x_0 \in \mathbb{R}^p$, we define the domain of $\Phi(x_0; \cdot)$ as:

$$U(x_0) = \{t \in \mathbb{R}^p \mid \Phi(x_0; t) \text{ is well defined}\}.$$

**Definition 2.4.24.** For any parameterization $q : \mathbb{R}^k \to \mathbb{R}^p$ in $C^2$ and for any function $L : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ in $C^1$, given any starting point $x_0 \in \mathbb{R}^p$, we define the reachable set $\mathcal{R}(x_0; q)$ as

$$\mathcal{R}(x_0; q) = \{\varphi_{L \circ q}^t(x_0) \mid t > 0\}.$$

For further results, the reader is referred to [3, 11, 15, 77].

# CHAPTER 3

# Regularization techniques

In this Chapter, we recall key ideas at the basis of our study, in particular with respect to the design regularization techniques.

## 3.1 Inverse problems

Many applied problems require estimating a function of interest from input/output data relation,

$$(x_i; y_i)_{i=1}^{d} \quad \mathbb{R}^p \quad \mathbb{R} \quad ?! \quad f : \mathbb{R}^p \; ! \; \mathbb{R}:$$

In this chapter, we are interested in the case when $f$ is a linear function, expressed as

$$f : \mathbb{R}^p \; ! \; \mathbb{R}\,; \; x \; ?! \; h \quad j\, x i;$$

for some $2\ \mathbb{R}^p$: Then, the problem simplifies to determining from the following linear equation

$$X \quad = y;$$

where $X$ represents a matrix with rows corresponding to input data points, and $y$ can be seen as the vector of measurements of the vector that we want to recover.

Observe that, if $y \ 2\ \mathrm{ran}(X)$, the previous problem does not have a solution. For this reason, it is necessary to introduce a loss function $L : \mathbb{R}^p \; ! \; \mathbb{R}\ [\ f+\ 1\ g$ such that it ensures the existence of at least one solution, for example, least squares. In general, this function is defined via a function $\grave{}\,: \mathbb{R} \quad \mathbb{R} \; ! \; \mathbb{R}$, through the following equation:

$$L(\ ) = \sum_{i=1}^{n} \grave{}(hx_i \, j \quad i; y_i):$$

On the other hand, the solution of the linear system may be not unique and a selection criterion is needed to choose a suitable solution. To deal with non-uniqueness and select a particular solution, a regularization term $R : \mathbb{R}^p \; ! \; \mathbb{R}\ [\ f+\ 1\ g$ is introduced. This regularization includes the prior knowledge on the solution in the model, and enforces a bias on the solution. Summarizing, a standard approach to recover is to assume that it is a minimizer of the following linearly constrained optimization problem:

$$\min_{2\mathbb{R}^p} fR(\ ) : \quad 2\ S = \operatorname{argmin} Lg; \tag{P}$$

The above ideas can be justified from different perspectives. For instance, compressed sensing [32, 34, 48, 115, 129], image processing [37, 39, 103, 105, 116, 117, 140], and various problems in machine learning [13, 50, 90, 114, 121, 140, 141].

**Example 3.1.1.** In this example, we show the most common losses used both in regression and classification. In classification tasks, loss functions are typically based on the concept of margin, i.e., $\hat{y}\langle h, x_i\rangle$, while in regression, they are often based on the difference $\hat{y} - \langle h, x_i\rangle$. Some well-known examples include:

- **Least Squares:** It is defined as $L(h) = \frac{1}{d}\sum_{i=1}^{d}(\langle h, x_i\rangle - y_i)^2$. This loss function measures the average squared difference between the predicted values and the actual values.
- **Huber Loss:** A piece-wise function that combines squared error and absolute error, defined as $L(h) = \frac{1}{d}\sum_{i=1}^{d}\ell(\langle h, x_i\rangle - y_i)$, where $\ell$ is defined as follow

$$\ell(t) = \begin{cases} \frac{1}{2}t^2 & \text{for } |t| \leq \frac{1}{2}; \\ |t| - \frac{1}{2} & \text{otherwise.} \end{cases}$$

- **Exact Penalization:** It is defined as

$$L(h) = \begin{cases} 0 & \text{if } X = y; \\ +1 & \text{otherwise.} \end{cases}$$

  This loss is the one we will use throughout this thesis.
- **Hinge Loss:** Typically used in Support Vector Machines (SVMs) for classification, it is defined as $L(h) = \frac{1}{d}\sum_{i=1}^{d}\max(0, 1 - y_i\langle h, x_i\rangle)$. Here, for every $i \in [d]$, $y_i \in \{-1, 1\}$.
- **Exponential Loss:** It is defined as $L(h) = \frac{1}{d}\sum_{i=1}^{d} e^{-y_i\langle h, x_i\rangle}$. Here, for every $i \in [d]$, $y_i \in \{-1, 1\}$.
- **Logistic Loss:** This is widely used in logistic regression for binary classification. It can be expressed as $L(h) = \frac{1}{d}\sum_{i=1}^{d}\log\left(1 + e^{-y_i\langle h, x_i\rangle}\right)$, where, for every $i \in [d]$, $y_i \in \{-1, 1\}$.

**Example 3.1.2.** We illustrate some examples of regularizers presented in the state of the art [32, 34, 52, 105, 116, 117, 148].

- **Squared norm**: It is defined as $R(h) = \|h\|_2^2$. This regularizer leads to the minimum norm solution that is the Moore-Penrose pseudoinverse $X^{\dagger}y$ [52, Theorem 2.5], when the loss is the least square.
- **$\ell_0$-norm**: It is defined as $R(h) = \|h\|_0$, where $\|h\|_0$ is the function that counts the non-zero elements of the vector.
- **$\ell_1$-norm**: It is defined as $R(h) = \|h\|_1$. This function is a convex surrogate for the $\ell_0$-norm, and promotes sparsity of the solution.
- **Nuclear norm**: It is defined as $R(h) = \|h\|_*$, where $h \in \mathbb{R}^{p \times p}$ is a square matrix. This function is a convex surrogate for the rank of a matrix, enhancing row/column correlation [28]. This function will be used in the numerical examples in Section 4.5.
- **Group lasso norm**: It is defined as $R(h) = \|h\|_J$, where $J = \{J_1, \dots, J_l\}$ is a partition of $[p]$.

- **Total variation**: It is defined as $R(\ ) = kD\ \ k_{1;2}$; where the $k\ \ k_{1;2}$ and $D$ denote the $`_{1;2}$ - norm and $D$ is the discrete gradient operator, respectively (see Chapter 2). This regularizer is usually used in image reconstruction (deblurring + denoising) since it maintains sharp edges between different areas with constant color (see for example [23, 37, 39, 87, 103, 105, 116, 117, 140]).

  This function will be used in the numerical examples in Section 4.5.

## 3.2 Tikhonov regularization, flows, and algorithms

A classical way to solve ($P$) is to relax the constraint, and use **Tikhonov regularization** [19, 52, 128]:

$$\min_{2R^p} \frac{1}{}L(\ ) + R(\ ): \qquad (P\ )$$

Note that the loss is explicitly added to the objective function and is multiplied by a regularization parameter $\ :$ The minimizers of the above problem define a sequence $f\ \ g_{\ >0}$, of possible solutions, called regularization path [51, 63]. Among these solutions, the best regularized solution is selected according to some criteria, such as the Morozov discrepancy principle in inverse problems [52] or cross-validation on left-out data in machine learning [57, 126]. Typically, this involves fixing a grid $\ $ within the interval $[\ _{\min};\ _{\max}]$ and then solving ($P\ $) for each $\ 2\ $, where the best regularized solution is selected according to an appropriate data-driven criterion.

**Example 3.2.1** (Ridge regression)**.** The most classic example is when $L$ is the least square, and $R$ is the square norm. Then, problem ($P\ $) becomes

$$\min_{2R^p} \frac{1}{}kX\ \ \ \ yk^2 + k\ \ k^2:$$

The above problem is also called ridge regression [57, 65, 128]. It is a classic fact that the sequence of solutions $f\ \ :=\ \ \mathrm{Id} + X^{>}X\ \ ^{1}X^{>}yg_{\ >0}$, converges to a minimum norm solution [52, Theorem 5.2], when $\ !\ 0$.

In the following section we present flows and algorithms that can be applied to solve ($P\ $). From the above discussion, it is clear that optimization to solve ($P\ $) (and also ($P$)) plays a crucial role in the solution of machine learning and inverse problems.

### 3.2.1 $R\ \ 0$: Continuous case

Note that when $R\ \ 0$, the problems described in equations ($P\ $) and ($P$) simplify to solve:

$$\min_{2R^p} L(\ );$$

Additionally, if we assume that $L : R^p\ !\ R$ is a convex and differentiable function such that $r\,L$ is Lipschitz continuous on $R^p$, a good approach to solving the previous problem is to consider the steepest descent trajectory. The gradient flow defines a trajectory for each point $_0$ in the space, which can be described by the following ordinary differential equation:

$$\begin{aligned} \dot{}(t) &= \ \ r\,L(\ (t)); \quad t > 0 \\ (0) &= \ \ _0: \end{aligned} \qquad (GF)$$

It follows from Theorem 2.4.18 that the previous flow, for every $_0\ \ 2\ \ R^p$, has unique solution.

An extension of gradient flow to non-Euclidean geometries is mirror flow [3], which can be modeled as the following ordinary differential equation:

$$r^2F(\ (t))\ \dot{-}(t) = \quad r\,L(\ (t)); \tag{MF}$$

where $F\colon \mathrm{dom}(F) \ ! \ \mathsf{R}$ is the mirror map and is strictly convex, Legendre and twice-differentiable in $U = \mathrm{int}(\mathrm{dom}(F))$, and the mapping $\ 2\ U\ 7!\ [r^2F(\ )]$ is invertible with locally Lipschitz inverse. In the case when the mirror map is defined as the Euclidean norm, i.e., $F(\ ) = \frac{k\ k^2}{2}$, the equation describing the mirror flow becomes equivalent to the gradient flow.

Observe that, for every $z\ 2\ S$;

$$\frac{d(D_F(z;\ (t)))}{dt} = \hbar\, r\,L(\ (t))\, j\ \ (t)\quad z i\quad L\quad\ L(\ (t))\quad 0; \tag{3.2.1}$$

where $L\ = \min_{2\mathsf{R}^p} L(\ )$: Since the Bregman distance is decreasing and bounded below we get that, for every $z\ 2\ S$; $D_F(z;\ (t))$ converges. Moreover, computing the derivative of $L(\ (t))$, we get that

$$\frac{d(L(\ (t))\quad L\ )}{dt} = \quad \overset{\mathsf{D}}{\quad r^2F(\ )\quad ^1}\ r\,L(\ (t))\, j\ r\,L(\ (t))\overset{\mathsf{E}}{\quad}\quad 0;$$

Which implies that $(L(\ (t))\quad L\ )$ is decreasing with respect to $t$. Since $D_F(\ ;\ )$ is non-negative, we can integrate (3.2.1) and obtain that

$$t\,(L(\ (t))\quad L\ )\quad \int_0^t (L(\ (s))\quad L\ )\,ds\quad D_F(z;\ _0);$$

which implies convergence in value.

Note that, if $F(\ ) = \frac{k\ k^2}{2}$, we get that $k\ \ (t)\quad zk$ converges for every $z\ 2\ S$; which is the first condition of Opial's Lemma. While the second condition, each limit point belongs to the solution set, is obtained by convergence in value. Then, by using Lemma 2.1.5, we conclude that $\ (t)$ converges to some $\quad 2\ S$:

### 3.2.2 $R\quad 0$: Discrete case

If we discretize $\dot{-}(t)$ with $t\ 2\ [k;k+1]$, using finite differences with a positive step-size $\ 2]0;\frac{2}{L}[$; where $L$ is the Lipschitz constant of $r\,L$, then

$$\dot{-}(t)\quad \frac{^{k+1}\qquad ^k}{\qquad\qquad}:$$

**Gradient descent** [35]: If we take $r\,L(\ (t))\qquad r\,L(\ ^k)$; then the algorithm has the following explicit update rule

$$^{k+1} = \ ^k\qquad r\,L(\ ^k): \tag{3.2.2}$$

In the following theorem, we establish sufficient conditions for the convergence of gradient descent in $\mathsf{R}^p$.

**Theorem 3.2.2.** Let $L : \mathbb{R}^p \to \mathbb{R}$ be a convex Gâteaux differentiable function, such that $S \neq \emptyset$. Suppose that $\nabla L$ is $L$-Lipschitz continuous on $\mathbb{R}^p$. Let $(x_k)_{k\in\mathbb{N}}$ be the sequence generated by (3.2.2) starting from some arbitrary $x_0 \in \mathbb{R}^p$ and with step-size $\lambda \in ]0; \frac{2}{L}[$. Then,

$$\lim_{k\to+\infty} x_k = x \in S.$$

**Example 3.2.3** (Gradient descent on the least squares)**.** Let $X$ be the matrix with rows corresponding the input data and let $y$ be the vector of measurements. In this case the iterates of gradient descent are given by

$$x_{k+1} = x_k - \lambda X^\top (X x_k - y).$$

The previous method is also known as the Landweber method [52, 72]. In Chapter 4 we use a single iteration of this algorithm with different step-sizes to improve the feasibility of iterations.

If we assume that $X$ is full rank and we continue assuming that $L$ is the least square, we can use a well-known stochastic algorithm.

**Randomized Kaczmarz**: Let $x_0 \in \mathbb{R}^p$ and consider the following algorithm:

$$x_{k+1} = x_k - \frac{\langle x_k, x_t\rangle - y_t}{\|x_t\|^2} x_t,$$

where $(x_k)_{k\in\mathbb{N}}$ is a sequence of independent $[d]$-valued random variables. It is well-known and has been demonstrated that the sequence $(x_k)_{k\in\mathbb{N}}$ converges to a feasible point (see [41, 127]). For further information, refer to [36, 41, 62, 70, 73, 95–98, 127, 150]. In Chapter 4 we use a single iteration of this algorithm to improve the feasibility of iterations.

In the case when $L$ is not necessary $L$-smooth and $\operatorname{prox}_L$ is easy to compute, we can use **proximal point algorithm** [81], which has the following implicit update rule

$$x_{k+1} = \operatorname{prox}_{\lambda L}(x_k). \tag{3.2.3}$$

Note that this method can be applied, unlike gradient descent, to non-differentiable functions. However, since it is an implicit rule, in most cases the prox cannot be easily calculated. In the following theorem, we provide the conditions for the convergence of proximal point algorithm in $\mathbb{R}^p$.

**Theorem 3.2.4.** Let $L : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ be a proper convex lower semi-continuous function such that $S \neq \emptyset$. Let $(x_k)_{k\in\mathbb{N}}$ be the sequence generated by (3.2.3) starting from some arbitrary $x_0 \in \mathbb{R}^p$ and with step-size $\lambda > 0$. Then,

$$\lim_{k\to+\infty} x_k = x \in S.$$

### 3.2.3 Case $R \neq 0$

If $R$ and $L$ are both $L$-smooth, then we can apply gradient descent, which has the following update rule

$$x_{k+1} = x_k - \lambda \nabla L(x_k) - \lambda \nabla R(x_k), \tag{3.2.4}$$

where $\lambda > 0$ is the regularization parameter. In Theorem 3.2.2 are provided conditions to ensure the convergence of (3.2.4). In the case when $R$ is non-smooth and $\operatorname{prox}_R$ is easy to compute, we can use forward-backward algorithm, which is

$$x_{k+1} = \operatorname{prox}_{\lambda R}(x_k - \lambda \nabla L(x_k)).$$

In the following theorem, we provide the conditions for the convergence of forward-backward in $\mathbb{R}^p$.

**Theorem 3.2.5.** Let $R : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ and $L : \mathbb{R}^p \to \mathbb{R}$ be two functions in $\Gamma_0(\mathbb{R}^p)$ such that $\operatorname{argmin}_{x \in \mathbb{R}^p}(L(x) + R(x)) \neq \emptyset$. Suppose that $L$ is Gâteaux differentiable such that $\nabla L$ is $L$-Lipschitz continuous on $\mathbb{R}^p$. Let $(x^k)_{k \in \mathbb{N}}$ be the sequence generated by (3.2.2) starting from some arbitrary $x^0 \in \mathbb{R}^p$ and with step-size $\gamma \in ]0, \frac{2}{L}[$. Then,

$$\lim_{k \to +\infty} x^k = x \in \operatorname{argmin}_{x \in \mathbb{R}^p}(L(x) + R(x)).$$

For all of the methods presented above, selecting an appropriate $\lambda$ can be challenging and typically requires solving the optimization problem multiple times with different parameter values.

## 3.3 Iterative regularization

In this section, we explore a more efficient alternative known as iterative regularization. Unlike Tikhonov regularization, this technique involves running a single optimization procedure that stops before convergence. Typically, since a single problem is solved and the algorithm is not run even until convergence, iterative regularization is numerically more efficient compared to Tikhonov.

In practice, the exact data $y$ is unknown, and only a noisy version is accessible. Given a noise level $\delta > 0$, we consider a worst-case scenario where the error is deterministic and the accessible data $y^\delta$ is such that

$$\|y^\delta - y\| \leq \delta;$$

which limits us to solve only a noisy version of ($P$).

Iterative regularization consists of finding an approximation of $x$ by running an iterative algorithm that generates a regularizing sequence $(x^k)_{k \in \mathbb{N}}$ and stopping when it is closed to the solution according to certain criteria. The algorithm must consider the properties of $R$, $L$, and $X$, and for any data $y^\delta \in \mathbb{R}^d$, the algorithm generates a sequence. To analyze the behavior of the sequence $(x^k)_{k \in \mathbb{N}}$, it is necessary to define the auxiliary sequence $(x^k)_{k \in \mathbb{N}}$, which is obtained by applying the same algorithm but using $y$ instead of $y^\delta$. We additionally assume that the sequence converges to $x$, the solution of ($P$). Then, we can decompose the error as

$$\|x^k - x\| \leq \|x^k - x^k\| + \|x^k - x\| \leq \|x^k - x^k\| + \|x^k - x\|. \tag{3.3.1}$$

The first term corresponds to an optimization error, which decreases with respect to $k$. While the second error measures the stability of noise, which increases with respect to $k$, since the iterations converge to a noisy solution or diverges. An early stopping strategy takes in account the different behaviours of the two terms in equation (3.3.1) and stops the iterations when these terms are approximately equal. In this way, the number of iterations plays the role of the regularization parameter in the same way as $\lambda$ in Tikhonov regularization.

For the next two algorithms, we consider a simpler noisy version of ($P$),

$$\min_{Ax = y^\delta} R(x); \tag{3.3.2}$$

which is studied in the Chapter 4. The above problem is obtained by choosing $L$ equal to least squares and taking $y$ instead of $y$: Note that (3.3.2) may not be feasible. We also define the noise free problem as

$$\min_{X = y} R( ): \tag{3.3.3}$$

We now present two algorithms: the first for strongly convex regularizers and the second for regularizers that are only convex. Additionally, both algorithms incorporate early stopping strategies.

**Dual Gradient descent**: Let $R : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ be proper, lower semicontinuous, and - strongly convex function. Let $w^0 = 0$ and $= \frac{}{\|X\|^2}$. Then, the iterations of dual gradient descent are:

$$^{k+1} = \text{prox}_{\underline{R}} \quad \frac{X^{>} w^k}{} \quad ;$$

$$w^{k+1} = w^k + \quad X \quad ^{k+1} \quad y \quad ;$$

$$p^{k+1} = \frac{1}{k+1} \sum_{i=1}^{k} {}^{i} :$$

Now we present an early stopping result for the previous algorithm.

**Theorem 3.3.1.** [82, Theorem 4.1] Let $2]0; 1]$. Assume that there exists $w \ 2 \ \mathbb{R}^p$ such that $X^{>} w \ 2 \ @R( \ )$. Set $c_1 = 2\|X\| \ ^1$ and $c_2 = \frac{\|X\| \|w^y\|^1}{}$, where $w^y$ is a solution of the dual problem of (3.3.3). Then, for every $k \ 2 \ \mathbb{N}$,

$$\|p^k \quad w^y\| \quad c_1 \frac{p}{k} \ + \frac{c_2}{p \overline{k}} : \tag{3.3.4}$$

In particular, choosing $\bar{k} \quad \frac{c}{}$ for some $c > 0$, we derive

$$\|p^{\bar{k}} \quad w^y\| \quad c_1(c^{\frac{1}{2}} + 1) + \frac{c_2}{c^{\frac{1}{2}}} \quad ^{\frac{1}{2}} :$$

The previous result gives us a decomposition of the error into optimization and stability errors as in (3.3.1) in (3.3.4). In some cases the regularizer is neither smooth nor strongly convex. In this situation it is suitable to use the dual primal-dual algorithm.

**Primal-dual**: Let $R : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ be a function in $_0(\mathbb{R}^p)$. Let $w^0 = w^{\ 1} \ 2 \ \mathbb{R}^d$, $^0 \ 2 \ \mathbb{R}^p$ and $> 0$ and $> 0$ such that $\|X\|^2 < 1$. Then, the iterations of primal dual are:

$$^{k+1} = \text{prox}_{\ R} \quad ^k \quad X^{>} \quad 2w^k \quad w^{k \ 1} \quad ;$$

$$w^{k+1} = w^k + \quad X \quad ^{k+1} \quad y \quad ;$$

$$p^{k+1} = \frac{1}{k+1} \sum_{i=1}^{k} {}^{i} :$$

Note that the above algorithm is a specific case of the primal-dual algorithm proposed in [40], where the function in the primal is $R$, the function in the dual is the indicator $f y \ g$, and the linear operator is $X$:

**Theorem 3.3.2.** [85, Proposition 7] Let $R : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ be a proper, convex, and lower semicontinuous function. Assume that the problem (3.3.3) has at least one solution. Let $(\quad ; w) \in \mathbb{R}^p \quad \mathbb{R}^d$, where $\quad$ is solution of (3.3.3) and $w$ is a solution of the dual problem of (3.3.3). Then, there exist strictly positive constants $c_1$ and $c_2$ such that, for every $k \in \mathbb{N}$,

$$\|X p^k \quad y\| \quad c_1 \quad + \quad ^2 k + \frac{1}{k} \tag{3.3.5}$$

and

$$R(p^k) + \quad \overset{D}{w} \; j \, X p^k \quad \overset{E}{y} \quad R(\quad) \quad \frac{c_2}{k}(1 + \quad k)^2 : \tag{3.3.6}$$

Moreover, if we choose $\bar{k} \quad \underline{c}$ for some $c > 0$, then there exist constants $c_3$ and $c_4$ such that

$$\|X p^{\bar{k}} \quad y\| \quad c_3 \quad + \quad ^2$$

and

$$R(p^{\bar{k}}) + \quad \overset{D}{w} \; j \, X p^{\bar{k}} \quad \overset{E}{y} \quad R(\quad) \quad c_4 :$$

Unlike (3.3.1), the previous result gives us a decomposition of the error into optimization and stability errors, but in this case applied to the feasibility gap in (3.3.5) and the duality gap in (3.3.6).

In Chapter 4, we introduced two algorithms inspired by the primal-dual approach with activation discussed in [24]. This method has shown significant numerical speed up in noise-free scenarios by improving the feasibility of the iterations. We combine this algorithm with the early stopping strategy presented in [84].

# CHAPTER 4

# Fast iterative regularization by reusing data

**Abstract**

Discrete inverse problems correspond to solving a system of equations in a stable way with respect to noise in the data. A typical approach to selecting a meaningful solution is to introduce a regularizer. While for most applications the regularizer is convex, in many cases it is neither smooth nor strongly convex. In this chapter, we propose and study two new iterative regularization methods, based on a primal-dual algorithm, to regularize inverse problems efficiently. Our analysis, in the noise-free case, provides convergence rates for the Lagrangian and the feasibility gap. In the noisy case, it provides stability bounds and early stopping rules with theoretical guarantees. The main novelty of our work is the exploitation of some a priori knowledge about the solution set: we show that the linear equations determined by the data can be used more than once along the iterations. We discuss various approaches to reusing linear equations that are at the same time consistent with our assumptions and flexible in their implementation. Finally, we illustrate our theoretical findings with numerical simulations for robust sparse recovery and image reconstruction. We confirm the efficiency of the proposed regularization approaches by comparing the results with state-of-the-art methods.

**Keywords.** Primal-dual splitting algorithms, Iterative regularization, Early stopping, Landweber method, Stability and convergence analysis.

**AMS Mathematics Subject Classification (2020):** 90C25, 65K10, 49M29.

## 4.1 Introduction

Many applied problems require the estimation of a quantity of interest from noisy linear measurements, for instance compressed sensing [32, 34, 48, 115, 129], image processing [37, 39, 103, 105, 116, 117, 140], matrix completion [28, 31, 33, 84], and various problems in machine learning [13, 50, 90, 114, 121, 140, 141]. In all these problems, we are interested in finding stable solutions to a system of equations where the accessible data is corrupted by noise. This is classically achieved by regularization. The most popular procedure in the literature is Tikhonov (or variational) regularization [52], which consists in minimizing the sum of a data fidelity term plus a regularizer, which is explicitly added to the objective function and entails some a priori knowledge or some desired property on the solutions that we want to select. A trade-off parameter is then introduced to balance the fidelity term and the regularizer. In practice, this implies that the optimization problem has to be solved many times for different values of the parameter. Finally, a parameter - and the correspondent solution - are chosen accordingly to the performance with respect to some criterion, such as the Morozov discrepancy principle in inverse problems [52] or cross-validation on left-out data in machine learning [57, 126].

A computationally efficient alternative to explicit regularization is iterative regularization, also known as implicit regularization [8, 21, 27, 52]. The minimization of the regularizer under noisy data constraints is considered, and a numerical algorithm to solve the optimization problem is chosen and early stopped, to avoid convergence to the noisy solution. In this setting, it is known that the number of iterations plays the role of the regularization parameter [52]. As for Tikhonov regularization, the best-performing iterate is chosen according to some a priori criterion and then considered as the regularized solution. Compared to Tikhonov regularization, this procedure is very efficient since only one optimization problem is solved, and not even until convergence.

The main novelty of this work is the design and analysis of two new iterative regularization methods for convex regularizers, which are neither necessarily smooth nor strongly convex. The new iterative regularization methods are based on primal-dual algorithms [40, 46, 133] combined with the idea of reusing the linear equations determined by the data at every iteration [24]. Primal-dual algorithms perform one minimization step on the primal variable followed by one on the dual and are well-suited for the large-scale setting, as only matrix-vector multiplications and the calculation of a proximity operator are required. The idea of exploiting redundant information was presented in [24] and turned out to be very effective in practice. The first method that we propose is a primal-dual algorithm (PDA) with additional activations of the linear equations: We propose different variants, depending on the extra activation steps. For instance, we are able to exploit the data constraints more than once at every iteration via gradient descent, with a fixed or adaptive step size. The second method is a dual-primal algorithm (DPA), where a subset containing the dual solutions is activated at each step. This subset is not affected by the noise in the data and is usually determined by a finite number of constraints.

These additional steps may seem artificial or inefficient. However, while maintaining easy implementation, our methods achieve better numerical performances and considerable speed-ups with respect to the vanilla primal-dual algorithm.We extend to the noisy case the techniques studied in [24, 25] for the exact case. The assumptions on the noise are the classical ones in inverse problems, see e.g. [27, 30, 82, 84], and the proposed results generalize the ones in [84], by including in the primal-dual procedure a diagonal preconditioning and an extra activation step. For the noisy case, we provide an early stopping

criterion to recover a stable approximation of an ideal solution, in the same spirit of [12, 20, 27, 30, 82, 112, 141, 146]. The early stopping rule is derived from theoretical stability bounds and feasibility gap rates for both algorithms, obtaining implicit regularization properties similar to those stated in [84] and [82]. Theoretical results are complemented by numerical experiments for robust sparse recovery and total variation, showing that state-of-the-art performances can be achieved with considerable computational speed-ups.

**Related works.** In this section, we briefly discuss the literature about variational and iterative regularization techniques.Tikhonov regularization has been introduced in [128]; see also [19, 52] and the references therein for an extensive treatment of the topic.The most famous iterative regularization method is the Landweber algorithm [52, 72], namely gradient descent on the least squares problem. Duality theory in optimization gives another interpretation that sheds light on the regularizing properties of this procedure. Indeed, consider the problem of minimizing the squared norm under linear constraints. Running gradient descent on its dual problem and mapping back to the primal variable, we obtain exactly the Landweber method. This provides another explanation of why the iterations of the Landweber algorithm converge to the minimal norm solution of the linear equation [82]. Stochastic gradient descent on the previous problem is the generalization of the Kaczmarz method [70, 78, 120, 127], which consists in applying cyclic or random projections onto single equations of the linear system. Accelerated and diagonal versions are also discussed in [52, 101] and [10, 71, 119], respectively. The regularization properties of other optimization algorithms for more general regularizers have also been studied. If strong convexity is assumed, mirror descent [16, 100] can also be interpreted as gradient descent on the dual problem, and its regularization properties (and those of its accelerated variant) have been studied in [82]. Diagonal approaches [9] with a regularization parameter that vanishes along the iterations have been studied in [54]; see [30] for an accelerated version. Another common approach relies on the linearized Bregman iteration [103, 140, 142, 143], which has found applications in compressed sensing [29, 104, 143] and image deblurring [29]. However, this method requires to solve non-trivial minimization problems at each iteration. For convex, but not strongly convex regularizers, the regularization properties of primal-dual algorithms have been investigated in [84]. Other optimization techniques are available to solve this kind of minimization problem (for instance, [86, 87] and [22, 83, 110]; see also [66, 123, 124]), but no iterative regularization properties have been studied so far for these algorithms.

## 4.2 Main problem and algorithm

Many applied problems require to estimate a quantity of interest $2 \mathrm{R}^p$ based on linear measurements $y = X$ , for some matrix $X 2 \mathrm{R}^{d\ p}$. A standard approach to recover the desired solution is to assume that it is a minimizer of the following linearly constrained optimization problem:

$$\min_{2\mathrm{R}^p} fR( ) : X = yg; \qquad (P)$$

where $R 2 {}_0(\mathrm{R}^p)$ encodes a priori information on the solution and is usually handcrafted. Typical choices are: the squared norm [52]; the elastic net regularization [47, 69, 74, 82, 148, 149]; the $`^1$-norm [32, 34, 48, 129]; and the total variation [37, 105, 116, 117]. Note that, in the previous examples, the first two regularizers are strongly convex, while the second two are just convex and non-smooth.

If we use the indicator function of $\{y\}$, the problem $(P)$can be written equivalently as

$$\min_{x \in \mathbb{R}^p} R(x) + \iota_{\{y\}}(Ax).$$

We denote by $\vartheta$ the optimal value of $(P)$ and by $S$ the set of its minimizers. We assume that $S \neq \emptyset$. In order to build our regularization procedure, we consider the Lagrangian functional for problem $(P)$:

$$\begin{aligned} L : \mathbb{R}^p \times \mathbb{R}^d &\to \mathbb{R} \cup \{+\infty\} \\ (x, w) &\mapsto R(x) + \langle w \mid Ax - y \rangle. \end{aligned}$$

This approach allows us to split the contribution of the non-smooth term $R$ and the one of the linear operator $R$, without requiring to compute the projection on the set

$$S := \{x \in \mathbb{R}^p \mid Ax = y\}.$$

We define the set of saddle points of $L$ by

$$Z = \left\{ (x, w) \in \mathbb{R}^p \times \mathbb{R}^d : L(x, v) \leq L(x, w) \leq L(y, w) \ \forall (y, v) \in \mathbb{R}^p \times \mathbb{R}^d \right\}.$$

The set $Z$ is characterized by the first-order optimality condition:

$$Z = \left\{ (x, w) \in \mathbb{R}^p \times \mathbb{R}^d : 0 \in \partial R(x) + A^\top w \text{ and } Ax = y \right\}.$$

In the following, we always assume that $Z \neq \emptyset$.

**Remark 4.2.1** (Saddle points and primal-dual solutions)**.** Observe that the objective function of $(P)$is the sum of two functions in $\Gamma_0(\mathbb{R}^p)$ where one of the two is composed with a linear operator. This formulation is suitable to apply Fenchel-Rockafellar duality. Recalling that $\iota_{\{y\}}(w) = \langle u \mid y \rangle$, the dual problem of $(P)$is given by

$$\min_{w \in \mathbb{R}^d} R^*(-A^\top w) + \langle w \mid y \rangle. \tag{D}$$

We denote its optimal value by $\vartheta^*$ and its set of minimizers by $S^*$. Then, $Z = S \times S^*$, and equality holds if the qualification condition (see [15, Proposition 6.19] for special cases when it holds)

$$y \in \operatorname{ri}(A(\operatorname{dom} R)) \tag{4.2.1}$$

is satisfied [15, Proposition 19.21 (v)]. In addition, condition (4.2.1) implies that problem $(D)$ has a solution. Then, under (4.2.1), since we assumed that $S \neq \emptyset$, we derive also that $Z \neq \emptyset$.

In practical situations, the exact data $y$ is unknown and only a noisy version is accessible. Given a noise level $\delta > 0$, we consider a worst case scenario, where the error is deterministic and the accessible data $y^\delta$ is such that

$$\|y - y^\delta\| \leq \delta.$$

This is the classical model in inverse problems [52, 71]. The solution set of the inexact linear system $Ax = y^\delta$ is denoted by $S^\delta$. Analogously, we denote by $S^\delta \times S^*$ and $S \times S^*$ the sets of primal and dual solutions with noisy data, respectively. It is worth pointing out that, if $y^\delta \notin \operatorname{ran}(A)$, then $S^\delta = \emptyset$ but our analysis and bounds still hold.

### 4.2.1 Primal-Dual Splittings with a priori Information

In this section, we propose an iterative regularization procedure to solve problem ($P$), based on a primal-dual algorithm with preconditioning and arbitrary activations of a predefined set of operators. While the use of primal-dual algorithms [40] as iterative regularization methods is somewhat established [84], in this chapter we focus on the possibility of reusing the data constraints along the iterations. This idea was originally introduced in [24], where the authors studied the case in which the exact data is available, and consists in the activation of extra operators, which encode information about the solution set, to improve the feasibility of the updates. In our setting, we reuse data constraints, and we project, in series or in parallel, onto some equations given by the (noisy) linear constraints. But we will show that other interesting choices are possible, as projections onto the set of dual constraints.

More formally, for $i \in [m]$, we consider a finite number of operators $T_i : \mathbb{R}^p \to \mathbb{R}^p$ or $T_i : \mathbb{R}^d \to \mathbb{R}^d$, such that the set of noisy primal (or dual) solutions is contained in $\mathrm{Fix}\, T_i$ for every $i \in [m]$. We refer to this as redundant a priori information. A list of operators suitable to our setting (and with a cheap practical implementation) can be found in Section 4.4.

The primal-dual algorithms with reuse of data which are given in Table 4.1 are a preconditioned and deterministic version of the one proposed in [24] applied to the case of linearly constrained minimization.

| Primal-Dual splitting with activations | Dual-Primal splitting with activations |
|---|---|
| **Input**: $(p^0; \ ^0; w^0) \in \mathbb{R}^{2p} \quad \mathbb{R}^d$. | **Input**: $(v^0; w^0; \ ^0) \in \mathbb{R}^{2d} \quad \mathbb{R}^p$. |
| **For** $k = 1; \dots; N$: | **For** $k = 1; \dots; N$: |
| $w^{k+1} = w^k + \ (Xp^k \quad y)$ | $^{k+1} = \mathrm{prox}_R(^k \quad X^\top v^k)$ |
| $^{k+1} = \mathrm{prox}_R(p^k \quad X^\top w^{k+1})$ | $w^{k+1} = v^k + \ (X \ ^{k+1} \quad y)$ |
| Choose $_{k+1} \in [m]$ and set (PDA) | Choose $_{k+1} \in [m]$ and set (DPA) |
| $p^{k+1} = T_{\ ^{k+1}} \ ^{k+1}$ | $v^{k+1} = T_{\ ^{k+1}} w^{k+1}$ |
| $p^{k+1} = p^{k+1} + \ ^{k+1} \quad p^k;$ | $v^{k+1} = v^{k+1} + w^{k+1} \quad v^k;$ |
| **End** | **End** |

Table 4.1: Proposed algorithms for iterative regularization.

We first focus on the Primal-Dual splitting. It is composed by four different steps, to be performed in series. The first step is the update of the dual variable, in which the residuals to the linear equation $X \quad = y$ are accumulated after preconditioning by the operator . The second step is an implicit prox-step, with function $R$ and norm $k \quad k_1$, on the primal variable. The third one is the activation of the operator related to reusing data constraints, on the primal variable. Finally, the last step is an extrapolation again on the primal variable. Notice that, if no operator is activated, it corresponds simply to $p^{k+1} = 2 \ ^{k+1} \quad ^k$, which is the classical update in the primal-dual algorithm. Observe that, the Dual-Primal Splitting algorithm, except for permutation in the order of the steps, differs from the previous one because the activation of the operator is done not on the primal variable but on the dual one.

**Remark 4.2.2.** Observe that in the proof convergence and stability (Theorem 4.3.1 and Theorem 4.3.2) we will never used that belongs to a finite dimensional space. This is in line with previous research on the convergence guarantees of the plain methods in Hilbert and Banach spaces, as outlined in [46, 122, 133]. It follows that the primal-dual

algorithms above can be formulated exactly in the same way when the unknown vector $x$ belongs to an infinite dimensional Hilbert space, and our analysis can be extended to that setting. Another possible extension of the algorithm, which we do not analyze explicitly in this work, is related with the stochastic version of primal-dual algorithm; see [1, 38, 61].

### 4.2.2 Equivalence between Primal-dual and Dual-primal algorithms.

The next lemma establishes that, if $T = \mathrm{Id}$ and the initialization is the same, then there is an equivalence between the $k$-th primal variable generated by PDA and the ones generated by DPA.

**Lemma 4.2.3.** Let

$$(p^0_{PD};\ {}^0_{PD}; w^0_{PD}) \in \mathbb{R}^{2p} \quad \mathbb{R}^d \text{ and } (v^0_{DP}; w^0_{DP};\ {}^0_{DP}) \in \mathbb{R}^{2d} \quad \mathbb{R}^p$$

be the initialization PDA and DPA, respectively, in the case when $m = 1$ and $T = \mathrm{Id}$. Suppose that

$${}^0_{PD} = p^0_{PD};\quad w^0_{DP} = v^0_{DP};\quad w^0_{PD} = v^0_{DP};\quad \text{and} \quad {}^1_{PD} = {}^1_{DP}:$$

Then for every $k \in \mathbb{N}$, ${}^k_{PD} = {}^k_{DP}$.

Proof. Since $m = 1$ and $T = \mathrm{Id}$ in both algorithms, for every $k \in \mathbb{N}$, we have that ${}^k_{PD} = p^k_{PD}$ and $w^k_{DP} = v^k_{DP}$. On one hand, by the definition of PDA, we have that

$$\begin{aligned} w^{k+1}_{PD} &= w^1_{PD} + \sum_{i=1}^{k} X p^i_{PD} \quad y \\ &= w^1_{PD} + \sum_{i=1}^{k} X (p^i_{PD} \quad p^{i\ 1}_{PD}) + \sum_{i=1}^{k} X \ {}^i_{PD} \quad y \\ &= w^1_{PD} + X (p^k_{PD} \quad p^0_{PD}) + \sum_{i=1}^{k} X \ {}^i_{PD} \quad y \\ &= w^0_{PD} + (X \ {}^k_{PD} \quad y) + \sum_{i=1}^{k} X \ {}^i_{PD} \quad y \quad ; \end{aligned} \tag{4.2.2}$$

where the last equality is obtained since $p^0_{PD} = p^0_{PD}$. Replacing (4.2.2) in the definition of ${}^{k+1}_{PD}$, we obtain

$${}^{k+1}_{PD} = \mathrm{prox}_R \quad {}^k_{PD} \quad X^{>} \quad w^0_{PD} + (X \ {}^k_{PD} \quad y) + \sum_{i=1}^{k} X \ {}^i_{PD} \quad y \quad ! \ !\ :$$

On the other hand, by DPA we have that

$$w^{k+1}_{DP} = v^{k+1}_{DP} = v^0_{DP} + \sum_{i=1}^{k\ 1} X \ {}^i_{DP} \quad y \quad ;$$

and

$$v^k_{DP} = v^k_{DP} + w^k_{DP} \quad v^{k\ 1}_{DP} = v^0_{DP} + (X \ {}^k_{DP} \quad y) + \sum_{i=1}^{k} X \ {}^i_{DP} \quad y \quad ;: \tag{4.2.3}$$

Replacing (4.2.3) in DPA, for every $k > 1$, we can deduce that

$$\begin{matrix} k+1 \\ DP \end{matrix} = \text{prox}_R \quad \begin{matrix} k \\ DP \end{matrix} \quad X^{>} \quad v^0_{DP} + \quad (X \quad \begin{matrix} k \\ DP \end{matrix} \quad y \;) + \sum_{i=1}^{N} X \quad \begin{matrix} i \\ DP \end{matrix} \quad y \quad \Bigg) \Bigg) \; :$$

Since $w^0_{PD} = v^0_{DP}$ and $\begin{matrix} 1 \\ PD \end{matrix} = \begin{matrix} 1 \\ DP \end{matrix}$ the result follows by induction. □

**Remark 4.2.4.** An analysis similar to that in the proof of Lemma 4.2.3 shows that

$$\begin{matrix} k+1 \\ PD \end{matrix} = \text{prox}_R \quad \begin{matrix} k \\ PD \end{matrix} \quad X^{>} \quad w^0_{PD} + \quad (X T_k \quad \begin{matrix} k \\ PD \end{matrix} \quad y \;) + \sum_{i=1}^{N} X \quad \begin{matrix} i \\ PD \end{matrix} \quad y \quad \Bigg) \Bigg) \; ;$$

which implies that the algorithm can be written in one step if we only care about the primal variable.

### 4.2.3 Assumptions

In the following, we list the assumptions that we require on the parameters and the operators involved in the algorithm.

**Assumption 4.2.5.** Consider the setting of PDA or DPA:

A1 The preconditioners $\quad 2 \; \mathrm{R}^{p \quad p}$ and $\quad 2 \; \mathrm{R}^{d \quad d}$ are two diagonal positive definite matrices such that

$$0 < \quad := 1 \quad k \; \tfrac{1}{2} X \; \tfrac{1}{2} k^2 : \tag{4.2.4}$$

A2 For every $k \; 2 \; \mathrm{N}$, $\quad _k \; 2 \; [m]$.

Consider the setting of PDA:

A3 The family of operators $f T_i g_{i 2 [m]}$ is from $\mathrm{R}^p$ to $\mathrm{R}^p$ and for every $i \; 2 \; [m]$:

(a) Fix $T_i \quad \mathrm{S}$ ;

(b) there exists $e_i \quad 0$ such that, for every $\quad 2 \; \mathrm{R}^p$ and $\quad 2 \; \mathrm{S}$,

$$k T_i \quad k^2 \; _1 \quad k \quad k^2 \; _1 + e_i \; ^2 : \tag{4.2.5}$$

We set $e = \max_{i 2 [m]} e_i$.

Now consider the setting of DPA:

A4 $f T_i g_{i 2 [m]}$ is a family of operators from $\mathrm{R}^d$ to $\mathrm{R}^d$ and for every $i \; 2 \; [m]$:

(a) Fix $T_i \quad \mathrm{S}$ ;

(b) for every $u \; 2 \; \mathrm{R}^d$ and $u \; 2 \; \mathrm{S}$ ,

$$k T_i u \quad u k^2 \; _1 \quad k u \quad u k^2 \; _1 :$$

**Remark 4.2.6** (Hypothesis about the operators)**.** If Assumption A3a holds and $= 0$, then Assumption A3b is implied by the quasi-nonexpansivity of $T_i$ on S. This is a weaker condition than the one proposed in [24], where, due to the generality of the setting, -averaged non-expansive operators were needed. A similar reasoning applies to Assumption A4.

## 4.3 Main results

In this section, we present and discuss the main results of the chapter. We derive convergence and stability properties of primal-dual and dual-primal splitting algorithms for linearly constrained optimization with a priori information.

First, we define the averaged iterates and the square weighted norm induced by and on $\mathrm{R}^p \quad \mathrm{R}^d$, namely

$$\hat{}^n; \hat{w}^n := \frac{\sum_{k=1}^{n} z^k}{n} \quad \text{and} \quad V(z) := \frac{k \quad k^2}{2} + \frac{k w k^2}{2};$$

where $z^k := ( \ ^k; w^k)$ is the $k$-th iterate and $z := ( \ ; w)$ is a primal-dual variable. We also recall the definition of the Lagrangian as $L( \ ; w) = R( \ ) + \langle w \mid X \quad y \rangle$.

The first result establishes the stability properties of the algorithm PDA, both in terms of the Lagrangian and feasibility gap. We recall that here we use activation operators based on the noisy data and corresponding constraints in the primal space, namely the set $C$ .

**Theorem 4.3.1.** Consider the setting of PDA under Assumptions A1, A2, and A3. Let $(p^0; \ ^0; w^0) \in \mathrm{R}^{2p} \quad \mathrm{R}^d$ be such that $^0 = p^0$. Then, for every $z = ( \ ; w) \in Z$ and for every $N \in \mathrm{N}$, we have

$$L \quad \hat{}^N; w \quad L \quad ; \hat{w}^N \quad \frac{V(z^0 \quad z)}{N} + \frac{2Nk \quad ^{\frac{1}{2}} k^2 \quad ^2}{} + k \quad ^{\frac{1}{2}} k \quad \frac{2V(z^0 \quad z)}{} \quad ^{\frac{1}{2}} + k \quad ^{\frac{1}{2}} k \quad \frac{Ne \quad ^2}{} \quad ^{\frac{1}{2}} + \frac{e \quad ^2}{2} \tag{4.3.1}$$

and

$$kX \hat{}^N \quad yk^2 \quad \frac{16Nk \quad kk \quad ^1 k \quad ^2}{2} + 8 \quad k \quad ^1 k \quad \frac{2k \quad kV(z^0 \quad z)}{3} \quad ^{\frac{1}{2}} + 8 \quad ^2 k \quad ^1 k \quad \frac{k \quad keN}{3} \quad ^{\frac{1}{2}} + \frac{8k \quad ^1 kV(z^0 \quad z)}{N} + 2 \quad ^2 + \frac{4k \quad ^1 ke \quad ^2}{}; \tag{4.3.2}$$

where we recall that the constant and $e$ are defined in Assumptions A1 and A3, respectively.

Proof. From PDA, we deduce that:

$$\begin{aligned} &^1(p^k \quad \ ^{k+1}) \quad X^{>} w^{k+1} \in @R( \ ^{k+1}) \\ &\quad ^1(w^k \quad w^{k+1}) + X p^k = y \end{aligned} \tag{4.3.3}$$

Thus

$$R( \ ^{k+1}) + \left\langle \ ^1(p^k \quad \ ^{k+1}) \quad X^{>} w^{k+1} \mid \quad ^{k+1} \right\rangle \quad R( \ ) \quad \text{for all} \quad x \in \mathrm{R}^p; \tag{4.3.4}$$

and (4.3.4) yields

$$\begin{aligned} 0 \quad & R(\ ^{k+1}) \quad R(\ ) + \Big\langle \quad ^{1}(p^k \quad p^{k+1}) \quad X^{>} w^{k+1} \, j \quad \Big\rangle_{k+1} \\ & = R(\ ^{k+1}) \quad R(\ ) + \frac{kp^k \quad k^{+1}k^2}{2} + \frac{k\ ^{k+1} \quad k^2}{2} \\ & \quad \frac{kp^k \quad k^2}{2} + \Big\langle \quad ^{k+1} \quad j \, X^{>} w^{k+1} \Big\rangle \end{aligned} \tag{4.3.5}$$

From (4.3.3) we get

$$\begin{aligned} 0 = & \Big\langle \quad ^{1}(w^k \quad w^{k+1}) + Xp^k \quad y \quad j \, w \quad w^{k+1} \Big\rangle \\ 0 = & \frac{kw^{k+1} \quad w^k k^2}{2} + \frac{kw^{k+1} \quad wk^2}{2} \quad \frac{kw^k \quad wk^2}{2} \\ & + \Big\langle y \quad Xp^k \, j \, w^{k+1} \quad w \Big\rangle \end{aligned} \tag{4.3.6}$$

Recall that

$$z := (\ ; w) \ 2 \ Z \quad S \quad \mathrm{R}^d; \ z^k := (\ ^k; w^k) \quad \text{and} \quad V(z) := \frac{k \ k^2}{2} + \frac{kwk^2}{2}:$$

Summing (4.3.5) and (4.3.6), and by Assumption A3, we obtain

$$\begin{aligned} & R(\ ^{k+1}) \quad R(\ ) + \frac{k\ ^{k+1} \quad p^k k^2}{2} + \frac{kw^{k+1} \quad w^k k^2}{2} \\ & + V(z^{k+1} \quad z) \quad V(z^k \quad z) + \Big\langle X(\ ^{k+1} \quad ) \, j \, w^{k+1} \Big\rangle \\ & \qquad + \Big\langle y \quad Xp^k \, j \, w^{k+1} \quad w \Big\rangle \quad \frac{e^{\ 2}}{2} \quad 0: \end{aligned} \tag{4.3.7}$$

Now compute

$$\begin{aligned} & R(\ ^{k+1}) \quad R(\ )) + \Big\langle X(\ ^{k+1} \quad ) \, j \, w^{k+1} \Big\rangle + \Big\langle y \quad Xp^k \, j \, w^{k+1} \quad w \Big\rangle \\ & = L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad \Big\langle X \ ^{k+1} \quad y \, j \, w \Big\rangle + \Big\langle X \quad y \, j \, w^{k+1} \Big\rangle \\ & \quad + \Big\langle X(\ ^{k+1} \quad ) \, j \, w^{k+1} \Big\rangle + \Big\langle y \quad Ap^k \, j \, w^{k+1} \quad w \Big\rangle \\ & = L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad \Big\langle X \ ^{k+1} \, j \, w \Big\rangle + \langle y \, j \, w \rangle + \Big\langle X \quad j \, w^{k+1} \Big\rangle \\ & \quad \Big\langle y \, j \, w^{k+1} \Big\rangle + \Big\langle X \ ^{k+1} \, j \, w^{k+1} \Big\rangle \quad \Big\langle X \quad j \, w^{k+1} \Big\rangle \\ & \quad + \Big\langle y \ j \, w^{k+1} \quad w \Big\rangle \quad \Big\langle Xp^k \, j \, w^{k+1} \quad w \Big\rangle \\ & = L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \Big\langle y \quad b \, j \, w^{k+1} \quad w \Big\rangle \\ & \quad + \Big\langle X \ ^{k+1} \quad Xp^k \, j \, w^{k+1} \quad w \Big\rangle \\ & \quad L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad k \tfrac{1}{2} kkw^{k+1} \quad wk \ _1 \\ & \quad + \Big\langle X \ ^{k+1} \quad Xp^k \, j \, w^{k+1} \quad w \Big\rangle : \end{aligned} \tag{4.3.8}$$

From (4.3.8) and (4.3.7) we obtain

$$
\begin{aligned}
& L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) + \frac{k\ ^{k+1} \quad p^k k^2\ _1}{2} + \frac{kw^{k+1} \quad w^k k^2\ _1}{2} \\
& + V(z^{k+1} \quad z) \quad V(z^k \quad z) \quad k\ ^{\frac{1}{2}} kkw^{k+1} \quad wk\ _1 \quad \frac{e\ ^2}{2} \\
& \quad \left\langle X(\ ^{k+1} \quad p^k)\, j\, w^{k+1} \quad w \right\rangle \qquad (4.3.9) \\
= & \quad \left\langle X(\ ^{k+1} \quad p^k)\, j\, w^{k+1} \quad w \right\rangle + \left\langle X(\ ^k \quad p^{k\ 1})\, j\, w^k \quad w \right\rangle \\
& + \left\langle X(\ ^k \quad p^{k\ 1})\, j\, w^{k+1} \quad w^k \right\rangle \qquad (4.3.10) \\
= & \quad \left\langle X(\ ^{k+1} \quad p^k)\, j\, w^{k+1} \quad w \right\rangle + \left\langle X(\ ^k \quad p^{k\ 1})\, j\, w^k \quad w \right\rangle \\
& + \left\langle \ ^{\frac{1}{2}} X\ ^{\frac{1}{2}} \ ^{\frac{1}{2}} (\ ^k \quad p^{k\ 1})\, j\ ^{\frac{1}{2}} (w^{k+1} \quad w^k) \right\rangle \\
& \quad \left\langle X(\ ^{k+1} \quad p^k)\, j\, w^{k+1} \quad w \right\rangle + \left\langle X(\ ^k \quad p^{k\ 1})\, j\, w^k \quad w \right\rangle \\
& + k\ ^{\frac{1}{2}} X\ ^{\frac{1}{2}} k^2 \frac{kw^{k+1} \quad w^k k^2\ _1}{2} + \frac{k\ ^k \quad p^{k\ 1} k^2\ _1}{2}
\end{aligned}
$$

Then, recalling that $= 1 \quad k\ ^{\frac{1}{2}} X\ ^{\frac{1}{2}} k^2$, we have the following estimate

$$
\begin{aligned}
& L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) + \frac{k\ ^{k+1} \quad p^k k^2\ _1}{2} \quad \frac{k\ ^k \quad p^{k\ 1} k^2\ _1}{2} \\
& + \frac{}{2} kw^{k+1} \quad w^k k^2\ _1 + V(z^{k+1} \quad z) \quad V(z^k \quad z) \\
& k\ ^{\frac{1}{2}} kkw^{k+1} \quad wk\ _1 \quad \left\langle X(\ ^{k+1} \quad p^k)\, j\, w^{k+1} \quad w \right\rangle \\
& + \left\langle X(\ ^k \quad p^{k\ 1})\, j\, w^k \quad w \right\rangle + \frac{e\ ^2}{2};
\end{aligned}
$$

Summing from 1 to $N \quad 1$, we obtain

$$
\begin{aligned}
& \sum_{k=1}^{N\ 1} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \\
& + \frac{k\ ^N \quad p^{N\ 1} k^2\ _1}{2} \quad \frac{k\ ^1 \quad p^0 k^2\ _1}{2} + \frac{}{2} \sum_{k=1}^{N\ 1} kw^{k+1} \quad w^k k^2\ _1 \\
& + V(z^N \quad z) \quad V(z^1 \quad z) \quad X(\ ^1 \quad p^0)\, j\, w^1 \quad w \\
& k\ ^{\frac{1}{2}} k \sum_{k=1}^{N} kw^k \quad wk\ _1 + \frac{(N \quad 1)e\ ^2}{} \\
& \left\langle \ ^{\frac{1}{2}} X\ ^{\frac{1}{2}} \ ^{\frac{1}{2}} (\ ^N \quad p^{N\ 1})\, j\ ^{\frac{1}{2}} (w^N \quad w)\ 2 \right\rangle \\
& k\ ^{\frac{1}{2}} k \sum_{k=1}^{N} kw^k \quad wk\ _1 + \frac{(N \quad 1)e\ ^2}{2} \\
& + \frac{k\ ^N \quad p^{N\ 1} k^2\ _1}{2} + k\ ^{\frac{1}{2}} X\ ^{\frac{1}{2}} k^2 \frac{kw^N \quad wk^2\ _1}{2} \qquad (4.3.11)
\end{aligned}
$$

Now, by choosing $k = 0$ in (4.3.9) we get

$$\begin{aligned} & L(\ ^1;w) \quad L(\ ;w^1) + \frac{k\ ^1 \quad p^0k^2\ _1}{2} + \frac{}{2}kw^1 \quad w^0k^2\ _1 \\ & + V(z^1 \quad z) \quad V(z^0 \quad z) + \ X(\ ^1 \quad p^0)\ j\ w^1 \quad w \\ & k\ \tfrac{1}{2}kkw^1 \quad wk\ _1 + \frac{e\ ^2}{2}: \end{aligned} \tag{4.3.12}$$

Adding (4.3.11) and (4.3.12) we obtain

$$\begin{aligned} & \sum_{k=0}^{N\ 1} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \ + \frac{}{2}kw^N \quad wk^2\ _1 \\ & + \sum_{k=1}^{N} \frac{}{2}kw^k \quad w^k\ ^1k^2\ _1 + \frac{k\ ^N \quad k^2\ _1}{2} \\ & k\ \tfrac{1}{2}k \sum_{k=1}^{N} kw^k \quad wk\ _1 + V(z^0 \quad z) + \frac{Ne\ ^2}{2} \end{aligned} \tag{4.3.13}$$

Next, by (4.3.10), we have the following estimate

$$\begin{aligned} & \frac{k\ ^{k+1} \quad p^kk^2\ _1}{2} \quad \mathrm{D}\ X(\ ^k \quad p^k\ ^1)\ j\ w^{k+1} \quad w^k\ \mathrm{E} \\ & + \frac{kw^{k+1} \quad w^kk^2\ _1}{2} + L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \\ & + V(z^{k+1} \quad z) \quad V(z^k \quad z) \\ & k\ \tfrac{1}{2}kkw^{k+1} \quad wk\ _1 \quad \mathrm{D}\ X(\ ^{k+1} \quad p^k)\ j\ w^{k+1} \quad w\ \mathrm{E} \\ & + \mathrm{D}\ X(\ ^k \quad p^k\ ^1)\ j\ w^k \quad w\ \mathrm{E} + \frac{e\ ^2}{2} \end{aligned}$$

Summing from 1 to $N \quad 1$ we obtain

$$\begin{aligned} & \sum_{k=1}^{N\ 1} \frac{k\ ^{k+1} \quad p^kk^2\ _1}{2} \quad \mathrm{D}\ X(\ ^k \quad p^k\ ^1)\ j\ w^{k+1} \quad w^k\ \mathrm{E} + \frac{kw^{k+1} \quad w^kk^2\ _1}{2}\ ! \\ & + \sum_{k=1}^{N\ 1} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \ + V(z^N \quad z) \quad V(z^1 \quad z) \\ & \quad X(\ ^1 \quad p^0)\ j\ w^1 \quad w \\ & k\ \tfrac{1}{2}k \sum_{k=1}^{N\ 1} kw^{k+1} \quad wk\ _1 + \frac{(N \quad 1)e\ ^2}{2} \\ & \quad X(\ ^N \quad p^N\ ^1)\ j\ w^N \quad w \\ = \ & k\ \tfrac{1}{2}k \sum_{k=1}^{N\ 1} kw^{k+1} \quad wk\ _1 + \frac{(N \quad 1)e\ ^2}{2} \\ & \quad \mathrm{D}\ \tfrac{1}{2}X\ \tfrac{1}{2} \quad \tfrac{1}{2}(\ ^N \quad p^N\ ^1)\ j \quad \tfrac{1}{2}(w^N \quad w)\ \mathrm{E} \\ & k\ \tfrac{1}{2}k \sum_{k=1}^{N\ 1} kw^{k+1} \quad wk\ _1 + \frac{(N \quad 1)e\ ^2}{2} \\ & + \frac{k\ \frac{1}{2}X\ \frac{1}{2}k^2}{2}k\ ^N \quad p^N\ ^1k^2\ _1 + \frac{kw^N \quad wk^2\ _1}{2} \end{aligned} \tag{4.3.14}$$

Now, since $w^{k+1} - w^k = Xp^k - y$ we derive that

$$\begin{aligned}
&\sum_{k=1}^{N-1} \frac{k^{k+1} - p^k k^2}{2} - \left\langle X(^k - p^k), w^{k+1} - w^k \right\rangle + \frac{kw^{k+1} - w^k k^2}{2} \\
&= \sum_{k=1}^{N-1} \frac{k^k - p^k k^2}{2} - \left\langle X(^k - p^k), w^{k+1} - w^k \right\rangle + \frac{kw^{k+1} - w^k k^2}{2} \\
&\quad + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&= \sum_{k=1}^{N-1} \frac{k^{\frac{1}{2}}X(^k - p^k)k^2}{2} - \left\langle \tfrac{1}{2}X(^k - p^k), \tfrac{1}{2}(Xp^k - y) \right\rangle + \frac{k^{\frac{1}{2}}(Xp^k - y)k^2}{2} \\
&\quad + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&\quad + \sum_{k=1}^{N-1} \frac{k^k - p^k k^2}{2} - \frac{k^{\frac{1}{2}}X(^k - p^k)k^2}{2} \\
&= \sum_{k=1}^{N-1} \frac{k^{\frac{1}{2}}(Xp^k - y)k^2}{2} + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&\quad + \sum_{k=1}^{N-1} \frac{k^k - p^k k^2}{2} - \frac{k^{\frac{1}{2}}X(^k - p^k)k^2}{2} \\
&\quad :
\end{aligned}$$

Furthermore, since

$$= 1 - k^{\frac{1}{2}}X^{\frac{1}{2}}k^2 > 0;$$

we obtain

$$\begin{aligned}
&\sum_{k=1}^{N-1} \frac{k^{\frac{1}{2}}(Xp^k - y)k^2}{2} + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&\quad + \sum_{k=1}^{N-1} \frac{k^k - p^k k^2}{2} - \frac{k^{\frac{1}{2}}X(^k - p^k)k^2}{2} \\
&\sum_{k=1}^{N-1} \frac{k^{\frac{1}{2}}(Xp^k - y)k^2}{2} + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&\quad + \frac{}{2} \sum_{k=1}^{N-1} k^{\frac{1}{2}}X(^k - p^k)k^2 \\
&\sum_{k=1}^{N-1} \frac{k^{\frac{1}{2}}(Xp^k - y)k^2}{2} + \frac{k^N - p^N k^2}{2} - \frac{k^1 - p^0 k^2}{2} \\
&\quad \frac{}{2} k^{\frac{1}{2}}X(^N - p^N)k^2 + \frac{}{2} k^{\frac{1}{2}}X(^1 - p^0)k^2 \\
&\quad + \frac{}{2} \sum_{k=1}^{N-1} k^{\frac{1}{2}}X(^{k+1} - p^k)k^2:
\end{aligned}$$

In turn, by the convexity of $k \quad k^2$, we obtain

$$\begin{aligned}
&\sum_{k=1}^{N-1} k^{\frac{1}{2}}(Xp^k \quad y)k^2 + \frac{k^{N} \quad p^{N-1}k^2_{1} \quad k^{1} \quad p^0k^2_{1}}{2} \\
&\quad \frac{}{2}k^{\frac{1}{2}}X(^{N} \quad p^{N-1})k^2 + \frac{}{2}k^{\frac{1}{2}}X(^{1} \quad p^0)k^2 \\
&+ \frac{}{2}\sum_{k=1}^{N-1} k^{\frac{1}{2}}X(^{k+1} \quad p^k)k^2 \\
&\frac{}{4}\sum_{k=1}^{N-1} k^{\frac{1}{2}}(X^{k+1} \quad y)k^2 \quad \frac{k^{1} \quad p^0k^2_{1}}{2} + \frac{}{2}k^{\frac{1}{2}}X(^{1} \quad p^0)k^2 \\
&+ \frac{^2 + k^{\frac{1}{2}}X^{\frac{1}{2}}k^2}{2}k^{N} \quad p^{N-1}k^2_{1} \\
&\frac{}{4}\sum_{k=2}^{N} k^{\frac{1}{2}}(X^{k} \quad y)k^2 \quad \frac{k^{1} \quad p^0k^2_{1}}{2} + \frac{}{2}k^{\frac{1}{2}}X(^{1} \quad p^0)k^2 \\
&+ \frac{^2 + k^{\frac{1}{2}}X^{\frac{1}{2}}k^2}{2}k^{N} \quad p^{N-1}k^2_{1}: \qquad (4.3.15)
\end{aligned}$$

On the other hand, we get

$$\begin{aligned}
k^{\frac{1}{2}}(X^{k} \quad y)k^2 \quad & \frac{kX^{k} \quad y\,k^2}{k \quad ^1k} \\
& \frac{1}{k \quad ^1k} \quad \frac{kX^{k} \quad yk^2}{2} \quad ky \quad yk^2 \quad : \qquad (4.3.16)
\end{aligned}$$

Combining (4.3.12), (4.3.14), (4.3.15), and (4.3.16) we have that

$$\begin{aligned}
&\sum_{k=0}^{N-1} L(^{k+1}; w) \quad L(\ ; w^{k+1}) \quad + \frac{^2}{2}k^{N} \quad p^{N-1}k^2_{1} \\
&\sum_{k=1}^{N} \frac{}{8k \quad ^1k}kX^{k+1} \quad bk^2 + \frac{k^{N} \quad k^2_{1}}{2} \\
&k^{\frac{1}{2}}k\sum_{k=1}^{N} kw^k \quad wk_{1} + V(z^0 \quad z) + \frac{Ne^{2}}{2} + N\frac{}{4k \quad ^1k}^{2} \qquad (4.3.17)
\end{aligned}$$

It remains to bound $k^{\frac{1}{2}}k\sum_{k=1}^{N} kw^k \quad wk_{1}$. From (4.3.13) and since $(x;u)$ is a saddle-point of the Lagrangian we deduce that

$$kw^N \quad wk^2 \quad \frac{2k^{\frac{1}{2}}k}{} \sum_{k=1}^{N} kw^k \quad wk + \frac{2V(z^0 \quad z)}{} + \frac{Ne^{2}}{}: \qquad (4.3.18)$$

Applying [111, Lemma A.1] to Equation (4.3.18) with

$$_k := \frac{2k^{\frac{1}{2}}k}{} \quad \text{and} \quad S_N := \frac{2V(z^0 \quad z)}{} + \frac{Ne^{2}}{}$$

we get

$$kw^N \quad wk \quad \frac{Nk\ ^{\frac{1}{2}}k}{} + @\frac{2V(z^0 \quad z)}{} + \frac{Ne\ ^2}{} + \quad \frac{Nk\ ^{\frac{1}{2}}k}{} \quad !_2 1_{\frac{1}{2}}$$

$$\frac{2Nk\ ^{\frac{1}{2}}k}{} + \quad \frac{2V(z^0 \quad z)}{} \quad ^{\frac{1}{2}} + \quad \frac{Ne\ ^2}{} \quad ^{\frac{1}{2}} : \tag{4.3.19}$$

Inserting (4.3.19) into (4.3.13) to obtain

$$\sum_{k=0}^{N} L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad \frac{2(Nk\ ^{\frac{1}{2}}k\ )^2}{} + Nk\ ^{\frac{1}{2}}k \quad \frac{V(z^0 \quad z)}{} \quad ^{\frac{1}{2}}$$

$$+ Nk\ ^{\frac{1}{2}}k \quad \frac{Ne\ ^2}{} \quad ^{\frac{1}{2}} + V(z^0 \quad z) + \frac{Ne\ ^2}{2}:$$

By (4.3.17), we have

$$\sum_{k=1}^{N} kX\ ^k \quad yk^2 \quad \frac{16N^2k\ kk \quad ^1k\ ^2}{2} + 8N\ k \quad ^1k \quad \frac{2k\ kV(z^0 \quad z)}{3} \quad ^{\frac{1}{2}}$$

$$+ 8N\ ^2k \quad ^1k \quad \frac{k\ keN}{3} \quad ^{\frac{1}{2}} + \frac{8k \quad ^1kV(z^0 \quad z)}{}$$

$$+ 2N\ ^2 + \frac{4Nk \quad ^1ke\ ^2}{}$$

and both results are straightforward from Jensen's inequality. □

Note that, the previous proof combines and extends the techniques developed in [24] and [84], based on the firm non-expansivity of the proximal point operator and discrete Bihari's lemma to deal with the error; see also [111].

In the next result, we establish upper bounds for the Lagrangian and feasibility gap analogous to those proposed in Theorem 4.3.1, but for the algorithm DPA. The main difference is that now the activation step is based on a priori information in the dual space $\mathrm{R}^d$, and not on $S$ . This set is represented by the intersection of fixed point sets of a finite number of operators and encodes some knowledge about the dual solution.

**Theorem 4.3.2.** Consider the setting of DPA under Assumptions A1, A2, and A4. Let $(v^0; w^0; \ ^0)\ 2\ \mathrm{R}^{2d} \quad \mathrm{R}^p$ be such that $w^0 = v^0$. Then, for every $z = (\ ; w)\ 2\ Z$ and for every $N\ 2\ \mathrm{N}$, we have that

$$L \quad ^{\wedge N}; w \quad L \quad ; \hat{w}^N \quad \frac{V(z^0 \quad z)}{N} + 2k\ ^{\frac{1}{2}}k^2N\ ^2 + k\ ^{\frac{1}{2}}k \quad 2V(z^0 \quad z)\ ^{\frac{1}{2}}; \tag{4.3.20}$$

and

$$kX\ ^{\wedge N} \quad yk^2 \quad \frac{8k\ ^{\frac{1}{2}}k^2k \quad ^1kN\ ^2}{} + \frac{4k\ ^{\frac{1}{2}}kk \quad ^1k \quad 2V(z^0 \quad z)\ ^{\frac{1}{2}}}{}$$

$$+ \frac{4k \quad ^1kV(z^0 \quad z)}{N} + 2\ ^2: \tag{4.3.21}$$

where we recall that the constant is defined in Assumptions A1.

Proof. It follows from DPA that

$$\begin{aligned} & {}^{1}( \ {}^{k} \quad {}^{k+1}) \quad X^{>} v^{k} \ 2 \ @R( \ {}^{k+1}) \\ & \quad {}^{1}(v^{k} \quad w^{k+1}) + X \ {}^{k+1} = y \end{aligned} \tag{4.3.22}$$

Thus,

$$R( \ {}^{k+1}) + \overset{D}{} \ {}^{1}( \ {}^{k} \quad {}^{k+1}) \quad X^{>} v^{k} \ j \quad {}^{k+1} \overset{E}{} \quad R( \ ) \tag{4.3.23}$$

and (4.3.23) yields

$$\begin{aligned} 0 \quad & R( \ {}^{k+1}) \quad R( \ ) + \overset{D}{} \ {}^{1}( \ {}^{k} \quad {}^{k+1}) \quad X^{>} v^{k} \ j \quad {}^{k+1} \overset{E}{} \\ & = R( \ {}^{k+1}) \quad R( \ ) + \frac{k \ {}^{k} \quad {}^{k+1} k^{2} \ {}^{1}}{2} + \frac{k \ {}^{k+1} \quad k^{2} \ {}^{1}}{2} \\ & \quad \frac{k \ {}^{k} \quad k^{2} \ {}^{1}}{2} + \overset{D}{} \ {}^{k+1} \quad j \ X^{>} v^{k} \overset{E}{} \end{aligned} \tag{4.3.24}$$

From (4.3.22), it follows that

$$\begin{aligned} 0 = & \overset{D}{} \ {}^{1}(v^{k} \quad w^{k+1}) + X \ {}^{k+1} \quad y \ j \ w \quad w^{k+1} \overset{E}{} \\ 0 = & \frac{k w^{k+1} \quad v^{k} k^{2} \ {}^{1}}{\overset{D}{} \ 2} + \frac{k w^{k+1} \quad w k^{2} \ {}^{1}}{2 \overset{E}{}} \quad \frac{k v^{k} \quad w k^{2} \ {}^{1}}{2} \\ & + \ y \quad X \ {}^{k+1} \ j \ w^{k+1} \quad w \end{aligned} \tag{4.3.25}$$

Recall that $z := ( \ ; w) \ 2 \ Z \quad S \quad \mathrm{R}^{d}$, $z^{k} := ( \ {}^{k}; w^{k})$, and $V(z) := \frac{k \ k^{2} \ {}^{1}}{2} + \frac{k w k^{2} \ {}^{1}}{2}$. Summing (4.3.24) and (4.3.25), we obtain

$$\begin{aligned} & R( \ {}^{k+1}) \quad R( \ ) + \frac{k \ {}^{k+1} \quad {}^{k} k^{2} \ {}^{1}}{\overset{D}{} \ 2} + \frac{k w^{k+1} \quad v^{k} k^{2} \ {}^{1}}{\overset{E}{} \ \overset{D}{} \ 2} + V(z^{k+1} \quad z) \\ & V(z^{k} \quad z) + \ X( \ {}^{k+1} \quad ) \ j \ v^{k} + \ y \quad X \ {}^{k+1} \ j \ w^{k+1} \quad w \overset{E}{} \quad 0: \end{aligned} \tag{4.3.26}$$

Now compute

$$
\begin{aligned}
&R(\ ^{k+1}) \quad R(\ ) + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \ + \ y \quad X \ ^{k+1} j \, w^{k+1} \quad w \Big\rangle \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad \Big\langle X \ ^{k+1} \quad y \, j \, w \ + \ X \quad y \, j \, w^{k+1} \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \ + \ y \quad X \ ^{k+1} j \, w^{k+1} \quad w \Big\rangle \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad \Big\langle X \ ^{k+1} j \, w \ + \hbar y \, j \, w i + \ X \quad j \, w^{k+1} \Big\rangle \\
&\quad \Big\langle y \, j \, w^{k+1} \ + \ X(\ ^{k+1} \quad ) \, j \, v^k \ + \ y \ j \, w^{k+1} \quad w \Big\rangle \\
&\quad \Big\langle X \ ^{k+1} j \, w^{k+1} \ + \ X \ ^{k+1} j \, w \Big\rangle \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \ \Big\langle y \quad y \, j \, w^{k+1} \quad w \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \Big\rangle \qquad (4.3.27) \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \ \Big\langle y \quad y \, j \, w^{k+1} \quad w \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \ + \ X(\ ^{k+1} \quad ) \, j \, w^k \quad v^{k \ 1} \Big\rangle \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \ \Big\langle y \quad y \, j \, w^{k+1} \quad w \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \ + \ X(\ ^{k} \quad ) \, j \, w^k \quad v^{k \ 1} \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ^k) \, j \, w^k \quad v^{k \ 1} \Big\rangle \\
&= L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \ \Big\langle y \quad y \, j \, w^{k+1} \quad w \Big\rangle \\
&\quad + \ \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \ + \ X(\ ^{k} \quad ) \, j \, w^k \quad v^{k \ 1} \Big\rangle \\
&\quad + \ \Big\langle \tfrac{1}{2} X(\ ^{k+1} \quad ^k) \, j \quad \tfrac{1}{2}(w^k \quad v^{k \ 1}) \Big\rangle : \qquad (4.3.28)
\end{aligned}
$$

From (4.3.28) and (4.3.26) we obtain

$$
\begin{aligned}
&L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) + \frac{k \ ^{k+1} \quad ^k k^2 \ ^1}{2} \\
&+ \frac{k w^{k+1} \quad v^k k^2 \ ^1}{2} + V(z^{k+1} \quad z) \quad V(z^k \quad z) \\
&\Big\langle y \quad y \, j \, w^{k+1} \quad w \Big\rangle \quad \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \Big\rangle \\
&+ \ \Big\langle X(\ ^{k} \quad ) \, j \, v^{k \ 1} \quad w^k \Big\rangle \quad \Big\langle \tfrac{1}{2} X(\ ^{k+1} \quad ^k) \, j \quad \tfrac{1}{2}(w^k \quad v^{k \ 1}) \Big\rangle \\
&k \ \tfrac{1}{2} k k w^{k+1} \quad w k \ ^1 \quad \Big\langle X(\ ^{k+1} \quad ) \, j \, v^k \quad w^{k+1} \Big\rangle \\
&+ \ \Big\langle X(\ ^{k} \quad ) \, j \, v^{k \ 1} \quad w^k \Big\rangle + \frac{k \ ^{k+1} \quad ^k k^2 \ ^1}{2} \\
&+ k \ \tfrac{1}{2} X \ \tfrac{1}{2} k^2 \frac{k w^k \quad v^{k \ 1} k^2 \ ^1}{2}
\end{aligned}
$$

Therefore we have that

$$\begin{aligned} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) + \frac{kw^{k+1} \quad v^k k^2\ _1}{2} \\ k\ ^{\frac{1}{2}}X\ ^{\frac{1}{2}}k^2\frac{kw^k \quad v^k\ ^1k^2\ _1}{2} + V(z^{k+1} \quad z) \quad V(z^k \quad z) \\ k\ ^{\frac{1}{2}}kkw^{k+1} \quad wk\ _1 \quad X(\ ^{k+1} \quad )\ j\ v^k \quad w^{k+1} \\ + \quad X(\ ^k \quad )\ j\ v^k\ ^1 \quad w^k \end{aligned} \tag{4.3.29}$$

Summing from 1 to $N \quad 1$ we obtain

$$\begin{aligned} \sum_{k=1}^{N\ 1} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \ + \frac{}{2}\sum_{k=1}^{N\ 1} kw^{k+1} \quad v^k k^2\ _1 \\ + V(z^N \quad z) + k\ ^{\frac{1}{2}}X\ ^{\frac{1}{2}}k^2\frac{kw^N \quad v^N\ ^1k^2\ _1}{2} \\ k\ ^{\frac{1}{2}}k\sum_{k=1}^{N\ 1} kw^{k+1} \quad wk \quad X(\ ^N \quad )\ j\ v^N\ ^1 \quad w^N \\ + \quad X(\ ^1 \quad )\ j\ v^0 \quad w^1 \ + V(z^1 \quad z) \\ k\ ^{\frac{1}{2}}k\sum_{k=1}^{N\ 1} kw^{k+1} \quad wk + k\ ^{\frac{1}{2}}X\ ^{\frac{1}{2}}k^2\frac{kw^N \quad v^N\ ^1k^2\ _1}{2} \\ + \frac{k\ ^N \quad k^2\ _1}{2} + \quad X(\ ^1 \quad )\ j\ v^0 \quad w^1 \ + V(z^1 \quad z) \end{aligned} \tag{4.3.30}$$

Reordering (4.3.30) we obtain

$$\begin{aligned} \sum_{k=1}^{N\ 1} L(\ ^{k+1};w) \quad L(\ ;w^{k+1}) \\ + \frac{}{2}\sum_{k=1}^{N\ 1} kw^{k+1} \quad v^k k^2\ _1 + \frac{kw^N \quad wk^2\ _1}{2} \\ k\ ^{\frac{1}{2}}k\sum_{k=1}^{N\ 1} kw^{k+1} \quad wk + \quad X(\ ^1 \quad )\ j\ v^0 \quad w^1 \ + V(z^1 \quad z): \end{aligned} \tag{4.3.31}$$

On the other hand, from (4.3.26) and (4.3.27) we get

$$\begin{aligned} L(\ ^1;w) \quad L(\ ;w^1) + \frac{}{2}kw^1 \quad v^0 k^2 \quad kw^1 \quad wk \\ X(\ ^1 \quad )\ j\ v^0 \quad w^1 \\ + V(z^0 \quad z) \quad V(z^1 \quad z) \end{aligned} \tag{4.3.32}$$

Summing (4.3.31) and (4.3.32) yields

$$\begin{aligned} \sum_{k=1}^{N} L(\ ^k;w) \quad L(\ ;w^k) \\ + \frac{}{2}\sum_{k=1}^{N} kw^k \quad v^k\ ^1k^2\ _1 + \frac{kw^N \quad wk^2\ _1}{2} \\ k\ ^{\frac{1}{2}}k\sum_{k=1}^{N} kw^k \quad wk + V(z^0 \quad z): \end{aligned} \tag{4.3.33}$$

Moreover, since $w^{k+1} \quad v^k = \quad (X \quad ^{k+1} \quad y\ )$ we have

$$kw^{k+1} \quad v^k k^2 \ _1 = \overset{D}{} \quad (X \quad ^{k+1} \quad y\ ) j\ X \quad ^{k+1} \quad y \overset{E}{}$$
$$\frac{kX \quad ^{k+1} \quad y\ k^2}{k \quad ^1k}$$
$$\frac{1}{k \quad ^1k} \quad \frac{kX \quad ^{k+1} \quad yk^2}{2} \quad ky \quad yk^2 \tag{4.3.34}$$

and from (4.3.33) and (4.3.34) we obtain

$$\overset{N}{\underset{k=0}{X}}{}^1 \quad L(\ ^{k+1}; w) \quad L(\ ; w^{k+1})$$
$$+ \frac{}{4k \quad ^1k} \overset{N}{\underset{k=1}{X}} kX \quad ^k \quad yk^2 + \frac{kw^N \quad wk^2 \ _1}{2}$$
$$k \quad ^{\frac{1}{2}}k \overset{N}{\underset{k=1}{X}} kw^k \quad wk \quad _1 + V(z^0 \quad z) + \frac{N \quad ^2}{2k \quad ^1k} \tag{4.3.35}$$

From (4.3.33) it follows that

$$kw^N \quad wk^2 \ _1 \quad 2 \ k \ ^{\frac{1}{2}}k \overset{N}{\underset{k=1}{X}} kw^k \quad wk \quad _1 + 2V(z^0 \quad z); \tag{4.3.36}$$

Applying [111, Lemma A.1] to Equation (4.3.36) with $_k := 2 \ k \ ^{\frac{1}{2}}k$ and $S_k := 2V(z^0 \quad z)$ to get

$$kw^k \quad wk \quad _1 \quad Nk \ ^{\frac{1}{2}}k \ + \ 2V(z^0 \quad z) + \ Nk \ ^{\frac{1}{2}}k \quad ^{2 \ \frac{1}{2}}$$
$$2Nk \ ^{\frac{1}{2}}k \ + \ 2V(z^0 \quad z) \ ^{\frac{1}{2}} \tag{4.3.37}$$

Insert the previous in Equation (4.3.33), to obtain

$$\overset{N}{\underset{k=0}{X}}{}^1 \quad L(\ ^{k+1}; w) \quad L(\ ; w^{k+1}) \quad 2k \ ^{\frac{1}{2}}k^2 N^{2 \ 2} + Nk \ ^{\frac{1}{2}}k \quad 2V(z^0 \quad z) \ ^{\frac{1}{2}}$$
$$+ V(z^0 \quad z) \tag{4.3.38}$$

and by (4.3.35) and (4.3.37) we have

$$\overset{N}{\underset{k=1}{X}} kA \quad ^k \quad bk^2 \quad \frac{4k \quad ^1k}{} \quad 2k \ ^{\frac{1}{2}}k^2 N^{2 \ 2} + Nk \ ^{\frac{1}{2}}k \quad 2V(z^0 \quad z) \ ^{\frac{1}{2}}$$
$$\frac{4k \quad ^1k}{} \quad V(z^0 \quad z) + \frac{N \quad ^2}{2k \quad ^1k} \tag{4.3.39}$$

and both results follows from Jensen's inequality. □

First, we comment on the chosen optimality measures. As discussed in [84, 85, 111], the Lagrangian gap is equivalent to the Bregman distance of the iterates to the solution. If the penalty is strongly convex, the Bregman divergence is an upper bound of the squared norm of the difference between the reconstructed and the ideal solution, while if $R$ is only convex, the Bregman divergence gives only limited information, and in general, it is a very

weak convergence measure. For instance, in the exact case, a vanishing Lagrangian gap does not imply that cluster points of the generated sequence are primal solutions. However, as can be derived from [85], a vanishing Lagrangian gap coupled with a vanishing feasibility gap implies that every cluster point of the primal sequence is a solution to the primal problem.

In both theorems, the established result ensures that the two optimality measures can be upper bounded with the sum of two terms. The first one, which can be interpreted as an optimization error, is of the order $O(N^{-1})$, and so it goes to zero as $N$ tends to $+1$. Note that, in the exact case $= 0$, only this term is present and both the Lagrangian and the feasibility gap are indeed vanishing, guaranteeing that every cluster point of the sequence is a primal solution. The second term, which can be interpreted as a stability control, collects all the errors due to the perturbation of the exact datum and also takes into account the presence of the activation operators $T$, when the data constraints are noisy. It is an increasing function of the number of iterations and the noise level .

**Remark 4.3.3.** Theorems 4.3.1 and 4.3.2 are an extension of [24, Theorem 1], where it is proved that the sequence generated by the algorithms converges to an element in $Z$ when $= 0$, but neither convergence rates nor stability bounds were given. In this work, we filled the gap for linearly constrained convex optimization problems. Moreover, in the noise-free case, our assumptions on the additional operators $T$ are weaker than those proposed in [24], where -averagedness is required. For the noisy case, without the activation operators (and so with $e = 0$), our bounds are of the same order as in [84] in the number of iterations and noise level ( ).

As mentioned above, in (4.3.1) and (4.3.2), when $> 0$ and $N \to +1$, the upper bounds for the PDA iterates tend to infinity, and the iteration may not converge to the desired solution. The same comment can be made for the DPA iterates, based on (4.3.20) and (4.3.21). In both cases, to obtain a minimal reconstruction error, we need to impose a trade-off between convergence and stability. The next corollary introduces an early stopping criterion, depending only on the noise level and leading to stable reconstruction.

**Corollary 4.3.4.** (Early-stopping). Under the assumptions of Theorem 4.3.1 or Theorem 4.3.2, choose $N = c=$ for some $c > 0$. Then, for every $z = (\ ; w) \ 2 \ Z$, there exist constants $c_1$, $c_2$, and $c_3$ such that

$$L \quad ^{N}; w \quad L \quad ; w^{N} \quad c_1$$
$$kX \ ^{N} \quad yk^2 \quad c_2 \ + c_3 \ ^2:$$

The early stopping rule prescribed above is computationally efficient in the sense that the number of iterations is proportional to the inverse of the noise level. In particular, if the error is small, then more iterations are useful, while if is big, it is convenient to stop sooner. So, the number of iterations plays the role of a regularization parameter. Using the early stopping strategy proposed above, we can see that the error in the data transfers to the error in the solution with the same noise level, which is the best that one can expect for a general operator $X$.

**Remark 4.3.5. Comparison with Tikhonov regularization.** The reconstruction properties of the proposed algorithms are comparable to the ones obtained using Tikhonov regularization [18, 52], with the same dependence on the noise level. We underline that in [18, Theorem 5.1] only the Bregman divergence is considered and not the feasibility. In addition, iterative regularization is way more efficient from a computational point of view, as it requires the solution of only one optimization problem, while Tikhonov regularization amounts to solving a family of problems indexed by the regularization parameter. Let us also note that, when is unknown, any principle used to determine a suitable can be used to determine the stopping time.

## 4.4 Implementation details

In this section, we discuss some standard choices to construct non-expansive operators $T$ that satisfy our assumptions and encode some redundant information on the solution set. We first present examples for PDA, and later for DPA.

To define the operators, we first recall how to compute the projection on the constraint determined by each datum. For every $j \in [d]$, we denote by $x_j$ the $j$-th row of $X$ and by $P_j$ the projection onto the $j$-th linear equation; namely,

$$P_j : \mathbb{R}^p \to \mathbb{R}^p; \quad \to \quad + \frac{y_j \quad \langle x_j, \quad \rangle}{\|x_j\|^2} x_j^{>}:$$

Analogously, for every $j \in [d]$, we denote by $P_j$ the projection operator as in the previous definition but with the noisy data $y$ instead of $y$.

We proceed to define the four families of operators proposed in this chapter for PDA.

**Definition 4.4.1.** Consider the operator $T : \mathbb{R}^p \to \mathbb{R}^p$ is a

1. **Serial projection** if

$$T = P_{\quad} \quad P_{\quad 1};$$

   where, for every $j \in [l]$, $\quad_j \in [d]$.

2. **Parallel projection** if

$$T = \sum_{j=1}^{} \quad_j P_{\quad j} \tag{4.4.1}$$

   where, for every $j \in [l]$, $\quad_j \in [d]$ and $(\quad_j)_{j=1}^{l}$ are real numbers in $[0; 1]$, such that $\sum_{j=1}^{p} \quad_j = 1$.

3. **Landweber operator** with parameter if

$$T : \mathbb{R}^p \to \mathbb{R}^p; \quad \to \quad X^{>}(X \quad y): \tag{4.4.2}$$

   where $\in ]0; \frac{2}{\|X\|\|X\|^2}[$.

4. **Landweber operator with adaptive step** and parameter $M$ if

$$T : \mathbb{R}^p \to \mathbb{R}^p; \quad \to \begin{cases} (\ )X^{>}(X \quad y) & \text{if } X^{>}X \not\in X^{>}y \\ \text{otherwise.} \end{cases} \tag{4.4.3}$$

where, for $M > 0$, $(\ ) = \min\left\{\frac{kX \quad y\ k^2}{kX^{>}(X \quad y)k^2}; M\right\}$.

The next lemma states that the operators in Definition 4.4.1 satisfy Assumption A3.

**Lemma 4.4.2.** Let $T : \mathbb{R}^p \to \mathbb{R}^p$ be one of the operators given in Definition 4.4.1. Then Assumption A3 holds with the following

1. If $T$ is a serial projection, then
$$e = \frac{1}{\max_{i=1,\dots,d} kx_i k}$$
2. If $T$ is a parallel projection, then
$$e_T = \sum_{j=1}^{} \frac{j}{kx_j k^2}:$$
3. If $T$ is the Landweber operator with parameter ,then
$$e_T = \frac{}{2 \quad kXk^2}:$$
4. If $T$ is the Landweber operator with adaptive step and parameter $M$, then $e_T = M$:

Proof. Let us first recall that

$$P : \to + \frac{y_j \quad hx_j\ j \quad i}{kx_j k^2} x_j^{>}:$$

Note that the $j$-th equation of $S$ and $S$ are parallel. Then, for every $j \in [d]$ and $\in S$, we get

$$\begin{aligned} kP_j \quad k^2 &= kP_j \quad k^2 + 2\ \langle P_j \quad j\ P_j \quad P_j \rangle \\ &\quad + kP_j \quad P_j \quad k^2 \\ &= kP_j \quad k^2 + kP_j \quad P_j \quad k^2: \end{aligned} \tag{4.4.4}$$

Analogously, we have that

$$k \quad k^2 = k \quad P_j \quad k^2 + kP_j \quad k^2: \tag{4.4.5}$$

It follows from (4.4.4) and (4.4.5) that

$$kP_j \quad k^2 + k \quad P_j \quad k^2 = k \quad k^2 + kP_j \quad P_j \quad k^2:$$

Hence,

$$\begin{aligned} kP_j \quad k^2 \quad k \quad k^2 + kP_j \quad P_j \quad k^2 \\ k \quad k^2 + \frac{(y_j \quad y_j)^2}{kx_j k^2} \\ k \quad k^2 + \frac{^2}{kx_j k^2} \end{aligned}$$

1. Since $T = P_l \cdots P_1$ it is clear that $S \subset \operatorname{Fix} T$ and by induction we have that,
$$\|Tx - x^\dagger\|^2 \le \|x - x^\dagger\|^2 + \varepsilon^2;$$
where $\varepsilon = \frac{l}{\max_{i=1,\dots,d} \|x_i\|}$.

2. The proof follows from the convexity of $x \mapsto \|x\|^2$ which is obtained with $\varepsilon = \frac{1}{\max_{i=1,\dots,d} \|x_i\|}$.

3. Let $x \in S$. By (4.4.2), we have
$$\begin{aligned}\|Tx - x^\dagger\|^2 &= \|x - x^\dagger\|^2 - 2\left\langle \frac{X_j X_j^\top (Xx - y)}{\|X_j\|^2}, x - x^\dagger \right\rangle + \frac{1}{\|X_j\|^2}\|X_j^\top (Xx - y)\|^2 \\ &= \|x - x^\dagger\|^2 - 2\left\langle \frac{X_j X_j^\top}{\|X_j\|^2} X x - y, x - x^\dagger \right\rangle + \frac{1}{\|X_j\|^2}\|X_j^\top (Xx - y)\|^2 \\ &= \|x - x^\dagger\|^2 - 2\left\langle \frac{X_j X_j^\top}{\|X_j\|^2} y - y, x - x^\dagger \right\rangle + \frac{1}{\|X_j\|^2}\|X_j\|^2 \|Xx - y\|^2 \\ &= \|x - x^\dagger\|^2 - 2\left\langle \frac{X_j X_j^\top}{\|X_j\|^2} y - y, x - x^\dagger \right\rangle + \|Xx - y\|^2\end{aligned}$$
Now using the Young inequality with parameter $\|X_j\|^2$, we have that
$$\begin{aligned}\|Tx - x^\dagger\|^2 &\le \|x - x^\dagger\|^2 + \frac{\|X_j\|^2}{\|X_j\|^2}\|y - y\|^2 + \|Xx - y\|^2 \\ &\le \|x - x^\dagger\|^2 + \frac{\varepsilon^2}{\|X_j\|^2}.\end{aligned}$$
It remains to prove that, if $S \neq \emptyset$, then $S \subset \operatorname{Fix} T$, which is clear from (4.4.2).

4. Let $x \in S$ and $x \in \mathbb{R}^p$. If $X^\top X x = X^\top y$, then (4.2.5) immediately holds. Otherwise, we have
$$\begin{aligned}\|Tx - x^\dagger\|^2 &= \|x - x^\dagger\|^2 - 2\lambda_j \left\langle \frac{X_j X_j^\top (Xx - y)}{\|X_j\|^2}, x - x^\dagger \right\rangle + \frac{\lambda_j^2}{\|X_j\|^2}\|X_j^\top (Xx - y)\|^2 \\ &= \|x - x^\dagger\|^2 - 2\lambda_j \left\langle \frac{X_j X_j^\top}{\|X_j\|^2} X x - y, x - x^\dagger \right\rangle + \frac{\lambda_j^2}{\|X_j\|^2}\|X_j^\top (Xx - y)\|^2 \\ &= \|x - x^\dagger\|^2 - 2\lambda_j \left\langle \frac{X_j X_j^\top}{\|X_j\|^2} y - y, x - x^\dagger \right\rangle + \frac{\lambda_j^2}{\|X_j\|^2}\|X_j\|^2 \|Xx - y\|^2 \\ &= \|x - x^\dagger\|^2 - 2\lambda_j \left\langle \frac{X_j X_j^\top}{\|X_j\|^2} y - y, x - x^\dagger \right\rangle + \lambda_j^2 \|Xx - y\|^2\end{aligned}$$
Now using the Young inequality with parameter $\lambda_j \frac{\|X_j^\top (Xx - y)\|^2}{\|Xx - y\|^2}$, we have that
$$\begin{aligned}\|Tx - x^\dagger\|^2 &\le \|x - x^\dagger\|^2 + \frac{\lambda_j}{\lambda_j \frac{\|X_j^\top (Xx - y)\|^2}{\|Xx - y\|^2}}\|y - y\|^2 + \lambda_j^2 \|Xx - y\|^2 \\ &\le \|x - x^\dagger\|^2 + \lambda_j^2 M^2.\end{aligned}$$
Finally, it is clear from (4.4.3) that, if $S \neq \emptyset$, then $S \subset \operatorname{Fix} T$.

□

**Remark 4.4.3. Relationship between Parallel projection and Landweber operator.** A particular parallel projection is the one corresponding to $l = d$, $\lambda_j = j$, and $\varepsilon_j = \frac{\|x_j\|^2}{\|X\|_F^2}$. Then, (4.4.1) reduces to
$$T(x) = x - \frac{1}{\|X\|_F^2} X^\top (Xx - y).$$

Observe that, since $\|X\| \leq \|X\|_F$, the previous is a special case of the Landweber operator with $\tau = \frac{1}{\|X\|_F^2}$.

**Remark 4.4.4. Steepest descent.** Let $\bar{x} \in \mathbb{R}^p$ be such that $X\bar{x} = b$. Then, from (4.4.3), we derive

$$\|T x - \bar{x}\|^2 = \|x - \bar{x}\|^2 - 2\lambda(\tau) \left\langle x - y, y - X\bar{x} \right\rangle - 2\lambda(\tau)\|X(x - y)\|^2 + \lambda(\tau)^2 \|X^\top (X(x - y))\|^2. \tag{4.4.6}$$

If $\alpha = 0$, then the choice of $\lambda(\tau)$ given in (4.4.3) minimizes the right-hand side of (4.4.6), if the minimizer is smaller than $M$. In this case, $\lambda$ is chosen in order to maximize the contractivity with respect to a fixed point of $T$. While we cannot repeat the same procedure for $\alpha > 0$, since we do not know $y$, we still keep the same choice. If $y \in \mathrm{ran}(X)$, then $\sup_{x \in \mathbb{R}^p} \|X(x - y)\|^2 / \|X^\top (X(x - y))\|^2 < +\infty$. However, in general, if $\alpha > 0$, this is not true and $M$ is needed to ensure that $\lambda(\tau)$ is bounded.

**Remark 4.4.5.** From a computational point of view, parallel projections and Landweber operators are more efficient than serial projections. In particular, note that the quantity $(X x^k - y)$ needs to be computed anyway in the other steps of the algorithm.

While for the primal space the data constraints that we want to reuse are clearly given by the linear constraints, for the dual there is not always a natural choice. In the following we present an example related to the $\ell^1$ norm. A similar implementation can be extended to the case of 1-homogenous penalty functions, for which the Fenchel conjugate is the indicator of a closed and convex subset of the dual space [15, Proposition 14.11 (ii)].

**Example 4.4.6.** Consider the noisy version of problem ($P$)with $R(\cdot) = \|\cdot\|_1$. Then the dual is given by

$$\min_{w \in \mathbb{R}^d} \left\{ \left\langle y, w \right\rangle : |(X^\top w)_i| \leq 1, \text{ for every } i \in [p] \right\}.$$

For every $i \in [p]$, set $D_i = \{w \in \mathbb{R}^d : |(X^\top w)_i| \leq 1\}$ and denote by $T_i$ the projection over $D_i$. Note that this projection is easy to compute, see for example [15, Example 28.17], since it is the projection onto the intersection of two parallel half-hyperplane. Clearly Assumption A4 holds. Differently from the primal case, here we are projecting on exact constraints, which are independent of the noisy data $y$.

## 4.5 Numerical results

In this section, to test the efficiency of the proposed algorithms, we perform numerical experiments in two settings: sparse reconstruction with $\ell^1$-norm regularization and Image denoising and deblurring with total variation regularization. For the $\ell^1$-norm regularization, we compare our results with other regularization techniques. In the more complex problem of total variation we explore the properties of different variants of our procedure.

**Code statement:** All numerical examples are implemented in MATLAB® on a laptop. In the second experiment we also use the library Numerical tours [108]. The corresponding

code can be downloaded at https://github.com/cristianvega1995/L1-TV-Experiments-of-Fast-iterative-regularization-by-reusing-data-constraints

### 4.5.1 $\ell^1$-norm regularization

In this section, we apply the routines PDA and DPA when $R$ is equal to the $\ell^1$-norm. We compare the results given by our method with two state-of-the-art regularization procedures: iterative regularization by vanilla primal-dual [84], and Tikhonov explicit regularization, solving each problem by using the forward-backward algorithm [45]. In addition, we compare to another classical optimization algorithm for the minimization of the sum of two non-differentiable functions, namely the Douglas-Rachford algorithm [26]. In the noise free case, this algorithm is very effective in terms of number of iterations, but at each iteration it requires the explicit projection on the feasible set. In the noisy case, a stability analysis of the latter is not available.

We use the four variants of the algorithm PDA corresponding to the different choices of the operators $T$ in Definition 4.4.1 and the version of DPA described in Example 4.4.6. Unless otherwise stated, in all the experiments we use as preconditioners $\quad = \quad = \frac{0.99}{\|X\|}\mathrm{Id}$, which both satisfy (4.2.4).

Let $d = 2260$, $p = 3000$, and let $X \in \mathbb{R}^{d\ p}$ be such that every entry of the matrix is an independent sample from $N(0; 1)$, then normalized column by column. We set $y := X \quad$, where $\in \mathbb{R}^p$ is a sparse vector with approximately 300 non-zero entries uniformly distributed in the interval $[0; 1]$. It follows from [53, Theorem 9.18] that $\quad$ is the unique minimizer of the problem with probability bigger than 0.99. Let $y \quad$ be such that $y \quad = y + \|y\|w$ where the vector $w$ is distributed, entry-wise, as uniformly on $[\ 0.2; 0.2]$. In this experiment, to test the reconstruction capabilities of our method, we use the exact datum to establish the best stopping time, i.e. the one minimizing $\|\ ^k \quad \|$. The exact data solution is also used for the other regularization techniques. In a real practical situation, when both $\quad$ and $\quad$ are unknown, we would need to use parameter tuning techniques in order to select the optimal stopping time, but we do not address this aspect here.

We detail the used algorithms and their parameters below.

(Tik) **Tikhonov Regularization**: We consider a grid of penalty parameters

$$G = \quad 1 \quad \frac{l \quad 1}{5} \quad 10^1 \ ^d\|X^{>}y\|_1 : \ l \in [5]; \ d \in [6]$$

and, for each value $\in G$, the optimization problem

$$\min_{\in\mathbb{R}^p} \quad \| \|_1 + \frac{1}{2}\|X \quad y\|^2 : \tag{4.5.1}$$

We solve each one of the previous problems with 300 iterations of forward-backward algorithm, unless the stopping criterion $\|\ ^{k+1} \quad ^k\| \quad 10^{\ 3}$ is satisfied earlier. Moreover, to deal efficiently with the sequence of problems, we use warm restart [17]. We first solve problem (4.5.1) for the biggest value of $\quad$ in $G$. Then, we initialize the algorithm for the next value of $\quad$, in decreasing order, with the solution reached for the previous one; and so on.

(DR) **Douglas Rachford**: see [26, Theorem 3.1].

(PD) **Primal-dual**: this corresponds to PDA with $m = 1$ and $T_1 = \mathrm{Id}$.

(PDS) **Primal-dual with serial projections**: at every iteration, we compute a serial projection using all the equations of the noisy system, where the order of the projections is given by a random shuffle.

(PDP) **Primal-dual with parallel projections**: Set $m = 1$ and $T_1 = \frac{1}{\|X\|\|X\|_F^2} X^\top (X \quad y)$, see Remark 4.4.3.

(PDL) **Primal-dual Landweber**: Set $m = 1$ and $T_1 = \frac{2}{\|X\|\|X\|^2} X^\top (X \quad y)$.

(PDAL) **Primal-dual Landweber with adaptive step**: Set $m = 1$, and $T_1 = (\quad) X^\top (X \quad y)$, where $(\quad) = \min \left\{ \frac{\|X \quad y\|^2}{\|X^\top (X \quad y)\|^2}; M \right\}$ for $M = 10^6$.

(DPS) **Dual primal with serial projections**: at every iteration, we compute a serial projection over every inequality of $\|X^\top w\|_1 \quad 1$, where the order is given by a random shuffle of the rows of $X^\top$.

| | Time [S] | Iteration | Reconstruction error |
|---|---|---|---|
| Tik | 1.89 | 109 | 3.07 |
| DR | 3.08 | 5 | 5.01 |
| PD | 0.36 | 14 | 3.11 |
| PDS | 1.41 | 11 | 2.58 |
| PDP | 0.35 | 14 | 3.11 |
| PDL | 0.28 | 12 | 2.60 |
| PDAL | 0.27 | 11 | 2.56 |
| DPS | 0.54 | 17 | 2.83 |

Table 4.2: Run-time and number of iterations of each method until it reaches the best reconstruction error. We compare the proposed algorithms with Tikhonov regularization (Tik), Douglas-Rachford (DR), and iterative regularization (PD).

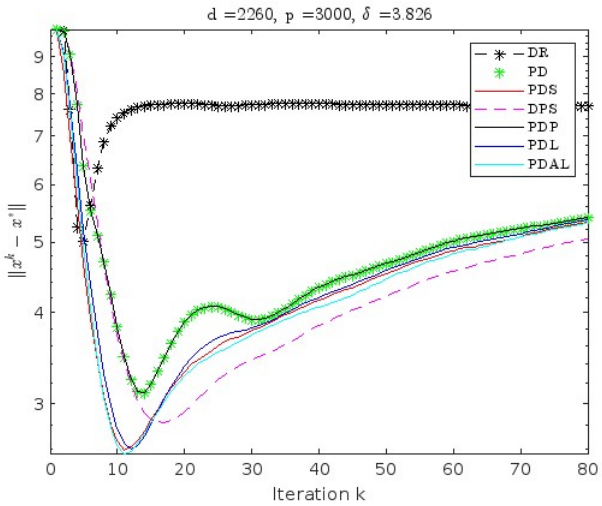


Figure 4.1: Graphical representation of early stopping. Note that the reconstruction error decreases and then increases, since the iterates first approach the exact solution and then converges to the noisy solution.

In Table 4.2, we reported also the number of iterations needed to achieve the best reconstruction error, but it is important to note that the iteration of each method has a different computational cost, so the run-time is a more appropriate comparison criterion.

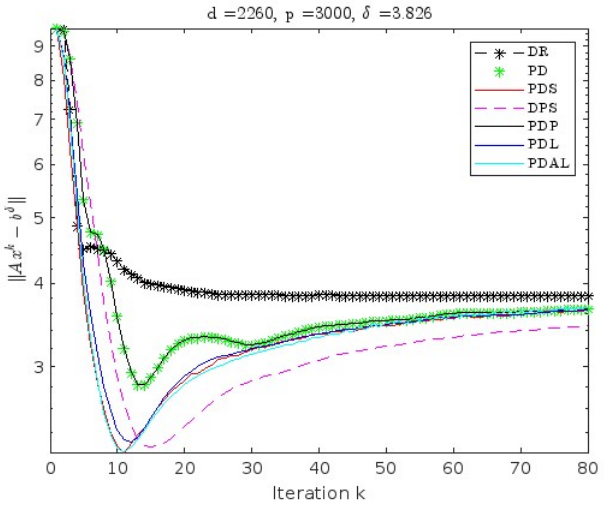


Figure 4.2: Early stopping with respect to the feasibility. Note that their behavior with respect to $k$ is similar to that in Figure 4.1.

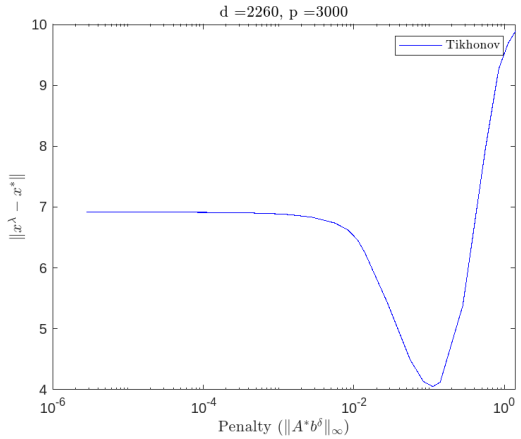


Figure 4.3: Reconstruction error of Tikhonov Method with different penalties.

Douglas-Rachford with early stopping is the regularization method performing worst on this example, both in terms of time and reconstruction error. This behavior may be explained by the fact that this algorithm converges fast (meaning in fews iterations) convergence to the noisy solution, from which we infer that Douglas-Rachford is not a good algorithm for iterative regularization. Moreover, since we project on the noisy feasible set at every iteration, the resolution of a linear system is needed at every step. This also explains the cost of each iteration in terms of time. Note in addition that in our example $y$ is in the range of $X$ and so the noisy feasible set is non-empty. Tikhonov's regularization performs similarly in terms of time, but it requires many more (cheaper) iterations (see Figure 4.3). The achieved error is smaller than the one of DR, but bigger than the minimal one achieved by other methods.

Regarding our proposals, we observe that in Table 4.2 the proposed methods perform better than (PD). This supports the idea that reusing the constraints determined by the data is beneficial with respect to vanilla primal-dual. The benefit is not evident for (PDP), which achieves the worst reconstruction error, since $kXk_F^2$ is very big and so $T_1$ is very close to the identity. All other methods give better results in terms of reconstruction error. On the other hand, (PDS) is the slowest since it requires computing several projections

at each iteration in a serial manner. We also observe that (PDL) and (PDAL) have better performance improving 22.2% and 25.0% in reconstruction error and 16.4% and 17.7% in run-time.

Figure 4.1 empirically shows the existence of the trade-off between convergence and stability for all the algorithms, and therefore the advantage of early stopping. Similar results were obtained for the feasibility gap (see Figure 4.2).

### 4.5.2 Total variation

In this section, we perform several numerical experiments using the proposed algorithms for image denoising and deblurring. As done in the classical image denoising method introduced by Rudin, Osher, and Fantemi in [117], we rely on the total variation regularizer. See also [37, 39, 103, 105, 116, 117, 140]. We compare (PD) with (PDL) and (PDAL) algorithms, which were the algorithms performing the best in the previous application. In this section, we use two different preconditioners, which have been proved to be very efficient in practice [109].

Let $2\ \mathrm{R}^{N^2}$ represent an image with $N\ \ N$ pixels in $[0;1]$. We want to recover from a blurry and noisy measurement $y$, i.e. from

$$y = Kx\ +\ ;$$

where $K$ is a linear bounded blurring operator and is a random noise vector. A standard approach is to assume that the original image is well approximated by the solution of the following constrained minimization problem:

$$\min_{u2\mathrm{R}^{N^2}} fkDuk_{1;2}:\ \ Ku = yg; \tag{TV}$$

Here,

$$k\ \ k_{1;2}:(\mathrm{R}^2)^{N^2}\ !\ \mathrm{R};\quad p\ !\ \sum_{i=1}^{N}\sum_{j=1}^{N} kp_{ij}k;$$

and $D:\mathrm{R}^{N^2}\ !\ (\mathrm{R}^2)^{N^2}$ is the discrete gradient operator for images, which is defined by

$$(Du)_{ij} = ((D_x u)_{ij};(D_y u)_{ij})$$

with

$$(D_y u)_{ij} = \begin{cases} u_{i+1;j}\ \ u_{i;j} & \text{if } 1\ \ i\ \ N\ \ 1 \\ 0 & \text{if } i = N \end{cases}$$

$$(D_x u)_{ij} = \begin{cases} u_{i;j+1}\ \ u_{i;j} & \text{if } 1\ \ j\ \ N\ \ 1 \\ 0 & \text{if } j = N: \end{cases}$$

In order to avoid the computation of the proximity operator of $kD\ \ k_{1;2}$, we introduce an auxiliary variable

$$v = Du\ 2\ Y\ :=\ \ \mathrm{R}^{2\ \ N^2}:$$

Since the value in each pixel must belong to $[0;1]$, we add the constraint $u\ 2\ X\ :=\ [0;1]^{N^2}$. In this way, (TV) becomes

$$\min_{(u;v)2X\ \ Y} (kvk_{1;2}:\ \ Ku = y;\ \ Du = vg: \tag{TV}$$

### Formulation and Algorithms

Problem (TV) is a special instance of (*P*), with

$$\begin{cases} J : \mathrm{R}^{N^2} \quad \mathrm{R}^{2\ N^2} \to \mathrm{R}\ [\ f+1\ g : x := (u;v) \to \|v\|_{1;2} + \ _X(u); \\ A = \begin{matrix} K & 0 \\ D & \mathrm{Id} \end{matrix} \ ; \ y = \begin{matrix} y \\ 0 \end{matrix} \ ; \text{ and } p = d = 3N^2: \end{cases}$$

Clearly, $A$ is a linear non-zero operator, and $R \in \ _0(\mathrm{R}^{N^2} \quad \mathrm{R}^{2\ N^2})$.

| Primal-Dual for total variation |
|---|
| **Input**: $(p^0; p^{\ 1}; \ ^0; v^0) \in \mathrm{R} \ R^{6\ N^2} \quad \mathrm{R}^{2\ N^2}$ and $(q^0; q^{\ 1}; z^0; w^0) \in \mathrm{R}^{3\ N^2} \quad \mathrm{R}^{N^2}$. |
| **For** $k = 1;\dots;$ L: |
| $v^{k+1} = v^k + \ (K(p^k + \ ^k \quad p^{k\ 1})^k \quad y)$<br>$w^{k+1} = w^k \quad (q^k + z^k \quad q^{k\ 1}) + D(p^k + \ ^k \quad p^{k\ 1})$<br>$^{k+1} = P_X(p^k \quad K^{>} v^{k+1} + \ w^{k+1})$<br>$z^{k+1} = \mathrm{prox}_{\ k\ k_{1;2}}(q^k \quad D^{>} w^{k+1})$<br>$p^{k+1} = x^k \quad (x^k)\ K^{>}(Kx^k \quad y) + (Dx^k \quad z^k)$<br>$q^{k+1} = q^k \quad (x^k)D^{>} \ Dx^k \quad z^k$ (4.5.2) |
| **End** |

Table 4.3: General form of the algorithms.

We compare the algorithms listed below. Note that all proposed algorithms are different instances of the general routine described in Table 4.3, and each of them corresponds to a different choice of $(\ ^k)$:

1. PD, the vanilla primal-dual algorithm, corresponding to $(\ ^k) = 0$;
2. PPD, the preconditioned primal-dual algorithm, obtained by $(\ ^k) = 0$ and and as in [109, Lemma 2];
3. PDL, corresponding to $(\ ^k) = 1/\|K\|^2$;
4. PDAL, corresponding to $(\ ^k) = \hat{}(\ ^k)$ as (4.4.3).

Initializing by $p^0 = p^0 = \ ^0$ and $q^0 = q^0 = z^0$, we recover the results of Theorem 4.3.1 and Corollary 4.3.4.

**Remark 4.5.1.** In order to implement the algorithm in 4.5.2, we first need to compute some operators.

1. It follows from [15, Proposition 24.11] and [15, Example 24.20] that

$$\mathrm{prox}_{k\ k_{1;2}}(v) = \ \mathrm{prox}_{k\ k}^{\ i}(v_i)\ _{i=1}^{N^2} = \ 1 \ \frac{}{\max\{\ ; \|v\|_k\}} \ v_i\ _{i=1}^{N^2};$$

   where $v_i \in \mathrm{R}^2$. The projection onto $X$ can be computed as

$$P_X(u) = \ P_{[0;1]}(u_i)\ _{i=1}^{N^2};$$

where $P_{[0;1]}(u_i) = \min\{1; \max\{u_i; 0\}\}$.

2. It follows from [37] that

$$D^{>}p = \operatorname{div} p = \begin{cases} (p_1)_{i;j} - (p_1)_{i-1;j} & \text{if } 1 < i < N \\ (p_1)_{i;j} & \text{if } i = 1 \\ -(p_1)_{i-1;j} & \text{if } i = N \end{cases} + \begin{cases} (p_2)_{i;j} - (p_2)_{i;j-1} & \text{if } 1 < j < N \\ (p_2)_{i;j} & \text{if } j = 1 \\ -(p_2)_{i;j-1} & \text{if } j = N. \end{cases}$$

### Numerical results

Set $N = 256$, and let be the image "boat" in the library Numerical tours [108]. We suppose that $K$ is an operator assigning to every pixel the average of the pixels in a neighborhood of radius 8 and that s $U([-0.025, 0.025])^{N^2}$. We use the original image as exact solution. For denoising and deblurring, we early stop the procedure at the iteration minimizing the mean square error (MSE), namely $\|x^k - x\|^2/N^2$, and we measure the time and the number of iterations needed to reach it. Another option for early stopping could be to consider the image with minimal structural similarity (SSIM). Numerically, in our experiments, this gives the same results. Additionally, we use the peak signal-to-noise ratio (PSNR) to compare the images. Note that the primal-dual algorithm with preconditioning is the method that needs less time and iterations among all procedures. Moreover, due to [40, Lemma 2], the condition (4.2.4) is automatically satisfied, while for the other methods we need to check it explicitly, which is computationally costly. However, (PPD) is the worst in terms of SSIM, PNSR, and MSE. We verify that all other algorithms have a superior performance in terms of reconstruction, with a small advantage for the Landweber with fixed and adaptive step-sizes, reducing the MSE of 94% with respect to the noisy image. In addition, compared to (PD), the algorithms (PDL) and (PDAL) require less iterations and time to satisfy the early stopping criterion. We believe that this is due to the fact that the extra Landweber operator improves the feasibility of the primal iterates. Visual assessment of the denoised and deblurred images are shown in Figure 4.4, which highlights the regularization properties achieved by the addition of the Landweber operator and confirms the previous conclusions.

| | Iterations | Time | SSIM | PNSR | MSE |
|---|---|---|---|---|---|
| Noisy image | - | - | 0.4468 | 21.4801 | 0.0071 |
| PD | 54 | 8.9773 | 0.8928 | 32.3614 | 0.0006 |
| PD (precondition) | 5 | 1.5515 | 0.8581 | 27.3753 | 0.0018 |
| PDL | 46 | 7.1846 | 0.9066 | 34.2174 | 0.0004 |
| PDAL | 31 | 5.4542 | 0.9112 | 34.3539 | 0.0004 |

Table 4.4: Quantitative comparison of the algorithms in terms of Structural similarity (SSIM), peak signal-to-noise ratio (PSNR), Mean square error (MSE), time, and iterations to reach the early stopping.

## 4.6 Conclusion and Future Work

In this chapter we studied two new iterative regularization methods for solving a linearly constrained minimization problem, based on an extra activation step reusing the data constraints. The analysis was carried out in the context of convex functions and worst-case deterministic noise. We proposed five instances of our algorithm and compared their numerical performance with state-of-the-art methods, and we observed considerable improvement in run-time.

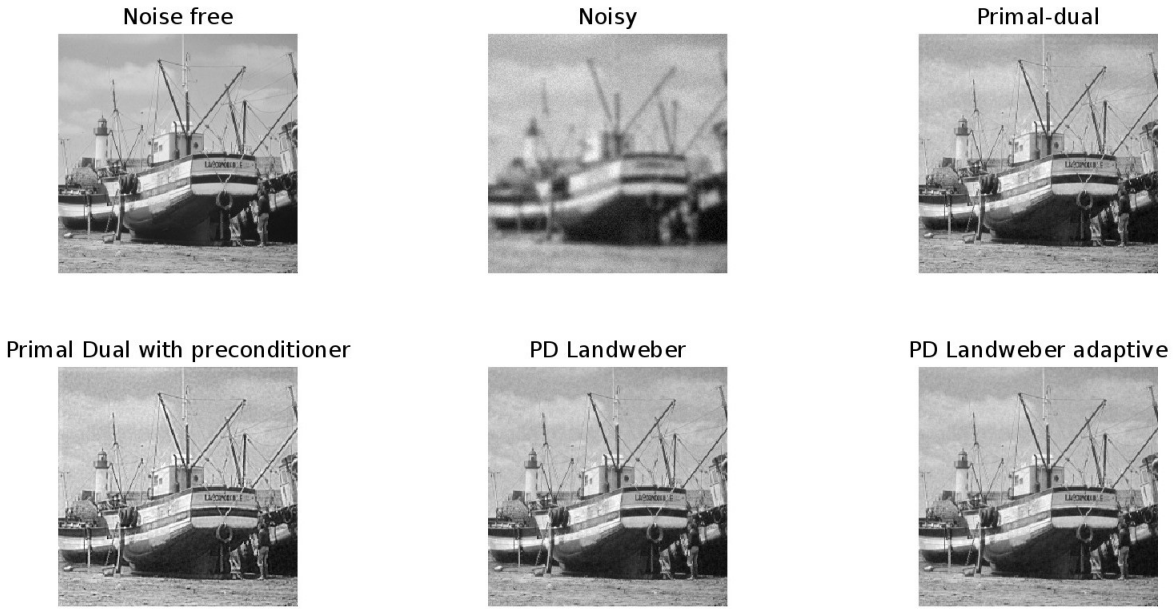


Figure 4.4: Qualitative comparison of the 4 proposed methods.

In the future, we would like to extend Theorem 4.3.1 to structured convex problems and other algorithms. Possible extensions are: 1) the study of problems including, in the objective function, a $L$-smooth term and a composite linear term; 2) the analysis of random updates in the dual variable (see [38]) and stochastic approximations for the gradient; 3) the theoretical study of the impact of different preconditioners; 4) the improvement of the convergence and stability rates for strongly convex objective functions.

# CHAPTER 5

# Implicit regularization and reparameterization

## 5.1 Reparameterization

The second part of this thesis is based on the recent success of overparameterization in machine learning trained with gradient descent. Instead of designing an algorithm with a fixed bias and loss, as we did in the first part of this thesis, we change our approach by reparametrizing the linear model by fixing the loss and the algorithm (which will be a gradient flow for simplicity). So, our goal is to find the implicit bias introduced by the selected optimization method and the reparametrization that has been introduced.

The map

$$2 \mathrm{R}^k \; 7! \quad = q( \ ) \; 2 \mathrm{R}^p$$

is the reparameterization. Typically, $k \quad p$, and in this view, the model becomes overparameterized. The key idea is that the sequence $(t)$ obtained with gradient descent defines a sequence $(t) = q( \ (t))$, which corresponds to some suitable optimization procedure to find under a bias $R$. Then, the reparameterization $q$ defines a corresponding bias $R$.

**Then, for a given pair of models and corresponding reparameterization, the question is: what are the associated optimization procedures and corresponding biases?**

In the following, we will tackle the above question, considering a simplified setting and recovering and extending a number of recent studies.

Further, we will consider a restricted set of models that are amenable to study; namely, we will assume that $f$ is some kind of linear neural network. In particular, we will consider the following networks:

- **Two layers diagonal network**: $= q( ) = \ _1 \ _2$, where $= ( _1; \ _2) \; 2 \mathrm{R}^p \quad \mathrm{R}^p;$ and $k = 2p$.
- **Deep diagonal networks**: $= q( ) = \ ^L$, where $2 \mathrm{R}^p$, $L \quad 2$, and $k = p$.
- **Multi-neuron fully linear network of depth 2**: $= q( ) = Ww$, where $W$ is a $p \quad d$ matrix, $w \; 2 \mathrm{R}^d$, $= (W; w)$, and $k = (p + 1)d$.
- **Fully connected normalized linear network of depth 2**: $= q( ) = Ww$, where $W$ is a $p \quad d$ matrix, $w$ is a unitary vector in $\mathrm{R}^d$, $= (W; w)$, and $k = (p + 1)d$.

- **Weight normalized networks**: $= h( )w$, where $h$: dom($h$) R 7! $R_+$ is a given function, $w$ is a unitary vector in $R^p$, $= ( ;w)$ 2 dom($h$) $R^p$, and $k = p + 1$.

### 5.1.1 Optimization by reparameterization

Let $L : R^p$ ! R be a convex functional, which can be interpreted as the data fit term. Let $q : R^k$ ! $R^p$, be a differentiable map, seen as the reparameterization. For some suitable initialization, consider the flow

$$-(t) = r L(q( (t))): \tag{5.1.1}$$

Instead of gradient descent, we consider the gradient flow, where we indicate for clarity the variable with respect to which the gradient is taken. The questions are:

1. If there exists a bias functional $R$ : $R^p$ ! R such that (t) = $q$( (t)) converges in a suitable sense to some 1 solving

$$\min_{2S} R( ); \quad S = \operatorname*{argmin}_{2R^p} L( ): \tag{5.1.2}$$

2. If it is possible to show that the trajectory followed by (t) corresponds to some optimization approach to solve (5.1.2).
3. If it is possible to find an explicit expression of $R$ in terms of $q$ and the initialization.

Recent research has partially addressed these questions by establishing a link between gradient flow and mirror flow. The idea can be simply summarized as follows: Starting from the reparameterization = $q$( ) and equation (5.1.1), a simple derivation of the composition shows that

$$- = J_q( )- = J_q( )r L(q( )) = J_q( )J_q( )^> r L( );$$

where we denote by $J_q$ the Jacobian of $q$ with respect to its variable. So, under the assumption that there exists a function $F$ : dom($F$) ! R and $G$: dom($G$) ! R such that, for every 2 $R^k$,

$$J_q( )J_q( )^> = G(q( )) r^2F(q( ))^{1}; \tag{5.1.3}$$

we get that the variable (t) follows the following generalized ("time-warping") mirror flow:

$$-(t) = G( (t)) r^2F( (t))^{1} r L( (t)): \tag{5.1.4}$$

The previous computations can be understood as follows: Parameterizing = $q$( ) and using gradient flow on is equivalent to applying "time-warping" mirror flow on , with some specific entropy $F$ depending on the reparameterization $q$. The "time-warping" function can be interpreted as a non-linear preconditioner of the dynamical system. The flow on (5.1.4) has an implicit bias $R$ towards a specific solution in the set of minimizers of the loss function. This bias, as we will see in the following, is related to $F$ and $G$, and so to the reparameterization $q$.

## 5.2 Case $G( )$ 1: Reparameterizing gradient flow as mirror flow.

In the case when $G( )$ 1, the dynamical system (5.1.4) is indeed a mirror flow. In [4] the authors show that mirror flow [3, 16, 99] on the global variable is equivalent to

use gradient flow over the reparameterization, assuming the existence of a suitable mirror map. This map allows for characterizing the implicit bias of the optimization process for a given model. The equivalence between mirror flow on and gradient flow over was formalized in the following theorem:

**Theorem 5.2.1.** [4, Theorem 2] Let $F$ be a strictly convex, continuously-differentiable function with domain in $\mathbb{R}^p$. Let $q\colon \mathbb{R}^k \to \mathbb{R}^p$ be a reparameterization function, such that $k \geq p$, expressing parameters of $F$ uniquely as $q(\ )$ and $\mathrm{ran}(q) \subseteq \mathrm{dom}(F)$. Moreover, assume that

$$J_q(\ )J_q(\ )^\top = \nabla^2 F(\ )^{-1};$$

for all $= q(\ )$. Then, the Mirror flow update on parameter for the convex function $F(\ )$ and loss $L(\ )$,

$$\dot{}(t) = -\nabla^2 F(\ (t))^{-1} \nabla L(\ (t));$$

coincides with the gradient flow update on parameters for the the composite loss $L \circ q$,

$$\dot{}(t) = -\nabla L(q(\ (t)));$$

provided that $(0) = q(\ (0))$.

**Remark 5.2.2.** In the special case where $p = k$ and $q$ is separable, meaning that the $i$-th component of $q$ depends only on the corresponding component of , (5.1.3) can be reduced to the following ordinary differential equation:

$$\nabla^2 F(q(\ ))^{-1} = \mathrm{Diag}\left(q'(\ )^{-2}\right); \tag{5.2.1}$$

In the appendix, Example A.1.1 presents the explicit computations for the case where the reparameterization is separable and finding the mirror map is trivial, demonstrating that the mirror map can be straightforwardly determined by solving Equation (5.2.1).

In practice, for many models, find $F$ from (5.1.3) is not straightforward, and the resulting bias $R$ is unclear. The first technical step is to express the product of the Jacobians $J_q(\ )J_q^\top(\ )$ in terms of the variable $= q(\ )$ (see Example A.1.2 in the appendix). The second is then to find a function $F(\ )$ such that the inverse of its Hessian is equal to the product of the Jacobians (5.1.3).

While the previous theorem sheds light on certain aspects of how to find implicit bias, it does not tackle the existence of a mirror map. Moreover, even if such a map exists, uniqueness is not guaranteed, and finding the parameterization can be a challenging task. However, in [77] the previous results are generalized, providing conditions for the existence of the parameterization. Moreover, the convergence to a feasible point is also proved, whereas in previous works it was only assumed.

Before to present the existence result, we recall some definitions presented in Chapter 2

**Definition 5.2.3.** A regular parameterization $q\colon \mathbb{R}^k \to \mathbb{R}^p$ is a $C^1$ parameterization such that $J_q(\ )$ is of rank $p$ for all $\in \mathbb{R}^p$.

**Definition 5.2.4.** A $C^2$ parameterization $q : \mathbb{R}^k \to \mathbb{R}^p$ is commuting if and only if for any $i, j \in [p]$, we have that,

$$\nabla^2 q_j(\ )\nabla q_i(\ ) - \nabla^2 q_i(\ )\nabla q_j(\ ) = 0;$$

for all $\in \mathbb{R}^k$.

**Definition 5.2.5.** For any $C^1$ function $f : \mathbb{R}^k \to \mathbb{R} \cup \{+\infty\}$, we denote by $\phi_f^t(\theta_0) = \theta(t)$, where $\theta(t)$ is the solution at time $t$ (when it exists) of

$$\begin{cases} \dot{\theta}(t) = -\nabla f(\theta(t)), & t > 0; \\ \theta(0) = \theta_0 \in \mathbb{R}^k. \end{cases}$$

We say $\phi_f^t(\theta_0)$ is well-defined at time $t$ when the above differential equation has a solution at time $t$.

**Definition 5.2.6.** Given a $C^2$ parameterization $q : \mathbb{R}^k \to \mathbb{R}^p$, for any $\theta_0 \in \mathbb{R}^p$ and $t \in \mathbb{R}^p$, we define

$$\Phi(\theta_0; t) := \phi_{q_1}^{t_1} \circ \cdots \circ \phi_{q_p}^{t_p}(\theta_0),$$

when it is well-defined, i.e., the corresponding differential equations have a solution. For any $\theta_0 \in \mathbb{R}^p$, we define the domain of $\Phi(\theta_0, \cdot)$ as:

$$U(\theta_0) = \{t \in \mathbb{R}^p \mid \Phi(\theta_0; t) \text{ is well defined}\}.$$

**Definition 5.2.7.** For any parameterization $q : \mathbb{R}^k \to \mathbb{R}^p$ in $C^2$ and for any function $L : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ in $C^1$, given any starting point $\theta_0 \in \mathbb{R}^p$, we define the reachable set $\mathcal{R}(\theta_0; q)$ as

$$\mathcal{R}(\theta_0; q) = \{\phi_{L \circ q}^t(\theta_0) \mid t > 0\}.$$

Now, we present sufficient assumptions for the reparametrization, presented in [77], which ensure the existence of a mirror map and allow its implicit bias to be characterized.

**Assumption 5.2.8.** [77, Assumption 3.5.] Let $q : \mathbb{R}^k \to \mathbb{R}^p$ be a parametrization. We assume that for any $\theta_0 \in \mathbb{R}^p$ and $i \in [p]$, $\phi_{q_i}^t(\theta_0)$ is well-defined for $t \in ]T_-, T_+[$ such that either $\lim_{t \to T_+} \|\phi_{q_i}^t(\theta_0)\| = +\infty$ or $T_+ = +\infty$ and similarly for $T_-$. Also, we assume that for any $\theta_0 \in \mathbb{R}^p$ and $i, j \in [p]$, we have that, $\phi_{q_i}^s \circ \phi_{q_j}^t(\theta_0)$ is well-defined if and only if $\phi_{q_j}^t \circ \phi_{q_i}^s(\theta_0)$ does, for every $t, s > 0$.

The following intermediate lemma states that the point reached by the gradient flow with any commuting parameterization is determined by the integral of the negative gradient of the loss along the trajectory.

**Lemma 5.2.9.** [77, Lemma 4.7.] Let $q : \mathbb{R}^k \to \mathbb{R}^p$ be a commuting parametrization. For any initialization $\theta_0 \in \mathbb{R}^k$, consider the gradient flow:

$$\dot{\theta}(t) = -\nabla (L \circ q)(\theta(t)) \qquad \theta(0) = \theta_0.$$

Further define $\gamma(t) = \int_0^t \nabla L(q(\theta(s)))ds$. Suppose $\gamma(t) \in U(\theta_0)$ for all $t \in [0, T)$ where $T \in \mathbb{R} \cup \{+\infty\}$, then it holds that $\theta(t) = \Phi(\theta_0; \gamma(t))$ for all $t \in [0, T)$.

Now, we present the existence of a mirror map corresponding to a specified parameterization.

**Theorem 5.2.10.** [77, Lemma 4.8] Let $q : \mathbb{R}^k \to \mathbb{R}^p$ be a commuting and regular parametrization satisfying Assumption 5.2.8. Then for any $\theta_0 \in \mathbb{R}^k$, there exists a Legendre function $Q : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ such that $\nabla Q(\cdot) = q(\Phi(\theta_0; \cdot))$ for all $\cdot \in U(\theta_0)$. Moreover, let $F$ be the convex conjugate of $Q$, then $F$ is also a Legendre function and satisfies that $\text{int}(\text{dom}(F)) = \mathcal{R}(\theta_0; q)$ and

$$\nabla^2 F(q(\Phi(\theta_0; \cdot))) = J_q(\Phi(\theta_0; \cdot))J_q^T(\Phi(\theta_0; \cdot))^{-1},$$

for all $\cdot \in U(\theta_0)$.

The next theorem characterizes the implicit bias for mirror flow when the loss function is the composition of a convex function with a linear operator. This implicit bias corresponds to the Bregman distance between the set of minimizers of the loss and the initialization, extending the results presented in [7, 139].

**Theorem 5.2.11.** [77, Lemma 4.16] Let $q\colon \mathbb{R}^k \to \mathbb{R}^p$ be a commuting and regular parametrization satisfying Assumption 5.2.8. Let $X \in \mathbb{R}^{d\ p}$, let $y \in \mathbb{R}^d$, let $\ell\colon \mathbb{R}^d \to \mathbb{R}$ be differentiable with a locally Lipschitz gradient. Assume that $L = \ell \circ X$ and $S = \{\ \in \mathbb{R}^p \mid X\ = y\}$. Consider the gradient flow given by (5.1.1). Then, there exists function $F$ given Theorem 5.2.10 and a solution $_1 \in \mathbb{R}^P$ such that $_1 = \lim_{t\to\infty} (t)$ and

$$_1 \in \operatorname*{argmin}_{z\in S} F(z) - \langle \nabla F(\ _0), z\rangle = \operatorname*{argmin}_{z\in S} D_F(z,\ _0).$$

Although [77] encompasses many of the existing parameterizations, there are also interesting reparameterizations for which a function $F$ such that (5.1.3) holds does not exist if $G(\ ) \neq 1$, making it necessary to consider $G \neq 1$. In this case, the dynamic cannot be expressed as vanilla mirror flow,but the flow can still be expressed as a mirror flow multiplied by a positive scalar function $G$, as it was studied in [7] for the case of a multi-neuron fully connected linear network.

## 5.3 Case $G(\ ) \neq 1$: Time warping Mirror Flow

If $G(\ ) \neq 1$, the dynamical system (5.1.4) is a time-warped mirror flow. In this case, the dynamics cannot be described by vanilla mirror flow; however, an entropy function can still be found, as was studied in [7] for multi-neuron fully connected linear networks. This flow can be understood as mirror descent with a non-linear time-warping, which gives degrees of freedom with respect to [77] on how to choose $F$. This map $G$ allows for characterizing the implicit bias of the optimization process for a given model.

Consider, for instance, the following two leading examples along the this thesis.

**Example 5.3.1** (Multi-neuron fully connected linear network of depth 2)**.** In linear networks of depth 2, the reparameterization is given by $= q(\ )$ with $= (W; w) \in \mathbb{R}^{p\ m} \times \mathbb{R}^m$ and $q(\ ) = Ww$. The gradient flow on the reparameterization is then given by

$$\begin{aligned} \dot{W}(t) &= -\nabla_W L(W(t)w(t)) = -\nabla L(\ (t))w^\top(t); \\ \dot{w}(t) &= -\nabla_w L(W(t)w(t)) = -W^\top(t)\nabla L(\ (t)); \end{aligned} \tag{5.3.1}$$

where we suppose, for simplicity in the computations, that the initialization satisfies $W^\top(0)W(0) = w(0)w^\top(0)$. In this case, by Lemma 2.4.5, it is possible to prove that there is no function $F$ such that (5.1.3) holds with $G(\ ) = 1$. On the other hand, in the original variable , we obtain the following dynamical system:

$$\nabla^2 F(\ (t))\ \dot{}(t) = -G(\ (t))\nabla L(\ (t)); \tag{5.3.2}$$

where

$$F(\ ) = (2/3)k\ ^{3/2} \text{ and } G(\ ) = k\ ^{1/2}.$$

For the proofs of the statements above, see Section A.1.3 in the appendix. Note that, due to the presence of the function $G$, (5.3.2) is not a mirror flow but a generalization of it.

**Example 5.3.2** (Standard weight normalization)**.** Consider the reparameterization given by $= q( )$ with $= (w; ) 2 \mathrm{R}^p \quad \mathrm{R}$ and $q( ) = \frac{w}{kwk}$. The gradient flow on the reparameterization is then given by

$$\underline{w}(t) = \quad r_w L( (t)) = \frac{(t)}{kw(t)k} \mathrm{Id} \quad \frac{w(t)w^{>}(t)}{kw(t)k^2} \quad r L( (t))$$

$$\_(t) = \quad r \quad L( (t)) = \quad \frac{w(t)}{kw(t)kj} j \; r L( (t)) \; :$$

with initialization $(w_0; \ _0) \; 2 \; \mathrm{R}^p \quad \mathrm{R}$ such that $kw_0k = 1$ and $_0 > 0$. In this case, by Lemma 2.4.5, it is possible to prove that there is no function $F$ such that (5.1.3) holds with $G( ) = 1$. On the other hand, in the original variable , we obtain the following dynamical system:

$$r^2 F( (t)) \; \_(t) = \quad G( ) r L( (t));$$

where $G( ) = k \; k \exp \frac{k \; k^2}{2}$ and $F$ is a function that cannot be expressed explicitly. For the proofs of the statements above, see Example 6.3.6 with $L = 1$.

The above examples were presented in [7] and [89, 118, 139]. However, the convergence of the trajectory is assumed but not proven (see [7, 89, 139]). In the next chapter, we study the well-posedness, convergence of the iterates, and convergence in value for time-warped mirror flow applied to convex losses. Additionally, when the loss function consists of a strictly convex function composed with a linear operator, we give an explicit expression of its implicit bias.

# CHAPTER 6

# Learning from data via overparameterization

**Abstract**

The goal of machine learning is to achieve a good prediction exploiting training data and some a-priori information about the model. The most common methods to achieve the last objective are explicit and implicit regularization. In the first technique, a regularizer is explicitly introduced to find, among all the solutions, a good generalizing one. The second technique, i.e. implicit regularization, is based on the inductive bias intrinsically induced by the specific method used to optimize the parameters involved.

Recently, the success of learning is related to re- and over-parameterization, that are widely used - for instance - in neural networks applications and the optimization method used. However, there is still an open question of how to find systematically what is the inductive bias hidden behind the model for a particular optimization scheme. The goal of this chapter is taking a step in this direction, studying extensively many reparameterizations used in the state of the art and providing a common structure to analyze the problem in a unified way. We show that gradient descent on the empirical loss for many reparameterizations is equivalent, in the original problem, to a generalization of mirror descent. The mirror function depends on the reparameterization and introduces an inductive bias, which plays the role of the regularizer. Our theoretical results provide asymptotic behavior and convergence in the simplified setting of linear models.

**Keywords.** Overparameterization, Implicit Regularization, Time-warping Mirror Flow, Fully connected normalized linear networks, Weight normalization.

**AMS Mathematics Subject Classification (2020):** 34A55, 90C25, 65K10.

In this chapter, we study a unified framework observed in various reparameterizations presented in the state of the art, called time-warp mirror flow given in equation (6.2.1). We provide a complete analysis consisting of several steps. Firstly, in Section 6.2, we establish conditions for the well-posedness of the dynamical system. Then, we show that for any convex function, the sequence converges to a stationary point that minimizes the loss function and avoids the extra stationary points that are produced by the reparameterization. For the specific case of a function composed with a linear operator, an implicit bias is provided. Next, these results will be used in Section 6.3, where we look at weight normalization techniques for functions $F$ and $G$ that only depend on the norm of . These functions are also called radial functions. Furthermore, we provide a criterion for determining a suitable weight normalization parameterization for a given function that depends only on the norm. Finally, we explore the flexibility of our formulation by applying the previous results to different examples related to weight normalization. Finally, in Section 6.4, we conclude this chapter with some remarks and future works.

Compared to the previous chapter, the main difference is that, for the given reparameterization, we assume the existence of $F$ and $G$. Unlike in Theorem 5.2.10, where conditions for the existence of such reparameterizations are provided and analyzed, our purpose here is not to study these conditions. Instead of focusing on reparameterization, we examine the convergence properties of the time-warping mirror flow and its implicit bias, as outlined in Theorem 5.2.11.

## 6.1 Introduction

Classic algorithm design in machine learning (ML) is based on fitting some chosen model to data while including some bias reflecting prior knowledge on the problem [64]. In fact, recent studies have shown that large overparameterized models can achieve excellent learning performance, even without enforcing any explicit bias [58, 59, 77, 88, 93, 102, 131, 135]. A possible explanation is that the bias is implicitly incorporated through the selection of the model and the optimization procedure used. However, uncovering such a bias is typically a complex challenge [5, 7]. An idea is to view the overparameterization as a reparameterization of some simpler, original model. This raises the question if the optimization of the over- or re-parameterized model has a clearer interpretation in the original model. A natural starting point for exploring the concept of overparameterization is to consider linear networks, which can be seen as overparameterizations of linear models. Indeed, recent results have started considering so-called diagonal networks and have shown that gradient flow for these models corresponds to a mirror flow for the original model [4, 77]. In this context, the bias is linked to a sparsity prior imposed on the linear model [43, 135, 147]. As observed in [7], a more general form of mirror flow, including a time-warping factor, allows for a much more general treatment. In this paper, we develop this latter line of work. First, we provide a general analysis of time-warped mirror flow. In particular, we study its convergence both in the value of the loss function and in the iterations. Additionally, we provide rates of convergence and characterize the implicit bias of the limit point. Second, we discuss in detail the case of one-hidden-layer linear networks and study the effect of different types of weight normalization. Our results show that for a given radial function, i.e., a function that only depends on the norm, we can find a suitable parameterization such that the implicit bias is given by the radial function. We note that weight normalization has been previously considered in [44, 89, 118, 139], but using the framework of time-warping mirror flow, we explore more general reparameterizations. To the best of our knowledge, the analysis of general linear one-hidden-layer neural networks is new.

### 6.1.1 Related work

In this subsection, we will briefly analyze the literature most related to this chapter.

**Reparameterizing gradient flow as mirror flow**: The first paper relating gradient flow on the reparametrization to mirror flow [3] on the original variable is [4]. The relation between the two flows holds under quite restrictive assumptions on the reparametrization, which ensure the existence of a mirror map. This map allows for characterizing the implicit bias of the optimization process for a given reparametrization. The existence of the mirror map and its analytic form given a reparametrization is not addressed in [4], neither the question of establishing whether different reparametrizations lead to the same implicit bias. A further step has been taken in [77]. This is the only paper where existence and convergence of a solution of the mirror flow is discussed. In addition, conditions on a parametrization are given for the existence of the mirror map. Although [77] encompasses many of the existing linear reparameterizations, there are still examples that cannot be cast under its setting, such as matrix times vector overparametrizations and weight normalization. In these two examples, the dynamic in the original variable cannot be expressed as a mirror flow. The paper [7] proposed a generalized version of mirror flow, that is the one analyzed in this paper, for the case of matrix times vector overparametrization. Their interpretation proposes to relate gradient descent on the reparametrization to mirror descent with a non-linear time-warping in the original variable. This gives more degrees of freedom with respect to [77] and allows to cover in principle a more general class of reparametrizations.

**Diagonal linear neural network using least square:** Deriving an Implicit bias for any general network is still not well understood. However, this problem has been addressed for simpler models. In [43, 135, 147], it is shown that for the simplest model, a diagonal linear neural network with depth $L$ (where each component is raised to the power of $L$), vanilla gradient flow approximates the minimal $\ell_1$-norm solution. However, to achieve a good approximation, this method requires an initialization close to zero, which in practice makes the convergence slower. Similar results for stochastic gradient flow were obtained in [2, 106], where it is proved that slower convergence implies better bias. The main advantage obtained with respect to the deterministic approach is that, with the same initialization, the sequence generated by stochastic gradient flow achieves greater sparsity than the one obtained by vanilla gradient flow.

**Implicit bias of least square**: The bias of different first-order optimization algorithms and the effect of the step-size on the optimization process have been studied in [58, 138], [76], and [93], respectively. More general models have been addressed in [7, 49, 130], where the case of a fully connected linear neural network and a two-Layer Single (Leaky) ReLU neuron is studied. In [44], inspired by the weight normalization technique proposed in [118], which involves decoupling a vector into its norm and its direction, a diagonal linear network is combined with weight normalization, providing a robust implicit bias for large scale initialization. While in [139] is studied the inductive bias for the squared loss function using weight normalization in the discrete and continuous settings. Moreover, the authors also proved the convergence of the sequence generated by the algorithm to the minimal norm solution among all the solutions of least squares. In this chapter, we generalize this result by studying the effect of the choice of different reparameterizations of the norm, including classical and exponential weight normalization, among others. Furthermore, we give conditions to find a suitable weight normalization reparameterization for functions that depend on the norm, including the analysis of its convergence and the characterization of its implicit bias.

**Implicit bias of logistic loss**: Similar results have been obtained by applying gradient descent to the logistic loss function, which is commonly used in classification problems. In [67, 68, 92, 125], it was proved that using gradient descent, the linear classifier converges to the direction of $L_2$ maximum margin. This result was later extended for SGD in [94]. For the case of Diagonal linear neural networks, the same result was obtained in [88] when the initialization tends to infinity. The implicit bias of the weight normalization for the logistic loss was also studied in [89]. The inductive bias of a 2-layer infinitely wide ReLU neural network (i.e., a neural network with a large number of hidden units), it was studied in [42]. In [79] the implicit regularization of the gradient descent algorithm using the exponential loss function is studied. This work encompasses different types of parameterization of homogeneous neural networks, including fully connected and convolutional neural networks with ReLU or LeakyReLU activations, and proves the convergence to a minimal norm solution subject to margin constraints. The same technique was used in [59], for a full-width linear convolutional network applying gradient descent.

**Deep Matrix Factorization**: Classic theory also covers how to induce a bias using matrices. A common approach is to use matrix factorization, which is a powerful tool for simplifying its representation by finding the underlying structure of a matrix. A well-known method to induce low rank for semi-definite positive symmetric matrices is factorization, whereby a matrix is expressed as the product of a matrix by its transpose. This technique is useful to find a low-rank solution since, with the dimension of the parameterization, we can impose an additional rank constraint. In [60, 75] it was demonstrated that the factorization reaches the minimal nuclear norm for a sufficiently small initialization, assuming that the sensing matrices are commutative and semi-positive definite. The generalization for the multiplication of more than two matrices is in [5], which shows that gradient flow can act as a preconditioner that prefers the direction of the larger eigenvalues. The algorithmic bias of using mirror descent was studied in [134, 137], and the generalization for a tensor formulation was studied in [145]. The case where the sensing matrices satisfy the restricted isometry property was studied in [131, 136, 144]. A simplified version of a neural network is the linear neural network, i.e., when all the activation functions are the identity. This was studied in [6, 55, 56, 80].

## 6.2 Global existence and asymptotic analysis

In the setting of Section 5.1.1, given a differentiable convex loss function $L : \mathbb{R}^p \to \mathbb{R}$, $\theta \mapsto L(\theta)$, we consider the trajectory $\theta(t)$ generated by the gradient flow dynamics on a reparameterization $\theta = q(w)$. We analyze the case in which the gradient flow on the reparameterization $w$ may not be expressed as a mirror flow on the original variable $\theta$; but it can still be written as a dynamical system involving the inverse of the Hessian of $F$ and a positive function $G$, as in (5.3.2):

$$\begin{cases} \dot{w}(t) = -G(\theta(t))\, \nabla L(\theta(t)), & t > 0 \\ w(0) = w_0 \in \mathbb{R}^p. \end{cases} \tag{6.2.1}$$

The positive function $G$, already introduced in [7], can be interpreted as a non-linear "time-warping". In the case when $G(\theta) \equiv 1$, the dynamical system (6.2.1) is indeed a mirror flow.

We present a comprehensive framework to study the existence and uniqueness of the trajectory of (6.2.1) and the its minimization properties, such as the decrease of the loss'

values along it and the corresponding convergence rates (see Theorems 6.2.4 and 6.2.8). Moreover, we show the convergence of the trajectory to $S = \operatorname{argmin} L$ and we explicitly compute its implicit bias towards a specific solution in $S$ (see Theorem 6.2.8). As our analysis shows, with respect to the classical mirror flow setting, the presence of the function $G$, when supposed to be positive, does not affect the fact that the loss is a non-increasing function in time, neither the fact that the trajectory convergences to the set of minimizers of the loss. On the other hand, it plays a role in the selection of the specific solution to which the trajectory is converging. The techniques used to study the mirror flows used in this chapter are presented in [3]. However, the main difference between their work and ours is that they focus on mirror flows in the context of constrained minimization problems, while in our work, the constraints are encoded within the loss function itself.

Differently from previous works [6, 7, 77], here we focus on the well-posedness and convergence over time of the dynamical system described by Equation (6.2.1). Instead of assuming the convergence of $(t)$ to a minimizer of the loss function as an hypothesis, as in [6, 7], we prove this fact. Additionally, we derive the implicit bias induced by reparametrization for a general convex loss function $L$, while in previous work only the least squares loss function has been considered. For the least squares loss, we recover that the implicit bias $R$ is the Bregman divergence of the entropy $F$ as in [4, 77] (see Corollary 6.2.9).

### 6.2.1 Well-posedness

In this section, using techniques developed in [3], we show that the system (6.2.1) is well-posed by proving the existence of a unique solution that is defined in the entire time-interval $[0; +1)$. To do so, we first list the assumptions that we require on the loss function $L$ (see A1), the initialization $(0) = {}_0$ (see A3), the mirror map $F$ and the time-warping function $G$ (see A2, A4, and A5). These hypotheses are also useful to prove, in Theorem 6.2.8, the convergence of the global variable $(t)$ to a specific minimizer of the loss function.

**Assumption 6.2.1.**

A1 $L : \mathbb{R}^p \to \mathbb{R}$ is a convex and differentiable function such that $\nabla L$ is locally Lipschitz continuous on $\mathbb{R}^p$ and $S = \operatorname{argmin}_{\in \mathbb{R}^p} L(\ ) \neq ;$.

A2 There exists an open set $U \subseteq \mathbb{R}^p$ such that $S \setminus U \neq ;$, the function $F : \operatorname{dom}(F) \to \mathbb{R}$ is strictly convex and twice-differentiable in $U$, and the mapping $\in U \mapsto [\nabla^2 F(\ )]$ is invertible with locally Lipschitz inverse. The function $G : U \to \mathbb{R}_+$ is locally Lipschitz continuous. Moreover, one of the following two conditions is satisfied:

- $U := \mathbb{R}^p \setminus \{0\}$;
- $F$ is Legendre on $U = \operatorname{int}(\operatorname{dom}(f))$ (see Definition 2.4.7 in the Appendix).

A3 The initialization satisfies ${}_0 \in U$. If $U = \mathbb{R}^p \setminus \{0\}$, we additionally suppose that $L({}_0) < L(0)$.

A4 Set

$$r = \frac{L(0) \quad L({}_0)}{2k \nabla L(0)k} > 0:$$

There exists a constant $C > 0$ such that, for all $\in B(0; r)$, $G( ) \quad C$.

A5 There exists $z \in S \setminus U$ such that the function $D_F(z; )$ is coercive (see (2.4.1) for the expression of the Bregman divergence $D_F$).

**Remark 6.2.2.** Assumption A3 implies in particular that $0 \in S$.

**Remark 6.2.3.** Assumption 6.2.1 is key to establish the well-posedness and the convergence analysis of the flow given by (6.2.1). Assumption A1 requires convexity of the loss function $L$, which is important to derive global convergence results. Assumptions A1 and A2 are standard in the classical theory of local existence and uniqueness of solutions of ordinary differential equations [11, Subsection 1.2]. Similar assumptions have been considered in a related but different setting in [3, Section 3.2]. As we will see Assumptions A3 and A4 ensure a suitable initialization for the flow, which avoids convergence of the solution towards undesired stationary points of (6.2.1) which are not minimizers. Condition A3 is the generalizetion of to the one presented in [139, Lemma 2.3] for least squares problems. Assumption A5 is crucial to establish boundedness of the trajectory, which is an intermediate step for proving convergence. Indeed, Theorem 6.2.4 and Theorem 6.2.8 would still hold replacing Assumption A5 by boundedness of the trajectory. Observe that the coercivity hypothesis could be replaced for instance by the boundedness of the level set of the loss function.

The following theorem establishes the existence and uniqueness of the global solution of the dynamical system (6.2.1).

**Theorem 6.2.4.** Under assumptions A1-A5 the dynamical system (6.2.1) has a unique solution defined on $[0; +1)$. Moreover, for every $z \in S$, both the Bregman divergence $D_F(z; (t))$ and $L( (t))$ are non-increasing functions of $t \quad 0$.

Proof. Denote by $T_M$ the maximal time for the existence of a solution of (6.2.1); namely,

$$T_M := \sup \{ T > 0 \mid \exists \text{ a } C^1 \text{ function } : [0; T) \to U \text{ solution of (6.2.1)} \}.$$

First we show that $T_M > 0$. Since $r^2 F$ is invertible on $U$, the dynamical system in (6.2.1) is equivalent to the following:

$$\begin{cases} \dot{}(t) = G( (t)) \quad r^2 F( (t)) \quad ^1 r L( (t)); & 0 < t < T_M \\ (0) = \quad _0 \in U. \end{cases} \tag{6.2.2}$$

Since $G( )$, $r^2 F( ) \quad ^1$, and $r L( )$ are locally Lipschitz on $U$, they are continuous and bounded on bounded subsets and their product is also locally Lipschitz. Since $_0 \in U$, it follows from [11, Theorem 1.18] that there exists a unique solution of the initial value problem (6.2.2), and so to (6.2.1), in some time-interval $[0; T_1)$ with $T_1 > 0$. Then, $T_M > 0$.
Next, we prove that $T_M = +1$; namely, that the solution of the dynamical system (6.2.1) is globally defined. Now we proceed to prove the claim that $T_M = +1$ separately for the two cases $U := \mathbb{R}^p \setminus \{0\}$ and $F$ Legendre on $U$.

In order to prove Theorem 6.2.4, we establish an auxiliary result. It states that both the Bregman distance and the loss function, along the trajectory of the flow in $[0; T_M)$, do not increase.

**Lemma 6.2.5.** Let $z \in S$ and suppose that assumptions A1 and A2 hold. Then, both the Bregman divergence $D_F(z, \theta(t))$ and the functional value $L(\theta(t))$ are non-increasing functions for $t \in [0; T_M)$.

Proof. To prove that $t \mapsto D_F(z, \theta(t))$ is decreasing, it is sufficient to note that, for every $t \in [0; T_M)$,

$$\begin{aligned}\frac{d}{dt}D_F(z, \theta(t)) &= \left\langle \nabla F(\theta(t)), \dot{\theta}(t) \right\rangle - \left\langle \nabla^2 F(\theta(t))\dot{\theta}(t), z - \theta(t) \right\rangle \\ &\quad + \left\langle \nabla F(\theta(t)), \dot{\theta}(t) \right\rangle \\ &= \left\langle \nabla^2 F(\theta(t))\dot{\theta}(t), \theta(t) - z \right\rangle \\ &= G(\theta(t))\langle \nabla L(\theta(t)), \theta(t) - z \rangle \\ &\leq G(\theta(t))(L(z) - L(\theta(t))) \\ &\leq 0, \end{aligned} \tag{6.2.3}$$

where we used that $\theta$ is a solution of (6.2.1), the gradient inequality for the convex function $L$ and the fact that, for every $\theta \in U$, $L(\theta) \geq L(z)$ and $G(\theta) \geq 0$. On the other hand, for $t \in [0; T_M)$, $\nabla^2 F(\theta(t))$ is invertible (and the inverse is positive definite). Then,

$$\begin{aligned}\frac{d}{dt}L(\theta(t)) &= \left\langle \nabla L(\theta(t)), \dot{\theta}(t) \right\rangle \\ &= -G(\theta(t)) \left\langle \nabla L(\theta(t)), \nabla^2 F(\theta(t))^{-1} \nabla L(\theta(t)) \right\rangle \\ &\leq 0, \end{aligned}$$

which completes the proof. □

**Case $U = \mathbb{R}^p \setminus \{0\}$**: In this case, we start by providing a strictly-positive lower bound for the norm of $\theta$ along the trajectory, which implies that $\theta([0; T_M)) \subset U$, then, based on this, we prove that $T_M = +\infty$.

To finish this case, we need an intermediate lemma.

**Lemma 6.2.6.** Suppose that assumptions A1, A2, A3 and A4 hold with $U := \mathbb{R}^p \setminus \{0\}$. Then, for every $t \in [0; T_M)$,

$$\|\theta(t)\| > r \quad \text{and} \quad G(\theta(t)) \geq C. \tag{6.2.4}$$

Proof. Let $\theta_0 \in U$ be such that $\|\theta_0\| \geq r$. Then, the gradient inequality for the convex function $L$ at the point 0, the Cauchy-Schwartz inequality, and Assumptions A3 and A4 imply

$$L(\theta_0) \geq L(0) + \langle \nabla L(0), \theta_0 - 0 \rangle \geq L(0) - r\|\nabla L(0)\| = \frac{L(0) + L(\theta_0)}{2} > L(\theta_0). \tag{6.2.5}$$

Next, by contradiction, suppose that there exists a $\hat{t} \in [0; T_M)$ such that $\|\theta(\hat{t})\| \leq r$. Then, by (6.2.5), we have that $L(\theta_0) < L(\theta(\hat{t}))$. This is a contradiction since Lemma 6.2.5 $t \mapsto L(\theta(t))$ is non-increasing on $[0; T_M)$. Then $\|\theta(t)\| > r$ for every $t \in [0; T_M)$. The second part of the statement in (6.2.4) is a consequence of the condition on $G$ assumed in A4. □

Lemma 6.2.6 yields $k\ (t)k > r$ for every $t\ 2\ [0; T_M)$. Lemma 6.2.5 implies that $D_F(z;\ (t))$ is non-increasing for every $z\ 2\ S$ and by Assumption A5, $D_F(z;\ )$ is coercive for some $z\ 2\ S$. Then, $([0; T_M))$ is bounded. Therefore $([0; T_M))$ is contained in a compact subset of $U$. We then derive from [11, Proposition 1.57] that $T_M\ =\ +1$.

**Case $F$ is Legendre on $U$**: We show that $T_M\ =\ +1$.
Let $K$ be the closure of $([0; T_M))$. Assumption A5 implies that $([0; T_M))$ is bounded; therefore, $K$ is compact. Recall that the trajectory is continuous and that $([0; T_M))$ is a subset of the open set $U$. If $K\ \ U$, by applying [11, Proposition 1.57], we conclude that $T_M\ =\ +1$. If $K$ is not contained in $U$, then there exists a point $\ 2\ \overline{U}\ n\ U$ for which there exists a sequence $(t_j)_{j=1}^{+\ 1}\ \ [0; T_M)$ with $t_j\ !\ T_M$ and $(t_j)\ !\ \ $ for $j\ !\ +1$. Integrating in time equation (6.2.1), we get that, for every $j\ 2$ N,

$$\begin{aligned} k\,r\,F(\ (t_j))\ \ \ r\,F(\ _0)k = & \int_0^{t_j} G(\ (t))\,r\,L(\ (t))dt \\ & \int_0^{t_j} G(\ (t))\,k\,r\,L(\ (t))k\,dt \\ & T_M \max_{2K} G(\ ) \max_{2K} k\,r\,L(\ )k: \end{aligned} \tag{6.2.6}$$

The maxima are well-defined due to the fact that the functions $G$ and $k\,r\,Lk$ are locally Lipschitz and so continuous, implying that they are bounded on the compact set $K$ (see, for instance, [91, Theorem 27.4]). For $j\ !\ +1$, we have that $(t_j)\ !\ \ 2\ \overline{U}\ n\ U$ and therefore, since $F$ is a Legendre function, $r\,F(\ (t_j))\ !\ +1$. Thus the left-hand side of (6.2.6) tends to infinity as well, and this implies that $T_M\ =\ +1$.

**Uniqueness**: Since the right hand side of (6.2.2) is locally Lipschitz at every point, local unqueness holds for every initial state, and therefore globally. □

**Remark 6.2.7.** To the best of our knowledge, in previous works, e.g. in [7], the existence and uniqueness of the solution of (6.2.1) in not proved. In contrast, we provide some sufficient conditions to ensure the well-posedness of the system. This is fundamental to guarantee that the implicit bias characterization is meaningful. Indeed, it is well known that the trajectory of a differential equation exists globally if and only if it is locally bounded. So, if the solution is defined on a bounded interval and not on $[0; +1)$, there is a finite time for which the norm of the trajectory diverges. In this case, the limit of the trajectory for that finite time cannot have any interesting properties, since clearly cannot converge to some specific minimizer of $L$.

## 6.2.2 Minimization properties and implicit bias

The next theorem illustrates the minimization properties of the solution of the dynamical system (6.2.1) for $t\ !\ +1$. It states the convergence rates for the loss function and the convergence of the trajectory towards a specific minimizer of the loss, characterized by an implicit bias.

**Theorem 6.2.8.** Assume that A1-A5 hold, then $L(\ (t))\ !\ L\ := \min L$ for $t\ !\ +1$. In addition, $[L(\ (t))\ \ L\ ]\ 2\ L^1([0; +1))$, and, for every $z\ 2\ S \setminus U$ and for every $t > 0$,

$$L(\ (t))\ \ L\ \ \ \ \frac{D_F(z;\ _0)}{Ct}:$$

Moreover, there exists $\theta_1 \in S \setminus U$ satisfying

$$\theta_1 \in \operatorname*{argmin}_{z \in S} \quad F(z) \quad \langle \nabla F(\theta_0), z \rangle + \int_0^{+1} G(\theta(t)) \langle \nabla L(\theta(t)), z - \theta(t) \rangle dt \quad ; \tag{6.2.7}$$

such that

$$\lim_{t \to +1} \theta(t) = \theta_1 :$$

Proof. Let $z \in S$. By the Fundamental Theorem of Calculus, inequality (6.2.3) and Lemma 6.2.6, we have that, for all $t \geq 0$,

$$\begin{aligned} D_F(z, \theta(t)) - D_F(z, \theta_0) &= \int_0^t \frac{d}{ds} D_F(z, \theta(s))\, ds \\ &\leq \int_0^t G(\theta(s))(L(\theta(s)) - L^*) ds \\ &\leq C \int_0^t (L(\theta(s)) - L^*) ds \leq 0. \end{aligned} \tag{6.2.8}$$

Since $D_F(z, \theta(t))$ is non-negative, from the above inequality we derive that, for every $t \geq 0$,

$$\int_0^t (L(\theta(s)) - L^*) ds \leq \frac{D_F(z, \theta_0)}{C} :$$

In particular, $t \mapsto [L(\theta(t)) - L^*]$ belongs to $L^1([0, +1))$. Moreover, since $t \mapsto L(\theta(t))$ is non-increasing by Lemma 6.2.5, we have that $(L(\theta(t)) - L^*)\, t \leq \int_0^t (L(\theta(s)) - L^*) ds \leq D_F(z, \theta_0)/C$ for every $t \geq 0$. From the latter we obtain the rate stated in the theorem.
Next we prove the convergence of the trajectory $(\theta(t))_{t>0}$. Since $z \in S$ by Assumption A5, then $D_F(z, \theta(t))$ is non-negative and decreasing by Lemma 6.2.5, and therefore there exists $\lim_{t \to +1} D_F(z, \theta(t)) \in \mathbb{R}$. Since $D_F(z, \cdot)$ is coercive by assumption A5, we also have that $\theta(t)$ is bounded. Then there exist a sequence $(t_k)_{k \geq 0}$ and a point $\theta_1$ such that, for $k \to +1$, $t_k \to +1$ and $\theta(t_k) \to \theta_1$. From continuity of $L$ and by $[L(\theta(t)) - L^*] \in L^1([0, +1))$, we deduce that

$$L(\theta_1) = \lim_{k \to +1} L(\theta(t_k)) = \lim_{t \to +1} L(\theta(t)) = L^* ;$$

i.e., that $\theta_1 \in S$. We are going to show that indeed $\lim_{t \to 1} \theta(t) = \theta_1$. Suppose that there exists another sequence $(t_n)_{n=1}^{1}$ such that $t_n \to +1$ and $\lim_{t \to 1} \theta(t_n) = \bar{\theta}$. With the same reasoning as above, we know that $\bar{\theta}$ belongs to $S$. Continuity of $D_F(\theta_1, \cdot)$ and the existence of $\lim_{t \to +1} D_F(\theta_1, \theta(t))$ yields

$$\begin{aligned} D_F(\theta_1, \bar{\theta}) &= \lim_{n \to +1} D_F(\theta_1, \theta(t_n)) = \lim_{t \to +1} D_F(\theta_1, \theta(t)) \\ &= \lim_{k \to +1} D_F(\theta_1, \theta(t_k)) = D_F(\theta_1, \theta_1) = 0. \end{aligned}$$

Since $F$ is strictly convex, we get $\bar{\theta} = \theta_1$. Finally, we get that $\lim_{t \to 1} \theta(t) = \theta_1$. To conclude, note that trivially

$$\theta_1 \in \operatorname*{argmin}_{z \in S} D_F(z, \theta_1);$$

where, by (6.2.8) and (6.2.3),

$$\begin{aligned} D_F(z; \theta_1) &= \lim_{t \to +\infty} D_F(z; \theta(t)) \\ &= \lim_{t \to +\infty} D_F(z; \theta_0) + \int_0^t G(\theta(s)) \langle \nabla L(\theta(s)), \nabla z \rangle \, \theta(s) \, ds \\ &= F(z) - F(\theta_0) - \langle \nabla F(\theta_0), z - \theta_0 \rangle + \int_0^{+\infty} G(\theta(s)) \langle \nabla L(\theta(s)), \nabla z \rangle \, \theta(s) \, ds. \end{aligned}$$

Discarding the terms that are constant with respect to $z$, we obtain the claim. □

The next corollary characterizes the implicit bias for time-warping mirror flow when the loss function is the composition of a convex function with a linear operator. This implicit bias corresponds to the Bregman distance between the set of minimizers of the loss and the initialization, extending the results presented in [7, 139].

**Corollary 6.2.9.** Suppose that assumptions A1-A5 hold. Let $X \in \mathbb{R}^{d \times p}$, let $y \in \mathbb{R}^d$, let $\ell : \mathbb{R}^d \to \mathbb{R}$ be differentiable with a locally Lipschitz gradient. Assume that $L = \ell \circ X$ and $S = \{z \in \mathbb{R}^p \mid Xz = y\}$. Then, there exists a solution $\theta_1 = \lim_{t \to +\infty} \theta(t)$ satisfying

$$\theta_1 \in \operatorname*{argmin}_{z \in S} \{F(z) - \langle \nabla F(\theta_0), z \rangle\} = \operatorname*{argmin}_{z \in S} D_F(z; \theta_0).$$

Proof. It follows from Theorem 6.2.8 that there exists $\theta_1 \in S \setminus U$ such that $\theta_1 = \lim_{t \to +\infty} \theta(t)$ and

$$\theta_1 \in \operatorname*{argmin}_{z \in S} \{F(z) - \langle \nabla F(\theta_0), z \rangle + \int_0^{+\infty} G(\theta(t)) \langle \nabla L(\theta(t)), \nabla z \rangle \, \theta(t) \, dt\}.$$

To conclude the assertion, it is sufficient to prove that the integral is constant with respect to $z$. By the chain rule, for every $z \in S \setminus U$,

$$\begin{aligned} \int_0^{+\infty} G(\theta(t)) \langle \nabla L(\theta(t)), \nabla z \rangle \, \theta(t) \, dt &= \int_0^{+\infty} G(\theta(t)) \left\langle X^\top \nabla \ell(X\theta(t)), \nabla z \right\rangle \theta(t) \, dt \\ &= \int_0^{+\infty} G(\theta(t)) \langle \nabla \ell(X\theta(t)), y - X\theta(t) \rangle \, dt \end{aligned}$$

which is negative, bounded below by $-D_F(\theta_1; \theta_0)$, and it is independent of $z \in S$. Consequently, from (6.2.7), we obtain that the limit of the trajectory satisfies

$$\theta_1 \in \operatorname*{argmin}_{z \in S} \{F(z) - \langle \nabla F(\theta_0), z \rangle\} = \operatorname*{argmin}_{z \in S} D_F(z; \theta_0).$$

□

**Remark 6.2.10.** (i) The point $\theta_1$ is the Bregman projection, with respect to the entropy $F$, of $\theta_0$ on $S$. Since $S$ is a convex set and $F$ is a strictly-convex function, the minimizer is unique. Therefore,

$$\operatorname*{argmin}_{z \in S} D_F(z; \theta_0) = \{\theta_1\}. \tag{6.2.9}$$

(ii) The limit $\theta_1 \in U$, the domain of the entropy $F$, which therefore can also implicitly enforce some constraints. For instance, the parametrization $\theta = \theta$ implicitly enforces the positivity of the solution: in this case the domain of $F$ is the positive orthant, see the case $L = 2$ in equation (A.1.1).

(iii) Even if the term $\int_0^{+1} G(\ (t))\langle r\ L(\ (t)),\ z\ \rangle\ (t)\,idt$ is not constant with respect $z$, it is upper bounded by the quantity $\int_0^{+1} G(\ (t))\,(L\ \ L(\ (t)))\,dt < 0$, that measures the cumulative functional value decrease.

### 6.2.3 Application to fully connected linear networks

We consider the setting of Example 5.3.1, where $\ = (W; w) \in \mathbb{R}^{p\ \ m} \ \mathbb{R}^m$ and $\ = Ww$ for a loss function $L$ satisfying Assumption A1. When $L$ is the square loss, this setting has been considered in [7] and is a specific instance of a more general class of overparametrizations analyzed in [6].

Suppose that the initialization satisfies $W_0^{>} W_0 = w_0 w_0^{>}$: Then, for every $t > 0$, the trajectory $\ (t)$ corresponding to the gradient flow trajectory in the reparameterization $(W(t); w(t))$ considered in (5.3.1) satisfies the differential equation

$$r^2\ \ \frac{2}{3}k\ \ (t)k^{3=2}\ \ \dot{}(t) = \ \ k\ \ (t)k^{1=2}\, r\, L(\ \ (t)); \tag{6.2.10}$$

namely (6.2.1) with $F(\ ) = (2=3)k\ \ k^{3=2}$ and $G(\ ) = k\ \ k^{1=2}$ (for more details, see Section A.1, Example A.1.3 in the appendix). The Hessian and its inverse are given by

$$r^2 F(\ ) = k\ \ k\ ^{1=2}\ \ \mathrm{Id}\ \ \frac{\ ^{>}}{2k\ \ k^2} \qquad \text{and} \qquad r^2 F(\ )\ ^{1} = k\ \ k^{1=2}\ \ \mathrm{Id} + \frac{\ ^{>}}{k\ \ k^2}\ :$$

In $U = \mathbb{R}^p \setminus f0g$, the function $G$ is locally Lipschitz, and $F$ is strictly convex, twice differentiable, and its Hessian is invertible, thus A2 is satisfied. If the initialization is such that $L(\ _0) < L(0)$, then A4 holds with $C = \overline{p\ r}$. Combining this with the fact that $L(\ _0) < L(0)$, as $U = \mathbb{R}^p \setminus f0g$, it follows that $S \setminus U \neq\ ;$. Consequently, A3 is also satisfied. To apply Theorem 6.2.4 and Theorem 6.2.8, we show that assumption A5 holds, namely that there exists $z \in S \setminus U$ such that $D_F(z;\ )$ is coercive. Indeed, the Bregman divergence $D_F(z;\ )$ is coercive for every $z \in \mathbb{R}^p$: from the Cauchy-Schwarz inequality,

$$\begin{aligned} D_F(z;\ \ ) &= \frac{2}{3}kzk^{3=2}\ \ \frac{2}{3}k\ \ k^{3=2}\ \ k\ \ k\ ^{1=2}\langle\ \ j\ z\ \ \ i \\ &= \frac{2}{3}kzk^{3=2} + \frac{1}{3}k\ \ k^{3=2}\ \ k\ \ k\ ^{1=2}\langle\ \ j\ z\, i \\ &\quad \frac{2}{3}kzk^{3=2} + \frac{1}{3}k\ \ k^{3=2}\ \ k\ \ k^{1=2}kzk \\ &= \frac{2}{3}kzk^{3=2} + k\ \ k^{1=3}\ \ \frac{1}{3}k\ \ k\ \ \ kzk\ \ : \end{aligned}$$

Then, when the norm of $\ $ tends to infinity, $D_F(z;\ \ )$ also tends to infinity.
Therefore, Theorem 6.2.4 yields the existence and uniqueness of the solution of the dynamical system (6.2.10). From Theorem 6.2.8 we get that the function $L(\ \ (t))$ is non-increasing in $t \ \ 0$ and that, for every $z \in S$ and for every $t > 0$,

$$L(\ \ (t))\ \ \ L \qquad \frac{(2k\, r\, L(0)k)^{1=2} D_F(z;\ \ _0)}{(L(0)\ \ \ L(\ \ _0))^{1=2} t}:$$

Moreover, $\lim_{t\ !\ +1} \ \ (t) = \ _1$, where

$$_1 \in \operatorname*{argmin}_{z \in S} kzk^{3=2}\ \ \frac{\langle\ _0\ j\ z\, i}{k\ \ _0 k} + \int_0^{+1} \frac{2}{3}k\ \ (t)k^{3=2}\langle r\, L(\ \ (t))\ j\ z\ \ \ (t)\,idt:$$

**Remark 6.2.11.** In the specific case of the quadratic loss function $L(\ ) = kX \quad yk^2$ for some $X \ 2 \ \mathbb{R}^{d \ p}$ and $y \ 2 \ \mathbb{R}^d$, Corollary 6.2.9, implies

$$\lim_{t \ ! \ +1} \ (t) = \ _1 \ 2 \operatorname*{argmin}_{z2S} kzk^{3=2} \quad \frac{\hbar \ _0 \ j \ z \ i}{k \ _0 k}:$$

Therefore we recover the implicit bias presented in [7], and our result generalizes the latter to general losses. Moreover, in [7], the authors assume the convergence of the trajectory to a feasible point while we prove it.

**Remark 6.2.12.** Our results can be contrasted to those in [6]. In the latter paper, given $N$ matrices $W_i \ 2 \ \mathbb{R}^{p_i \quad p_{i+1}}$ for every $i$ with $p_1 = p$, $p_{N+1} = 1$, the authors consider the reparametrization $= W_1 ::: W_N$. For the special case $N = 2$, this setting coincides with ours. In [6, Claim 2], the flow (A.1.9) has been derived, and interpreted in a different way. Our results shed new light on their result, thanks to time warping mirror flow interpretation. Thanks to this, we can transform the impossibility result [6, Theorem 2], stating that the trajectory $(t)$ does not follow any gradient flow, into an existence result if the geometry is modified through a time warped mirror function. Morever, the mirror flow interpretation gives the possibility of naturally deriving an implicit bias.

**Remark 6.2.13.** The paper [7] considered the case when $W^>(0)W(0) \quad w(0)w^>(0) = \mathrm{Id}$ for $\ 6= 0$. Observe that for suitable initializations, the assumptions of Theorem 6.2.8 still remain valid. In this case, we could compute the implicit bias, which is the generalization of the one proposed in [6][7, Theorem 2] for the least square, see [7, Appendix B]. The computations are very technical and are beyond the scope of this paper.

We can further characterize the limit point for fully connected linear networks of depth 2, in the linear setting of Corollary 6.2.9. The proof uses techniques similar to those in [139].

**Corollary 6.2.14.** In the setting of this section, suppose that, as in Corollary 6.2.9, $L(\ ) = `(X \ )$ and $S = f \ 2 \ \mathbb{R}^p \ j \ X \ = yg$, where $X \ 2 \ \mathbb{R}^{d \ p}$ and $y \ 2 \ \mathbb{R}^d$. Additionally, suppose that assumption A1 is satisfied, that the initialization satisfies $W^>(0)W(0) = w(0)w^>(0)$ and that $L(\ _0) < L(0)$. Let be the (unique) solution of

$$\min_{z2S} kzk^2: \tag{6.2.11}$$

Let $P : \mathbb{R}^p \ ! \ \mathbb{R}^p$ be the projection operator onto the kernel of $X$. Then the trajectory $(t) = W(t)w(t)$ satisfies that $\lim_{t \ ! \ +1} \ (t) = \ _1$, where

$$_1 = \quad + P(\ _0) \quad \frac{k \ _1 k}{k \ _0 k}; \quad \text{and}$$

$$k \ _1 k = \frac{kP(\ _0)k^2 + \sqrt{kP(\ _0)k^4 + 4k \ _0 k^2 k \quad k^2}}{2k \ _0 k}:$$

Proof. As shown in Section 6.2.3 all the assumptions of Theorem 6.2.4 and Theorem 6.2.8 are satisfied. Recall that $\ker(X) \quad \mathbb{R}^p$ is a closed subspace and that $\ker(X)^? = \overline{\mathrm{ran}(X^>)} = \mathrm{ran}(X^>)$ (see for instance [15, Fact 2.25 (iv)]). Let us define $P_? = \mathrm{Id} \quad P$, the projection onto $\ker(X)^?$. Then we can write $_1$ as

$$_1 = P(\ _1) + P_?(\ _1):$$

First, we compute $P(\ _1)$. Theorem 6.2.4 implies that equation (A.1.9),

$$\dot{}(t) = \quad k \ (t)k \quad \mathrm{Id} + \frac{(t) \ ^>(t)}{k \ (t)k^2} \quad r \, L(\ (t)); \tag{6.2.12}$$

has a unique solution defined on $[0; +1)$ and $k\ (t)k > 0$ for every $t \quad 0$. Since $P\ (r\,L(\ )) = P\ \ X^{>} r\,`(X\ \ ) \ = 0$, if we apply the linear operator $P$ to (6.2.12), we obtain that

$$\frac{d}{dt}P(\ (t)) = P(\_(t)) = \quad k\ (t)kP\ (r\,L(\ (t))) \quad \frac{P\ (\ (t))}{k\ (t)k}h\ (t)\ j\ r\,L(\ (t))i$$
$$= \quad \frac{P\ (\ (t))}{k\ (t)k}h\ (t)\ j\ r\,L(\ (t))i: \tag{6.2.13}$$

On the other hand, (6.2.12) yields

$$\frac{d}{dt}k\ (t)k = \quad \frac{(t)}{k\ (t)k}\ j\ \_(t) \quad = \quad 2h\ (t)\ j\ r\,L(\ (t))i: \tag{6.2.14}$$

Combining equations (6.2.13) and (6.2.14), we obtain the following differential equation:

$$\frac{d}{dt}(P\ (\ (t))) = \frac{P(\ (t))}{2k\ (t)k} \quad \frac{d}{dt}k\ (t)k \quad : \tag{6.2.15}$$

Equation (6.2.15) implies

$$\frac{d}{dt} \quad \frac{P(\ (t))}{k\ (t)k^{1=2}} \quad = \frac{1}{k\ (t)k^{1=2}}\frac{d}{dt}(P\ (\ (t))) \quad \frac{P(\ (t))}{2k\ (t)k^{3=2}} \quad \frac{d}{dt}k\ (t)k$$
$$= \frac{1}{k\ (t)k^{1=2}} \quad \frac{d}{dt}(P\ (\ (t))) \quad \frac{P(\ (t))}{2k\ (t)k} \quad \frac{d}{dt}k\ (t)k \quad = 0:$$

The previous result implies that the term $\frac{P(\ (t))}{k\ (t)k^{1=2}}$ remains constant for every $t > 0$ and so it is equal to $\frac{P(\ _0)}{k\ _0k^{1=2}}$. Consequently, we obtain the following expression:

$$P\ (\ (t)) = P(\ _0) \quad \frac{k\ (t)k}{k\ _0k}^{1=2}:$$

Recall that, from Theorem 6.2.8, $\lim_{t\,!\,1}\ (t) = \ _1$. From the continuity of the projection and of the norm, taking the limit for $t\ !\ +1$ in the previous equation, we get

$$P(\ _1) = P(\ _0) \quad \frac{k\ _1k}{k\ _0k}^{1=2}:$$

Next we compute $P_?(\ _1)$. Recall that $\ _1\ 2\ S$, where $S = f\ \ 2\ \mathbb{R}^p\ j\ X\ \ = yg$. We have that

$$y = X\ \ _1 = X(P(\ _1) + P_?(\ _1)) = XP_?(\ _1):$$

Thus, $P_?(\ _1)\ 2\ S$. Similarly, since $\ 2\ S$, we have that

$$y = X\ \ = X\ (P_?(\ \ ) + P(\ \ )) = XP_?(\ \ ):$$

Then, $P_?(\ \ )\ 2\ S$, meaning that it is a feasible point for the minimization problem (6.2.11). Moreover, since is a solution of problem (6.2.11), we have that

$$k\ \ k^2 \quad kP_?(\ \ )k^2 \quad kP_?(\ \ )k^2 + kP(\ \ )k^2 = k\ \ k^2:$$

Therefore, we conclude that $P(\ \ ) = 0$ and so that $\ 2\ \ker(X)^?$. Moreover, since $X\ \ =$ $y = X\ \ _1$, we obtain that $\ _1\ 2\ \ker X$. Then, since $\ 2\ \ker(X)^?$, we derive from

$$8p\ 2\ \ker(X)^? \quad hp \quad \ \ j\ \ _1 \quad \ \ i = 0;$$

that $P_?(\ _1) = \ $. Then,

$$_1 = P(\ _1) + P_?(\ _1) = P(\ _0) \ \sqrt{\frac{k\ _1k}{k\ _0k}} + \ :$$

Finally, we obtain the value of $k\ _1k$ by solving the following second order equation:

$$k\ _1k^2 = kP_?(\ _1)k^2 + kP(\ _1)k^2 = k\ \ k^2 + kP(\ _0)k^2\frac{k\ _1k}{k\ _0k}:$$

□

The previous result establishes that when we run gradient flow on $\ = (W; w)$, the corresponding trajectory is biased towards a specific element of $S$, that is $\ _1$ given by the sum of two terms: $\ $, the minimal-norm vector in $S$; and a rescaling of $P(\ _0)$, the projection of the inizialization onto $\ker(X)$.

**Remark 6.2.15.** We compare the previous result with the well-known one for vanilla gradient flow, namely

$$\begin{cases} \tilde{\ } = \ \ r\,L(\tilde{\ }(t)) \\ \tilde{\ }(0) = \tilde{\ }_0: \end{cases} \tag{6.2.16}$$

On the one hand, the system (6.2.16) is an instance of (6.2.1) with $F(\ ) = (1=2)k\ \ k^2$ and $G(\ ) = 1$. Therefore, $D_F(z;\ _0) = (1=2)kz\ \ \ _0k^2$, and we derive from (6.2.9) that $\tilde{\ }(t)\ !\ \tilde{\ }_1$, where $\tilde{\ }_1$ is the projection onto $S$ of $\tilde{\ }_0$. Following a similar reasoning to the one in the proof of Corollary 6.2.14, it is possible to show that

$$\tilde{\ }_1 = \ \ + P(\tilde{\ }_0):$$

We derive that the main difference in the implicit bias between running vanilla gradient flow on $\ $ and gradient flow on the reparametrization $\ = Ww$ is given by the scaling factor on $P(\ _0)$, namely $\sqrt{\frac{k\ _1k}{k\ _0k}}$. In particular $\ _1 = \tilde{\ }_1$ if $\ _0 \ 2\ \ker(X)^?$.

## 6.3 Reparameterizing Mirror Descent for radial functions as Projected Gradient Descent

In this section we consider reparametrizations in terms of polar coordinates, related to weight normalization [89, 118, 139], namely of the form

$$= h(\ )\frac{w}{kwk};$$

where $h\colon \mathbb{R}\ !\ \mathbb{R}$ is a given function and $\ \ 2\ \mathbb{R}$ and $w\ 2\ \mathbb{R}^p$ are the new variables. The associated gradient flow on the reparameterization is given by:

$$\dot{w}(t) = \ \ r_w L(\ (t)) = \frac{h(\ (t))}{kw(t)k}\ \ \mathrm{Id}\ \ \frac{w(t)w^>(t)}{kw(t)k^2}\ \ r\,L(\ (t)) \tag{6.3.1}$$

$$\dot{\ }(t) = \ \ r\ \ L(\ (t)) = h'(\ (t))\ \ \frac{w(t)}{kw(t)kj}\ j\ r\,L(\ (t))\ : \tag{6.3.2}$$

In the following we will show that this flow is equivalent to a generalized mirror one, for a mirror function of the form $F(\ ) = f(k\ \ k)$ and a time warping of the form $g(k\ \ k)$. To derive the main result of this section, we need an auxiliary Lemma, providing a reformulation of (6.2.1) for radial functions. From now on, unless explicitly stated otherwise, when we refer to assumption A2, we mean it with $U = \mathbb{R}^p n\ f0g$.

**Lemma 6.3.1.** Let assumptions A1 and A3 hold. Let $f : [0; +1) \to [0; +1)$ be a twice differentiable function on $(0; +1)$ such that $f'(s) > 0$, $f''(s) > 0$ for every $s > 0$ and $1=f''$ is locally Lipschitz on $(0; +1)$. Let $g : [0; +1) \to [0; +1)$ be locally Lipschitz on $(0; +1)$ and such that, defining $r$ as in Assumption A4, there exists $C$ such that $g(s) > C$ for every $s > r$. For every $\in \mathbb{R}^p$, define $F( ) = f(k k)$ and $G( ) = g(k k)$. Then assumptions A2 and A4 hold, and the dynamical system (6.2.1) corresponding to $F$ and $G$ can be written equivalently as

$$\begin{cases} \dot{-}(t) = \dfrac{g(k (t)k)k (t)k}{f'(k (t)k)} \mathrm{Id} - \dfrac{(t) \otimes (t)}{k (t)k^2} r L( (t)) \\ \qquad\quad \dfrac{g(k (t)k)}{f''(k (t)k)} \dfrac{(t) \otimes (t)}{k (t)k^2} r L( (t)); \qquad t > 0 \\ (0) = {}_0 \in \mathbb{R}^p \setminus \{0\}. \end{cases} \tag{6.3.3}$$

Proof. The function $F$ is strictly convex since it is the composition of the increasing strictly convex function $f$ with the convex function $\mapsto k k$. The gradient and the Hessian of $F( ) = f(k k)$, for $\neq 0$ are

$$\begin{aligned} r F( ) &= f'(k k) \frac{}{k k} \quad \text{and} \\ r^2 F( ) &= f''(k k) \frac{\otimes}{k k^2} + \frac{f'(k k)}{k k} \left( \mathrm{Id} - \frac{\otimes}{k k^2} \right); \end{aligned} \tag{6.3.4}$$

and clearly $F$ is twice differentiable on $\mathbb{R}^p \setminus \{0\}$. Using the Sherman-Morrison formula (2.2.1) with $a = f''(k k) \frac{f'(k k)}{k k} \frac{}{k k^2}$, $b = $, and $M = \frac{f'(k k)}{k k}\mathrm{Id}$, since $b^\top M^{-1} a + 1 = \frac{k k f''(k k)}{f'(k k)} \neq 0$ we get that

$$r^2 F( )^{-1} = \frac{1}{f''(k k)} \frac{\otimes}{k k^2} + \frac{k k}{f'(k k)} \left( \mathrm{Id} - \frac{\otimes}{k k^2} \right); \tag{6.3.5}$$

which is locally Lipschitz in $U \setminus \{0\}$ since $1=f''$ and $f'$ are locally lipschitz in $(0; +1)$. In addition, $S \setminus U \neq ;$ since 0 is not a minimizer of $L$ by Assumption A3. Local lipschitzianity of $g$ implies that $G$ is locally lipschitz. Therefore Assumption A2 is satisfied. Assumption A4 is also trivially satisfied, therefore Lemma 6.2.6 implies that the solution of the dynamical system is bounded away from zero; namely, that $k (t)k \geq r > 0$ for every $t \geq 0$. This implies that all the quantities in (6.3.4) and in (6.3.5) are well-defined along the trajectory. The statement then follows by plugging the explicit form of $r^2 F( )^{-1}$ in equation (6.2.2), which is equivalent to (6.2.1). □

**Remark 6.3.2.** By (6.3.3), we get that

$$\begin{aligned} \frac{1}{2}\frac{d}{dt} k (t)k^2 &= \left\langle (t), \dot{-}(t) \right\rangle = \frac{g(k (t)k)}{f''(k (t)k)} \left\langle (t), r L( (t)) \right\rangle \\ \frac{d}{dt} \frac{(t)}{k (t)k} &= \frac{1}{k (t)k} \left( \mathrm{Id} - \frac{(t) \otimes (t)}{k (t)k^2} \right) \dot{-}(t) = \frac{g(k (t)k)}{f'(k (t)k)} \left( \mathrm{Id} - \frac{(t) \otimes (t)}{k (t)k^2} \right) r L( (t)). \end{aligned}$$

This computation shows that if the second term on the right-hand side of (6.3.3) is zero, then the norm is constant.

In the next theorem, we focus on reparametrizations of the form $= h( )\frac{w}{kwk}$. This theorem establishes a sufficient condition that allows us to derive from the gradient flow with respect to and $w$ a time-warping mirror flow with respect to , as in (6.2.1).

**Theorem 6.3.3.** Let assumption A1 hold. Consider the weight normalization parameterization $= h(\ )\frac{w}{kwk}$ and the dynamics on $(\ ;w)$ defined in (6.3.1)-(6.3.2) with initialization $(\ _0;w_0)$ such that $h(\ _0) > 0$ and $kw_0k = 1$ and assume that assumption A3 holds for $\ _0 = h(\ _0)w_0=kw_0k$. Assume that there exists a function $f : [0;+1) \ ! \ [0;+1)$ twice differentiable on $(0;+1)$ such that $f^{00}(s) > 0$ for every $s > 0$, $1=f^{00}$ is locally Lipschitz on $(0;+1)$ and satisfying the following equality

$$(h^0(\ ))^2 = \frac{f^0(h(\ ))h(\ )}{f^{00}(h(\ ))}: \tag{6.3.6}$$

Suppose in addition that there exists $c > 0$ such that $f^0(s) > c=s$ for every $s > r$ ($r$ is defined in Assumption A3) and define $g\colon \ R_+ \ ! \ R$ as $g(s) = sf^0(s)$. Then the assumptions A2 and A4 are satisfied, and for every $t > 0$, $kw(t)k = 1$, $h(\ (t)) > 0$, and $\ (t)$ solves the following dynamical system, for $t \ \ 0$

$$\begin{cases} \dot{\ }(t) = k\ (t)k^2 \ \ \mathrm{Id} \ \ \frac{(t)\ ^>(t)}{k\ (t)k^2} \ \ r\,L(\ (t)) \\ \qquad\quad \frac{f^0(k\ (t)k)}{f^{00}(k\ (t)k)}\frac{(t)\ ^>(t)}{k\ (t)k}\, r\,L(\ (t)); \ t > 0 \\ (0) = \ _0 \ 2 \ \mathbb{R}^p \ n \ f0g: \end{cases} \tag{6.3.7}$$

i.e., the dynamic of $\ (t)$ satisfies (6.2.1), with $F(\ ) = f(k\ \ k)$ and $G(\ ) = g(k\ \ k)$.

Proof. We first show that $t \ 2 \ (0;+1) \ 7! \ kw(t)k$ is constant. Since $w$ is a solution of the dynamical system in (6.3.1), we obtain

$$\frac{d}{dt}kw(t)k^2 = 2hw(t) \ j \ \underline{w}(t)i = 0;$$

therefore the norm of $w(t)$ is constant. Moreover, since $kw(0)k = 1$ by assumption, for every $t > 0$, $kw(t)k = 1$. We next derive an equation for $\ (t)$ using (6.3.1)-(6.3.2). The derivative of the product yields

$$\begin{aligned} \dot{\ }(t) &= h(\ (t))\underline{w}(t) + h^0(\ (t))\ \dot{\ }(t)w(t) \\ &= \frac{(h(\ (t)))^2}{kw(t)k}\ \ \mathrm{Id} \ \ \frac{w(t)w^>(t)}{kw(t)k^2} \ \ r\,L(\ (t)) \\ &\quad\ h^0(\ (t))\ ^2 w(t)w^>(t)\, r\,L(\ (t)): \end{aligned} \tag{6.3.8}$$

We now prove that, for every $t > 0$, $h(\ (t)) > 0$ and consequently $k\ (t)k = h(\ (t))$. It follows from (6.3.8) that

$$\begin{aligned} \frac{d}{dt}L(\ (t)) &= \ \left\langle r\,L(\ (t)) \ j \ \dot{\ }(t) \right\rangle \\ &= \frac{(h(\ (t)))^2}{kw(t)k}k\, r\,L(\ (t))k^2 + \frac{(h(\ (t)))^2}{kw(t)k} \ \ r\,L(\ (t)) \ j \ \frac{w(t)}{kw(t)k}\ ^2 \\ &\quad\ h^0(\ (t))\ ^2 h\, r\,L(\ (t)) \ j \ w(t)i^2 \\ &\quad\ h^0(\ (t))\ ^2 h\, r\,L(\ (t)) \ j \ w(t)i^2 \\ &\quad\ 0: \end{aligned}$$

Therefore $t \ 2 \ (0;+1) \ 7! \ L(\ (t))$ is decreasing. Next, we provide a strictly-positive lower bound for the norm of $\ $ along the trajectory. Let $\ \ 2 \ \mathbb{R}^p \ n \ f0g$ be such that $k\ \ k \ \ \ r =$

$\frac{L(0) \quad L(\ _0)}{2k\, r\, L(0)k}$. Then, convexity of $L$ and the Cauchy-Schwarz inequality yield

$$\begin{aligned} L(\ ) \quad & L(0) + h\, r\, L(0)\, j \qquad 0i \\ & L(0) \quad rk\, r\, L(0)k \\ = \ & \frac{L(0) + L(\ _0)}{2} \\ > \ & L(\ _0): \end{aligned} \tag{6.3.9}$$

This implies on the one hand that $kh(\ _0)w_0k = h(\ _0) > r$ otherwise we would derive a contradiction from (6.3.9) choosing $\quad = \ _0$. On the other hand, from (6.3.9) we derive that $k\ \ (t)k > r$ for every $t > 0$ since $L(\ \ (t))$ is decreasing. Then,for every $t \quad 0$, $kh(\ \ (t))\, k > r$, which implies that $h(\ \ (t)) = k\ \ (t)k$, since the sign of $h(\ \ (t))$ remains positive. Replacing $w(t) = \frac{(t)}{k\ \ (t)k}$, $kw(t)k = 1$, $k\ \ (t)k = h(\ \ (t))$, and

$$(h^0(\ \ ))^2 = \frac{f^0(h(\ \ ))h(\ \ )}{f^{00}(h(\ \ ))};$$

in (6.3.8), we deduce that

$$\begin{aligned} {}_-(t) = \quad & k\ \ (t)k^2 \quad \mathrm{Id} \quad \frac{(t) \quad {}^>(t)}{k\ \ (t)k^2} \quad r\, L(\ \ (t)) \\ & \frac{f^0(k\ \ (t)k)k\ \ (t)k}{f^{00}(k\ \ (t)k)} \frac{(t) \quad {}^>(t)}{k\ \ (t)k^2}\, r\, L(\ \ (t)) \end{aligned}$$

Finally, Lemma 6.3.1 with $F(\ \ ) = f(k\ \ k)$ and $G(\ \ ) = g(k\ \ k) = k\ \ kf^0(k\ \ k)$ implies (6.3.7) and proves the statement. □

**Remark 6.3.4.**

1. If $h(\ \ ) > 0$, from the definition of weight normalization reparameterization $\quad = h(\ \ )\frac{w}{kwk}$, we have that $k\ \ k = h(\ \ )$. The previous theorem guarantees also that, if the initialization satisfies $h(\ _0) > 0$, then $h(\ \ (t)) > 0$ for every $t > 0$ and, consequently, $k\ \ (t)k = h(\ \ (t))$ for every $t > 0$.

2. The same dynamic for $\quad$ could be derived from other reparametrizations (for instance, see Example 5.3.1 and Remark 6.3.6(ii)). However, the proposed reparameterization only requires $p + 1$ parameters, which, in practice, makes this approach more efficient than other overparameterization schemes. Furthermore, this technique involves only one additional variable compared to vanilla gradient descent, and requires only matrix-vector multiplications and gradient calculations.

3. The previous theorem gives a sufficient condition on the functions $h$, $f$, and $g$ to cast the gradient flow over $\quad$ and $w$ into a time-warping mirror flow on $\quad$, as in Theorems 6.2.4 and 6.2.8. This result allows us to establish the existence and uniqueness of the trajectory $\quad$ and convergence to an implicit bias.

4. If $(h^0(\ \ ))^2$ can be expressed in terms of $h(\ \ )$ as $(h^0(\ \ ))^2 = \ '(h(\ \ ))$ for some function $\ ' : \mathrm{R}_+ \ ! \ \mathrm{R}_+$, we can determine $f^0$ in (6.3.6) by solving the following differential equation:
$$'(s)f^{00}(s) \quad f^0(s)s = 0; \qquad s > 0:$$

In the context of Corollary 6.2.9, where the loss is a composition of a convex function with a linear operator and the solution set is a linear subspace, the next corollary allows us to further characterize the limit point of the flow deriving from weight normalization. Similarly to Corollary 6.2.14, the limit point can be expressed as the sum of the minimal norm solution and a term in the kernel of the linear operator. We generalize the techniques used in [139, Theorem 2.6] to a broader class of weight-normalization reparameterizations.

**Corollary 6.3.5.** Under the same assumptions of Theorem 6.3.3, suppose that, $L(\ ) = \ell(X\ )$ and $S = \{\ \in \mathbb{R}^p \mid X\ = y\}$, where $X\ \in \mathbb{R}^{d\ p}$ and $y \in \mathbb{R}^d$. Additionally, suppose that assumption A5 is satisfied. Let be the (unique) solution of

$$\min_{z \in S} \|z\|^2.$$

Moreover, let $P : \mathbb{R}^p \to \mathbb{R}^p$ be the projection operator onto the kernel of $X$. Then, for $t \to +1$, $(t) \to {}_1 = \operatorname{argmin}_{z \in S} f(\|z\|) \quad \frac{f'(\|\ _0\|)}{\|\ _0\|} \langle\ _0, z\rangle$, and

$${}_1 = \quad + P(\ _0)\frac{f'(\|\ _0\|)\|\ _1\|}{\|\ _0\| f'(\|\ _1\|)}.$$

Proof. Let $P$ be the projection operator onto $\ker(X)$ and let $P_? := \mathrm{Id} \quad P$, the projection onto $\ker(X)^?$. Then

$${}_1 = P(\ _1) + P_?(\ _1).$$

First, we compute $P(\ _1)$. It follows from Theorem 6.3.3 that the differential equation

$$\begin{aligned} \dot{}(t) = \quad & \|\ (t)\|^2 \quad \mathrm{Id} \quad \frac{(t)\ ^{\top}(t)}{\|\ (t)\|^2} \quad r L(\ (t)) \\ & \frac{f'(\|\ (t)\|)\|\ (t)\|}{f''(\|\ (t)\|)} \frac{(t)\ ^{\top}(t)}{\|\ (t)\|^2} r L(\ (t)). \end{aligned} \tag{6.3.10}$$

has a unique solution such that $(0) = {}_0$ defined on $[0; +1)$. Since $P(r L(\ )) = P\ X^{\top} r\ell(X\ ) = 0$, if we apply the linear operator $P$ to both sides of the previous equation, we obtain that the differential equation

$$\begin{aligned} \frac{d}{dt} P(\ (t)) = P(\dot{}(t)) = \quad & \|\ (t)\|^2 P\,(r L(\ (t))) \\ & + P(\ (t))\langle\ (t)\ r L(\ (t))\rangle \quad 1 \quad \frac{f'(\|\ (t)\|)}{f''(\|\ (t)\|)\|\ (t)\|} \\ = \quad & P(\ (t))\langle\ (t)\ r L(\ (t))\rangle \quad \frac{f'(\|\ (t)\|)}{f''(\|\ (t)\|)\|\ (t)\|} \quad 1 \ : \end{aligned} \tag{6.3.11}$$

Equation(6.3.10) implies

$$\frac{d}{dt}\|\ (t)\| = \quad \frac{(t)}{\|\ (t)\|}\ \dot{}(t) \quad = \quad \frac{f'(\|\ (t)\|)}{f''(\|\ (t)\|)}\langle\ (t)\ r L(\ (t))\rangle. \tag{6.3.12}$$

Combining equations (6.3.11) and (6.3.12), we obtain the following differential equation:

$$\frac{d}{dt} P(\ (t)) = P(\ (t)) \quad \frac{1}{\|\ (t)\|} \quad \frac{f''(\|\ (t)\|)}{f'(\|\ (t)\|)} \quad \frac{d}{dt}\|\ (t)\|. \tag{6.3.13}$$

By multiplying the equation (6.3.13) by the integrating factor $\frac{f'(\|\ (t)\|)}{\|\ (t)\|}$ (which is strictly positive as proven in Lemma 6.2.6) and then rearranging the terms, we obtain the following result:

$$\begin{aligned} \frac{d}{dt} \quad \frac{P(\ (t)) f'(\|\ (t)\|)}{\|\ (t)\|} \quad = & \frac{f'(\|\ (t)\|)}{\|\ (t)\|}\frac{d}{dt}(P\ (\ (t))) \\ & + P(\ (t)) \quad \frac{f''(\|\ (t)\|)}{\|\ (t)\|} \quad \frac{f'(\|\ (t)\|)}{\|\ (t)\|^2} \quad \frac{d}{dt}\|\ (t)\| \\ = & \frac{f'(\|\ (t)\|)}{\|\ (t)\|}\frac{d}{dt}(P\ (\ (t))) \\ & \frac{f'(\|\ (t)\|)}{\|\ (t)\|} P(\ (t)) \quad \frac{1}{\|\ (t)\|} \quad \frac{f''(\|\ (t)\|)}{f'(\|\ (t)\|)} \quad \frac{d}{dt}\|\ (t)\| = 0. \end{aligned}$$

Thus the term $\frac{P(\ (t))f^0(k\ (t)k)}{k\ (t)k}$ remains constant for every $t > 0$ and it is equal to $\frac{P(\ _0)f^0(k\ _0k)}{k\ _0k}$. Consequently, we obtain the following expression:

$$P(\ (t)) = P(\ _0)\frac{f^0(k\ _0k)k\ (t)k}{k\ _0kf^0(k\ (t)k)}:$$

Recall that, from Theorem 6.2.8, $\lim_{t\ !\ 1}\ (t) =\ _1$. From the continuity of the projection and the continuity of the norm, taking the limit for $t\ !\ +1$ in the previous equation, we get

$$P(\ _1) = P(\ _0)\frac{f^0(k\ _0k)k\ _1k}{k\ _0kf^0(k\ _1k)}:$$

Next we proceed with the computation of the second term, namely $P_?(\ _1)$. Since $\ _1\ 2\ S$,

$$y = X\ _1 = X(P(\ _1) + P_?(\ _1)) = XP_?(\ _1):$$

Thus, $P_?(\ _1)\ 2\ S$. Similarly, since $\ 2\ S$, $P_?(\ )\ 2\ S$, meaning that it is a feasible point for the minimization problem (6.2.11). Moreover, since is a solution of problem (6.2.11), we have that

$$k\ k^2\ \ kP_?(\ )k^2\ \ kP_?(\ )k^2 + kP(\ )k^2 = k\ k^2:$$

Therefore, we conclude that $P(\ ) = 0$ and so that $\ 2\ \ker(X)^?$. Moreover, since $X\ \ = y = X\ _1$, we obtain that $\ _1\ 2\ \ker X$. Then, since $\ 2\ \ker(X)^?$,

$$8p\ 2\ \ker(X)^?\ \ hp\ \ \ j\ _1\ \ \ i = 0:$$

This, according to [15, Theorem 3.16], implies that $P_?(\ _1) =\ $. Then,

$$_1 = P(\ _1) + P_?(\ _1) = P(\ _0)\frac{f^0(k\ _0k)k\ _1k}{k\ _0kf^0(k\ _1k)} +\ :$$

□

Note that, compared to the analogous result in Corollary 6.2.14, this is a fixed point equation, in the sense that we cannot compute the norm of $\ _1$ explicitly.

In the following two examples we explore the flexibility of our formulation. By choosing appropriately $h$ and $f$, we obtain the two well-known reparameterizations corresponding to polynomial and exponential weight normalization [89, 118, 139].

**Example 6.3.6. Polynomial Weight normalization.** Let Assumption A1 be satisfied, and consider the dynamics defined in (6.3.1)-(6.3.2) corresponding to

$$h(\ ) =\ ^L;$$

with an initialization such that $kw_0k = 1$ and $h(\ _0) > 0$, so that Assumption A3 holds. Let $f : [0; +1)$ be such that

$$f^0(s) = \exp\left(\frac{s^{2=L}}{2L}\right)\ \text{for every } s > 0;$$

so that equation (6.3.6) is satisfied. Note that $f$ is twice differentiable on $(0; +1)$, $f^0(s) > 0$, $f^{00}(s) > 0$ for all $s > 0$, and $1=f^{00}$ is locally Lipschitz. Define $g(s) = sf^0(s)$ for every $s\ \ 0$ and let $r$ be as in Assumption A4. Since $g$ is increasing, $g(s)\ \ g(r)$ for every $s\ \ r$. Then Theorem 6.3.3 and Lemma 6.3.1 imply that Assumptions A2 and A5 are satisfied

with $U = \mathbb{R}^p \setminus \{0\}$ and the differential equation in the original variable corresponding to the gradient flow on the variables $(\ ; w)$ is

$$\dot{\ }(t) = \quad k\ (t)k^2 \quad \mathrm{Id} \quad \frac{(t)\ ^{\top}(t)}{k\ (t)k^2} \quad r\,L(\ (t))$$

$$L^2 \frac{(t)\ ^{\top}(t)}{k\ (t)k^{2-L}}\, r\,L(\ (t)): \qquad t > 0$$

This equation has a unique solution $(t)$ defined on $[0; +1)$ for the initial condition $_0 = h(\ _0)w_0$. Note that, from Lemma 6.2.6, along the trajectory $(t)$, the scalar function $G$ is bounded below by a strictly positive constant. This allows us to avoid the undesired stationary point zero. To derive the implicit bias, we need to apply Theorem 6.2.8. To do so, it remains to show that assumption A5 holds. Let $z \in S \setminus U$. We next prove that $D_F(z; \ )$ is coercive. It follows from convexity of $f$ that for every $s > 0$

$$f(s) + f'(s)(k\ \ k \quad s) \quad f(k\ \ k):$$

Dividing the previous inequality by $k\ \ k$ yields

$$f'(s) = \lim_{k\ k \to 1} \frac{f(s) + f'(s)(k\ \ k \quad s)}{k\ \ k} \qquad \liminf_{k\ k \to 1} \frac{f(k\ \ k)}{k\ \ k} = \liminf_{k\ k \to 1} \frac{F(\ )}{k\ \ k}:$$

Since $s > 0$ is arbitrary and $f'(s) \to +1$ as $s \to +1$, we deduce that $F$ is supercoercive. It follows from [14, Lemma 7.3 (viii)] that the $D_F(z; \ )$ is coercive for every $z \in U$, which implies assumption A5. The implicit bias is given in Theorem 6.2.8 and specialized in Corollary 6.3.5. For the weight normalization proposed by [118], which corresponds to the choice $L = 1$, in the setting of Corollary 6.3.5, we obtain

$$_1 = \quad + P(\ _0)\frac{k\ \ _1 k}{k\ \ _0 k} e^{\frac{k\ _0 k^2 \quad k\ \ _1 k^2}{2}}; \tag{6.3.14}$$

where we recall that $P$ is the projection operator on the kernel of $X$. Note that when $_0 \in \mathrm{ran}(X^{\top}) = \ker(X)^{?}$, then $_1$ is equal to the minimal norm solution . Our results guarantee that the trajectory exists for $t \to +1$, which is something usually assumed, and not proved. The implicit bias for this example has been studied in [139] and [89] for the least squares and the logistic loss, respectively. In [139], the authors analyze the convergence to a stationary point (zero loss or zero norm) in the continuous and discrete settings, and they derive the expression in (6.3.14). The squared weight normalization corresponds to the case $L = 2$. For this choice, we can explicitly compute the function $f$. Indeed, let

$$h(\ ) = \ ^2; \quad f(s) = 4e^{\frac{s}{4}}; \quad \text{and} \quad g(s) = se^{\frac{s}{4}}:$$

In addition, in the setting of Corollary 6.3.5, we get that

$$_1 = \quad + P(\ _0)\frac{k\ \ _1 k}{k\ \ _0 k} e^{\frac{k\ _0 k \quad k\ \ _1 k}{4}}:$$

**Example 6.3.7. Exponential weight normalization.** Under Assumption A1, consider the dynamic defined in (6.3.1)-(6.3.2) with an initialization $(\ _0; w_0)$ such that $kw_0k = 1$ and $L(\ _0) < L(0)$, where $_0 = h(\ _0)w_0$. Let, for some $> 0$,

$$h(\ ) = e^{-}; \quad f(s) = \frac{s^{\ ^2+1}}{\ ^2 + 1}; \quad \text{and} \quad g(s) = s^{\ ^2+1}:$$

On the interval $(0; +1)$, we have $f' > 0$, $f'' > 0$ and $1/f''$ locally Lipschitz. In addition, equation (6.3.6) is satisfied, $g(s) = sf'(s)$ for all $s \quad 0$ and $g(s) > r^{2+1}$ for every $s > r$. Then it follows from Theorem 6.3.3 that $(t) = h( (t))\frac{w(t)}{kw(t)k}$, where the trajectories of $w(t)$ and $(t)$ follow the dynamical system (6.3.1)-(6.3.2), solves the differential equation

$$\dot{}(t) = \quad k \ (t)k^2\mathrm{Id} \quad 1 \quad \frac{1}{2} \quad (t) \ ^{>}(t) \quad r\,L( \ (t)): \qquad t > 0$$

To apply Theorem 6.2.8, it remains to prove assumption A5, that requires the existence of $z \ 2 \ S \setminus U$ such that $D_F(z; \ )$ is coercive. Indeed,

$$\begin{aligned} D_F(z; \ ) &= \frac{kzk^{2+1}}{2+1} \quad \frac{k \ k^{2+1}}{2+1} \quad k \ k^{2} \ ^{1}h \ j\,z \quad i \\ &= \frac{kzk^{2+1}}{2+1} + \frac{k \ k^{2+1}}{2+1} \quad 2 \quad \frac{2+1}{k \ k^2}h \ j\,zi \\ &\quad \frac{kzk^{2+1}}{2+1} + \frac{k \ k^{2+1}}{2+1} \quad 2 \quad \frac{2+1}{k \ k}kzk \end{aligned}$$

and so $D_F(z; \ )$ is coercive with respect to for every $z \ 2 \ U$. Theorem 6.2.8 then yields $\lim_{t \,!\, +1} (t) = \ _1$, where

$$_1 \ 2 \operatorname*{argmin}_{z2S} \frac{kzk^{2+1}}{2+1} \quad k \ _0k^{2} \ ^{1}h \ _0 \,j\,zi + \int_0^{+1} k \ (t)k^{2+1}h\,r\,L( \ (t))\,j\,z \quad (t)i\,dt:$$

Moreover, in the setting of Corollary 6.3.5, we obtain that

$$_1 = \quad + P( \ _0)\frac{k \ _0k^{2} \ 1}{k \ _1k^{2} \ 1}:$$

Note that, when $= \pm\frac{1}{2}$, the mirror function $F$ is the same function obtained in Example 5.3.1, showing that different reparametrizations may have the same implicit bias.

## 6.3.1 Weight normalization of a fully connected network

In this subsection, we combine the two main examples presented in this chapter: fully connected linear network and weight normalization. The reparameterization involves decomposing the vector into the product of three terms: a scalar that represents the norm of , and the product of a matrix by a vector, product that represents the direction of with unitary norm.

Consider the over-parameterization $= h( )\frac{Ww}{kWwk}$, where $h\colon \mathrm{R} \ ! \ \mathrm{R}_+$ is a given function, while $2 \ \mathrm{R}$, $W \ 2 \ \mathrm{R}^{p \ m}$ and $w \ 2 \ \mathrm{R}^m$ are the new variables. The associated gradient flow on the reparameterization is given by the following dynamical system:

$$\underline{w}(t) = \quad r_w L( \ (t)) = \quad \frac{h( \ (t))}{kW(t)w(t)k}W^{>}(t) \quad \mathrm{Id} \quad \frac{(t) \ ^{>}(t)}{k \ (t)k^2} \quad r\,L( \ (t)) \qquad (6.3.15)$$

$$W(t) = \quad r_W L( \ (t)) = \quad \frac{h( \ (t))}{kW(t)w(t)k} \quad \mathrm{Id} \quad \frac{(t) \ ^{>}(t)}{k \ (t)k^2} \quad r\,L( \ (t))w^{>}(t) \qquad (6.3.16)$$

$$\dot{}(t) = \quad r \quad L( \ (t)) = \quad h'( \ (t)) \quad \frac{(t)}{k \ (t)k} \,j\, r\,L( \ (t)) \ : \qquad (6.3.17)$$

We consider, for simplicity of computations, an initialization $( \ _0; W_0; w_0)$ such that $kW_0w_0k = 1$ and $w_0w_0^> = W_0^>W_0$. In the next theorem, we derive a time warping mirror flow on as in (6.2.1) from the gradient flow defined in (6.3.15)-(6.3.17) over $w$, $W$, and , respectively. We then apply Theorems 6.2.4 and 6.2.8, providing the existence and uniqueness of solution for the dynamical system and convergence in value and in trajectory. In the case where the loss is a composition of a convex function and a linear operator, the limit point can be characterized as the Bregman projection on the solution set for a given mirror map, which depends on the choice of the reparameterization.

**Theorem 6.3.8.** Consider the overparameterized model $= h( ) \frac{Ww}{kWwk}$ and the dynamic defined in (6.3.15)-(6.3.17) with initialization such that $kW_0w_0k = 1$ and $w_0w_0^> = W_0^>W_0$. Assume that assumptions A1 and A3 holds for $_0 = h( \ _0) \frac{W_0w_0}{kW_0w_0k}$. Assume that there exists a function $f : [0; +1) \ ! \ [0; +1)$ twice differentiable on $(0; +1)$ such that $f^{00}(s) > 0$ for every $s > 0$, $1=f^{00}$ is locally Lipschitz on $(0; +1)$ and satisfying the following equality

$$(h^0( \ ))^2 = \frac{f^0(h( \ ))h( \ )}{f^{00}(h( \ ))}: \tag{6.3.18}$$

Suppose in addition that there exists $c > 0$ such that $f^0(s) > c=s$ for every $s > r$ ($r$ is defined in Assumption A3) and define $g : \ R_+ \ ! \ R$ as $g(s) = sf^0(s)$. Then the assumptions A2 and A4 are satisfied, and for every $t > 0$, $kW(t)w(t)k = 1$, $h( \ (t)) > 0$, and $(t)$ solves the following dynamical system, for $t \ 0$

$$r^2(f(k \ (t)k)) \ _-(t) = \ g(k \ (t)k) \, r \ L( \ (t)); \tag{6.3.19}$$

i.e., the dynamic of $(t)$ satisfies (6.2.1), with $F( \ ) = f(k \ k)$ and $G(k \ k) = g(k \ k)$.

Proof. If we multiply (6.3.15) by $w^>(t)$ on the right and (6.3.16) by $W^>(t)$ on the left, we obtain that

$$W^>(t)W(t) = w(t)w^>(t): \tag{6.3.20}$$

Adding the transpose of (6.3.20) yields

$$W^>(t)W(t) + W^>(t)W(t) = w(t)w^>(t) + w(t)w^>(t):$$

Since $W^>(0)W(0) = w^>(0)w^>(0)$, we obtain

$$W^>(t)W(t) = w(t)w^>(t): \tag{6.3.21}$$

Multiplying (6.3.21) by $w^>(t)$ on the left and by $w(t)$ on the right, we have that $kw(t)k^4 = kW(t)w(t)k^2$, for every $t$. Analogously, multiplying by $W(t)$ the equation (6.3.21) on the left and by $W^>(t)$ on the right we deduce that

$$(W(t)W^>(t))^2 = \frac{(t) \ ^>(t)}{k \ (t)k^2} kW(t)w(t)k^2: \tag{6.3.22}$$

Thus, $kw(t)k^2 = kW(t)w(t)k$. In addition, since $W(t)W^T(t)$ is positive semidefinite and symmetric, it is the unique square root of the right hand side of (6.3.22) and therefore

$$W(t)W^>(t) = \frac{(t) \ ^>(t)}{k \ (t)k^2} kW(t)w(t)k;$$

since the right hand side of (6.3.22) is a rank one matrix. Moreover, note that

$$\begin{aligned}
W(t)\dot{w}(t) &= W(t)w(t) + W(t)\underline{w}(t) \\
&= \frac{h(\ (t))}{kW(t)w(t)k}\ kw(t)k^2\mathrm{Id} + W(t)W^{>}(t) \quad \mathrm{Id} \quad \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad r\,L(\ (t)) \\
&= h(\ (t))\ \ \mathrm{Id} + \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad \mathrm{Id} \quad \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad r\,L(\ (t)) \\
&= h(\ (t))\ \ \mathrm{Id} \quad \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad r\,L(\ (t))
\end{aligned}$$

and

$$\begin{aligned}
@_t\frac{kW(t)w(t)k^2}{2} &= \Big\langle W(t)w(t)\ j\ W(t)w(t) + W(t)\underline{w}(t)\Big\rangle_F \\
&= \frac{(t)}{k\ (t)k}\ j\ W(t)w(t) + W(t)\underline{w}(t)\Big\rangle_F \\
&= \ ^{>}(t)\ \ \mathrm{Id} \quad \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad r\,L(\ (t)) = 0;
\end{aligned}$$

which implies that $kW(t)w(t)k = 1$ for every $t$. Then,

$$\begin{aligned}
\dot{}(t) &= \dot{}(t)h^{0}(\ (t))W(t)w(t) + h(\ (t))\dot{W}(t)w(t) + h(\ (t))W(t)\underline{w}(t) \\
&= h^{0}(\ (t))^2\frac{(t)\ ^{>}(t)}{k\ (t)k^2}\,r\,L(\ (t)) \quad k\ (t)k^2 \quad \mathrm{Id} \quad \frac{(t)\ ^{>}(t)}{k\ (t)k^2} \quad r\,L(\ (t)):
\end{aligned}$$

Equation (6.3.19) then follows from (6.3.18) and Lemma 6.3.1. □

**Remark 6.3.9.** Choosing $h$, $f$, and $g$ as in Examples 6.3.6 and 6.3.7, the assumptions of Theorem 6.3.8 are satisfied. Consequently, $(t)$ satisfies the properties derived in Theorem 6.2.4, Theorem 6.2.8 and Corollary 6.2.9.

## 6.3.2 Fully connected normalized linear network of depth 2

In this subsection, we study a new reparameterization of $(t)$. The reparametrization decomposes the vector into the product of a matrix times a unitary vector. Then, we apply vanilla gradient flow to update the vector and the matrix. The main difference with respect to the previous examples is that the unitary vector is multiplied by a matrix which encode the norm of , rather than a scalar. We name this parametrization connected normalized linear network of depth 2. More formally, we set $= W\frac{w}{kwk}$ where $W\ 2\ \mathrm{R}^{p\ \ m}$ and $w\ 2\ \mathrm{R}^m$. We define the gradient flows of $W$ and $w$ as:

$$\dot{W}(t) = \ r_W L\ \ W(t)\frac{w(t)}{kw(t)k} = \ r\,L(\ (t))\frac{w^{>}(t)}{kw(t)k} \tag{6.3.23}$$

$$\begin{aligned}
\underline{\dot{w}}(t) &= \ r_w L\ \ W(t)\frac{w(t)}{kw(t)k} \\
&= \frac{1}{kw(t)k}\ \ \mathrm{Id} \quad \frac{w(t)w^{>}(t)}{kw(t)k^2} \quad W^{>}(t)\,r\,L(\ (t)):
\end{aligned} \tag{6.3.24}$$

The next theorem shows that the corresponding trajectory in is a preconditioned gradient flow as in (6.2.1). It turns out that, for some specific initialization, the corresponding trajectory $(t)$ is just vanilla gradient flow. We then apply Theorems 6.2.4 and 6.2.8 to get existence and uniqueness of the trajectory of the flow, optimization properties and

convergence to a specific solution asymptotically for $t$ approaching infinity.

**Theorem 6.3.10.** Consider the dynamic defined in (6.3.23)-(6.3.24) initialized with $(W_0; w_0)$ such that $kw_0k = 1$, set $_0 = W_0w_0=kw_0k$ and suppose that $L( _0) < L(0)$. Then:

1. The variable $(t)$ is governed by the following differential equation:

$$-(t) = (\mathrm{Id} + S(t)) \; r\,L( (t)); \tag{6.3.25}$$

where $S(t) := W(t)W^{>}(t)$ $(t)$ $^{>}(t)$ 0.

2. If $S_0 = W_0W_0^{>}$ $_0$ $_0^{>} = 0$, $(t)$ is governed by the following differential equation:

$$-(t) = r\,L( (t)):$$

Proof. We provide the proof of Theorem 6.3.10, which is divided in several steps.

1. We first prove that $kw(t)k$ is constant. Multiplying (6.3.24) by $w^{>}(t)$ we obtain

$$@_t kw(t)k^2 = 2hw(t) \; j \; \underline{w}(t)i = 0:$$

Then, if $kw(0)k = 1$, we get that $kw(t)k = 1$. Equations (6.3.23)-(6.3.24) and the derivative of the product's rule yield

$$\begin{aligned}
-(t) &= W(t)\frac{w(t)}{kw(t)k} + W(t) \quad \frac{\underline{w}(t)}{kw(t)k} \quad \frac{w(t)w^{>}(t)\underline{w}(t)}{kwk^3} \\
&= W(t)\frac{w(t)}{kw(t)k} + W(t)\frac{\underline{w}(t)}{kw(t)k} \\
&= r\,L( (t))\frac{w^{>}(t)w(t)}{kw(t)k^2} \\
&\quad \frac{W(t)}{kw(t)k^2} \quad \mathrm{Id} \quad \frac{w(t)w^{>}(t)}{kw(t)k^2} \quad W^{>}(t)\, r\,L( (t)) \\
&= \mathrm{Id} + W(t)W^{>}(t) \quad (t) \quad ^{>}(t) \quad r\,L( (t)) \\
&= (\mathrm{Id} + S(t)) \; r\,L( (t));
\end{aligned}$$

which establishes equation (6.3.25). Moreover, note that $S(t) = W(t)W^{>}(t)$ $(t)$ $^{>}(t)$ 0. Indeed,

$$\begin{aligned}
(8y \; 2 \; \mathrm{R}^p) \quad y^{>}(WW^{>} \quad ^{>})y &= y^{>} \quad WW^{>} \quad W\frac{w}{kwk}\frac{w^{>}}{kwk}W^{>} \quad y \\
&= kW^{>}yk^2 \quad \frac{w^{>}}{kwk}W^{>}y \quad ^2 \\
&\quad kW^{>}yk^2 \quad \frac{w^{>}}{kwk} \quad ^2 \; kW^{>}yk^2 \\
&= 0:
\end{aligned}$$

2. The proof is divided in three parts.

(a) **Derivation of the evolution of $S(t) = W(t)W^\top(t)$ $(t)$ $^\top(t)$:**

$$S(t) = W(t)W^\top(t) \quad -(t) \ ^\top(t) + W(t)W^\top(t) \quad (t) \ -^\top(t): \tag{6.3.26}$$

It follows from (6.3.23) that

$$W(t)W^\top(t) = \quad r L( \ (t)) \ ^\top(t): \tag{6.3.27}$$

Multiplying (6.3.25) on the right by $^\top$, (6.3.27) implies

$$\begin{aligned} -(t) \ ^\top(t) =& W(t)W^\top(t) \\ & W(t)W^\top(t) \quad (t) \ ^\top(t) \quad r L( \ (t)) \ ^\top(t) \\ =& W(t)W^\top(t) \quad S(t) \, r L( \ (t)) \ ^\top(t): \end{aligned} \tag{6.3.28}$$

Analogously

$$\begin{aligned} (t) \, -^\top(t) =& \quad r L( \ (t))^\top (I + S(t)) \\ =& \ W(t)W^\top(t) \quad (t) \, r L( \ (t))^\top S(t): \end{aligned} \tag{6.3.29}$$

It follows from (6.3.26), (6.3.27), (6.3.28) and (6.3.29) that

$$S(t) = S(t) \, r L( \ (t)) \ ^\top(t) + \ (t) \, r L( \ (t))^\top S(t): \tag{6.3.30}$$

(b) **Show that $S(t) = 0$ in a neighborhood of zero:** Let us denote by $k \ k_F$ the frobenius norm of a matrix. Then $\frac{1}{2}\partial_t kS(t)k_F^2 = hS(t)S(t)i_F = \mathrm{Tr}(S(t)S(t))$. Equation (6.3.30) and the linearity and cyclical property of the trace imply

$$\begin{aligned} \frac{1}{2}\partial_t kS(t)k_F^2 =& \ \mathrm{Tr} \ \ S(t) \, r L( \ (t)) \ ^\top(t) S(t) \\ & + \mathrm{Tr} \quad (t) \, r L( \ (t))^\top S^2(t) \\ =& \ \mathrm{Tr} \quad r L( \ (t)) \ ^\top(t) S^2(t) \\ =& \ + \mathrm{Tr} \quad (t) \, r L( \ (t))^\top S^2(t) \\ =& \ \mathrm{Tr} \quad r L( \ (t)) \ ^\top(t) + \ (t) \, r L( \ (t))^\top \ S^2(t) \\ =& \ \Big\langle r L( \ (t)) \ ^\top(t) + \ (t) \, r L( \ (t))^\top \ j \, S^2(t) \Big\rangle_F \\ & k \, r L( \ (t)) \ ^\top(t) + \ (t) \, r L( \ (t))^\top k_F \, kS^2(t)k_F \\ & 2k \, r L( \ (t))kk \ (t)kkS(t)k_F^2; \end{aligned} \tag{6.3.31}$$

where in the last inequality we used that, for every $u \ 2 \ \mathbb{R}^p$ and $v \ 2 \ \mathbb{R}^p$, $kuv^T k_F = kukkvk$.

We next prove that the term $k \, r L( \ (t))kk \ (t)k$ is bounded, to apply Grönwall's inequality. We proceed computing the derivative of the norm along the trajectory:

$$\begin{aligned} (8z \ 2 \ S) \ \ \partial_t k \ (t) \quad zk^2 = & \quad 2h(\mathrm{Id} + S(t)) \, r L( \ (t)) \, j \quad (t) \quad z i \\ & 2hS(t) \, r L( \ (t)) \, j \quad (t) \quad z i \\ & + 2 \, (L(z) \quad L( \ (t)) : \end{aligned}$$

If $L(W_0 w_0 = k w_0 k) = L$, then $r L(\ _0) = 0$. So the norm is constant and the trajectory remains bounded. Otherwise, $L(z) \quad L(\ _0) < 0$ and, from $S_0 = 0$ we derive

$$(t \mapsto @_t k\ (t)) \quad zk^2)(0) \quad L(z) \quad L(\ _0) < 0:$$

Since the norm and $(t)$ are continuous, there exists an interval $[0; t_1]$ with nonempty interior such that $k\ (t) \quad zk^2$ is decreasing and bounded by $k\ _0 \quad zk^2$. This implies that $fk\ (t)k : t \in [0; t_1]g$ is bounded. Similarly, since $r L$ is locally Lipschitz, $k r L(\ (t))k$ is bounded on $[0; t_1]$ (see, for instance, [91, Theorem 27.4]). Define, for every $t \in [0; t_1]$;

$$G(t) = e^{\ 2 \int_0^t k r L(\ (s)) k k^{>}(s) k d s} > 0;$$

then, for every $t \in [0; t_1]$, (6.3.31) implies that

$$@_t(kS(t)k_F^2 G(t)) = G(t)(@_t kS(t)k_F^2 \quad 2k r L(\ (t))kk^{>}(t)kkS(t)k_F^2) \quad 0$$

and therefore $kS(t)k_F^2 \quad kS(0)k_F^2 e^{2 \int_0^t k r L(\ (s)) k k^{>}(s) k d s} = 0$.

(c) **Extend the proof, for every $t > 0$:** For every $t \in [0; t_1]$, we obtain

$$@_t k\ (t) \quad zk^2 \quad L(z) \quad L(\ (t)) \quad 0:$$

Now define $t = \sup fT \quad 0 j 8t \quad T\ j\ S(t) = 0g$. The previous step implies $T \quad t_1$. Suppose by contradiction that $t < +1$, i.e., $S$ is null only in a finite interval. Since $S(t)$ is continuous $S(t\ ) = 0$ and we can proceed analogously to the case when $S(0) = 0$ and prove that there exist a $t_2 \quad t$ such that, for every $t \in [t\ ; t_2)$, $kS(t)k_F^2 \quad kS(t\ )k_F^2 e^{2 \int_t^{>} k r L(\ (s)) k k^{>}(s) k d s} = 0$. This leads to a contradiction with the maximality of $t$ and we conclude that $t = +1$ and $S(t) = 0$, for every $t \quad 0$. It follows from (6.3.25) that

$$\dot{}(t) = \quad r L(\ (t));$$

and 2 follows.

□

**Remark 6.3.11.** Consider the dynamics defined in (6.3.23)-(6.3.24) and suppose that A1 holds. In addition, consider an initialization such that $kw(0)k = 1$, $W(0)W^{>}(0) \quad W(0)w(0)w^{>}(0)W^{>}(0) = 0$ and $L(\ _0) < L(0)$. Moreover, by Theorem 6.3.10, Assumption A2-A5 is satisfied with $F(\ ) = \frac{k\ k^2}{2}$ and $G(\ ) = 1$. Note that, in this case, the assumption A2 holds with $U = \mathbb{R}^p$. Consequently, $(t)$ can be characterized as described in Theorem 6.2.4 and Theorem 6.2.8.

## 6.4 Conclusion and Future Work

In this chapter, we studied a unified framework encompassing several reparemeterization schemes from the existing literature. We provided assumptions that guarantee the well-posedness of the dynamical system and convergence of the trajectory. This convergence is both in functional value and in the variable itself, that tends to a minimizer of the loss function avoiding the additional stationary point caused by the reparametrization. In the case where the function is composed by a linear operator, an explicit expression of the implicit bias is given. This comprehensive approach allows us to gain insights into the behavior of the time warped mirror flow, providing a more comprehensive understanding

of its properties.

We also provide a criterion to determine, for a certain function that depends only on the norm, a suitable weight normalization parameterization. Finally, we apply the obtained results to a general multi-neuron fully connected linear network of depth 2, different schemes of weight normalization, and two different extensions of these two previous models.

In the future, we would like to extend this work to models that are more closely related to neural networks and that introduce non-smooth activation functions. Possible extensions are: 1) Extend the results to more general models, such as neural networks; 2) Study new ways to train the overparameterization, for example Stochastic Gradient Descent or second-order methods; 3) Extend this approach to differential inclusions; 4) Study the problem in the matricial case.

# APPENDIX A

# Appendix for Chapter 6

## A.1 Examples

The following three examples have been adapted from [43, Theorem 1], [6, Theorem 1], and [7, Theorem 4.1], respectively. The main objective is to illustrate how to use equation (5.1.3) and its limitations. First, in example A.1.1, we study the case when the reparameterization is separable and find the mirror map is trivial (calculate an integral twice). Second, in example A.1.2, we give a scenario where the derivation of the mirror flow is not straightforward and requires a technical effort. Lastly, in Example 5.3.1, we show that the derivation of Mirror Flow is not possible at all, but it is still possible to find a mirror map $F$ to express the dynamics of $(t)$ as mirror flow multiplied by a positive function $G$, emphasizing the limitations of equation (5.1.3) justifying our choice of model.

**Example A.1.1.** Consider a $L$-layer's overparameterized vector factorization, which can be mathematically expressed as $= q( ) = {}^{L} 2 \mathsf{R}^{p}$ with $L \quad 2$. We define the flow over as:

$$-(t) = \quad r \; L(q( \; (t))) = \quad L \, r \, L( \; (t)) \qquad {}^{(L \;\; 1)}(t);$$

where the last equality comes from the chain rule.

Note that $q$ satisfies (5.1.3) with $F$ equal to:

$$F: \quad \nexists \begin{cases} \frac{1}{4} \sum_{n=1}^{p} ( \; _n \log( \; _n) \quad \; _n) & \text{If } L = 2 \\ \frac{1}{2L(2 \;\; L)} \sum_{n=1}^{p} n^{\frac{2}{L}} & \text{If } L > 2. \end{cases} ; \tag{A.1.1}$$

Indeed, if we compute $J_q( \; )J_q( \; )^T$, we get:

$$J_q( \; )J_q( \; )^T = \; q^{0}( \; ) \quad {}^{2} = L^{2} \quad {}^{(2L \;\; 2)} = L^{2} \quad {}^{(2 \;\; \frac{2}{L})} = \quad r^{2}F( \; ) \quad {}^{1}:$$

Then, $(t)$ follows the dynamic:

$$-(t) = \qquad r^{2}F( \; (t)) \quad {}^{1} \; r \, L( \; (t)):$$

The previous flow coincides with the one derived in [43, 135]. Moreover, if $L( \; ) = kX \qquad yk^{2}$, for $X \; 2 \; \mathsf{R}^{d \;\; p}$ and $y \; 2 \; \mathsf{R}^{d}$, then by [77, Theorem 4.17], we get:

$$(t) \;\; ! \quad _1 \text{ and } X \quad _1 = y \qquad \text{(Convergence and Feasibility)} \tag{A.1.2}$$

$$r \, F( \; _1) \quad r \, F( \; _0) \; 2 \; \mathrm{ran}(X^{>}) \qquad \text{(Stationarity point)} \tag{A.1.3}$$

which are the KKT conditions of

$$\min_{Xz=y} \quad D_F(z;\ _0): \tag{A.1.4}$$

For a deeper discussion about the KKT conditions we refer the reader to [113, Section 28].

**Example A.1.2.** Consider the Two layers vector-vector parameterization; i.e., $\ = q(\ )$ with $\ = (\ _1;\ _2)\ 2\ \mathrm{R}^k\quad \mathrm{R}^k$ and $q(\ ) = \ _1\quad \ _2$, which is indeed an overparameterization because there are more learnable parameters than variables to predict. Then, the gradient flow on the reparameterization is given by

$$\begin{aligned} \dot{\ }_1(t) &= \quad r\ _1 L(\ _1(t)\quad \ _2(t)) = \quad r\, L(\ (t))\quad \ _2(t); \\ \dot{\ }_2(t) &= \quad r\ _2 L(\ _1(t)\quad \ _2(t)) = \quad r\, L(\ (t))\quad \ _1(t) \qquad \text{with } \ _1{}^2(0)\quad \ _2{}^2(0) = \ ^!; \end{aligned}$$

where $^!\ 2\ \mathrm{R}^p$. Let us compute the Jacobian of $q$

$$J_q(\ ) = [\mathrm{Diag}(\ _2)\quad \mathrm{Diag}(\ _1)] \text{ and } \quad J_{q(\ )}(\ )J^{>}_{q(\ )}(\ ) = \mathrm{Diag}\quad \ _1{}^2 + \ _1{}^2\ :$$

Which is not straightforward to express the previous in terms of $\ $. However, if we multiply the equation of $\dot{\ }_1(t)$ component-wise by $\ _1(t)$ and we do the same for $\dot{\ }_2(t)$ by $\ _2(t)$, we get:

$$\ _1(t)\quad \dot{\ }_1(t) = \quad r\quad L(\ (t))\quad \ _1(t)\quad \ _2(t) = \ _2(t)\quad \dot{\ }_2(t):$$

From the previous equality and the initialization $\ _1{}^2(0)\quad \ _2{}^2(0) = \ ^!$, we deduce that, for every $t\quad 0$,

$$\ _1{}^2(t) = \ _2{}^2(t) + \ ^!\ : \tag{A.1.5}$$

Multiplying the latter component-wise by $\ _2{}^2(t)$ and recalling that $\quad ^2(t) = \ _1(t)\quad \ _2(t)$, we obtain the following biquadratic equation:

$$\ _2{}^4(t) + \ ^!\quad \ _2{}^2(t)\qquad ^2(t) = 0:$$

Since $\ _2{}^2(t)\quad 0$, the previous equation has only one positive solution and from (A.1.5), we have that:

$$\ _2{}^2(t) = \frac{\ ^! + \sqrt{\ ^!{}^2 + 4\quad ^2(t)}}{2} \quad \text{and} \quad \ _1{}^2(t) = \frac{\ ^! + \sqrt{\ ^!{}^2 + 4\quad ^2(t)}}{2}:$$

Consider the following entropy function:

$$F(\ ) := \frac{1}{4}\sum_{n=1}^{\not K} \quad 2\ _n \log\quad 2\ _n + \sqrt{\ ^!{}^2 + 4\ \tfrac{2}{n}}\qquad \sqrt{\ ^!{}^2 + 4\ \tfrac{2}{n}}\ :$$

It follows from the definition of the derivative of the product and the computation of the gradient that

$$\begin{aligned} \dot{-}(t) = \dot{\ }_1(t)\quad \ _2(t) + \ _1(t)\quad \dot{\ }_2(t) &= \quad \frac{\ _1{}^2 + \ _2{}^2}{\sqrt{\ ^!{}^2 + 4\ ^2}}\quad r\, L(\ (t)) \\ &= \quad r\ ^2 F(\ (t))\quad {}^1\, r\, L(\ (t)): \end{aligned}$$

Finally by [77, Theorem 4.17], we have convergence to feasible stationary point (equations (A.1.2) and (A.1.3)), and we can characterize the implicit bias (A.1.4).

**Example A.1.3.** Consider a multi-neuron fully connected linear network of depth 2; i.e., $= q( )$ with $= (W; w)$ $2$ $\mathbb{R}^{p}$ $^{m}$ $\mathbb{R}^{m}$ and $q( ) = Ww$. The gradient flow on the reparameterization is then given by

$$\begin{cases} \dot{W}(t) &= -r_{W} L(W(t)w(t)) = -r L( (t))w^{>}(t) \\ \dot{w}(t) &= -r_{w} L(W(t)w(t)) = -W^{>}(t) r L( (t)) \end{cases} \tag{A.1.6}$$

with $W^{>}(0)W(0) = w(0)w^{>}(0)$.

Observe that $J_{q}(W; w)J_{q}^{>}(W; w) = kwk^{2}\mathrm{Id} + WW^{>}$ and therefore it is not clear how to express it in terms of . Proceeding in the same way as in [5], if we multiply the expression of $\dot{W}(t)$ on the left by $W^{>}(t)$ and the one of $\dot{w}(t)$ on the right by $w^{>}(t)$, we get that Who derived this expression first

$$W^{>}(t)\dot{W}(t) = -W^{>}(t) r \quad L( (t))w^{>}(t) = \dot{w}(t)w^{>}(t):$$

Adding to the previous equality the one obtained by transposing it, we have that,

$$\begin{aligned} \frac{d}{dt} \quad W^{>}(t)W(t) \quad &= \dot{W}^{>}(t)W(t) + W^{>}(t)\dot{W}(t) \\ &= \dot{w}(t)w^{>}(t) + w(t)\dot{w}^{>}(t) \\ &= \frac{d}{dt} \quad w(t)w^{>}(t) \quad : \end{aligned}$$

Then, from the latter equation and the initialization $W^{>}(0)W(0) = w(0)w^{>}(0)$, we deduce that, for every $t$ 0,

$$W^{>}(t)W(t) = w(t)w^{>}(t): \tag{A.1.7}$$

Multiplying by $w(t)$ on the right, by $w^{>}(t)$ on the left and recalling that $(t) = W(t)w(t)$, we have that $k \ (t)k^{2} = kw(t)k^{4}$ and so $k \ (t)k = kw(t)k^{2}$:

Analogously, multiplying (A.1.7) by $W(t)$ on the left and by $W^{>}(t)$ on the right, we deduce that $(W(t)W^{>}(t))^{2} = (t) \ ^{>}(t)$. Since both $W(t)W^{>}(t)$ and $(t) \ ^{>}(t)$ are symmetric positive semi-definite, equality $(W(t)W^{>}(t))^{2} = (t) \ ^{>}(t)$, implies

$$W(t)W^{>}(t) = \frac{(t) \ ^{>}(t)}{k \ (t)k}:$$

Equation (A.1.6) yields

$$\dot{}(t) = \dot{W}(t)w(t) + W(t)\dot{w}(t) = -r L( (t))kw(t)k^{2} \quad W(t)W^{>}(t) r L( (t)): \tag{A.1.8}$$

Plugging in (A.1.8) the expressions of $kw(t)k^{2}$ and $W(t)W^{>}(t)$ in terms of $(t)$, we get

$$\dot{}(t) = \quad k \ (t)k\mathrm{Id} + \frac{(t) \ ^{>}(t)}{k \ (t)k} \quad r L( (t)): \tag{A.1.9}$$

According to Lemma 2.4.5, there does not exist an entropy function $F$ such that the previous dynamical system can be written as a mirror flow, therefore the flow cannot be formulated as vanilla mirror flow and the results in [77, Theorem 4.6] cannot be applied. However, as derived in [7]

$$r^{2} \quad \frac{2}{3}k \ k^{3=2} \quad = k \ k \ ^{1=2} \quad \mathrm{Id} \quad \frac{^{>}}{2k \ k^{2}} \quad : \tag{A.1.10}$$

Then, multiplying (A.1.9) on the left by (A.1.10), we obtain the following time-warped dynamical system:

$$r^{2} \quad \frac{2}{3}k \;\; (t)k^{3=2} \quad \text{-}(t) = \quad k \;\; (t)k^{1=2}\, r\, L(\;\; (t)); \tag{A.1.11}$$

which is a special case of (6.2.1) with $F; G : \mathsf{R}^{p} \;\; ! \;\; \mathsf{R}$ given by $F(\;\;) = \frac{2}{3}k \;\; k^{3=2}$ and $G(\;\;) = k \;\; k^{1=2}$. which comprises a time-warping factor.

# Bibliography


[1] A. Alacaoglu, O. Fercoq, and V. Cevher, "On the convergence of stochastic primal-dual hybrid gradient," SIAM Journal on Optimization, vol. 32, no. 2, pp. 1288–1318, 2022.

[2] A. Ali, E. Dobriban, and R. Tibshirani, "The implicit regularization of stochastic gradient flow for least squares," in International conference on machine learning, PMLR, 2020, pp. 233–244.

[3] F. Alvarez, J. Bolte, and O. Brahic, "Hessian riemannian gradient flows in convex programming," SIAM journal on control and optimization, vol. 43, no. 2, pp. 477–501, 2004.

[4] E. Amid and M. K. Warmuth, "Reparameterizing mirror descent as gradient descent," Advances in Neural Information Processing Systems, vol. 33, pp. 8430–8439, 2020.

[5] S. Arora, N. Cohen, W. Hu, and Y. Luo, "Implicit regularization in deep matrix factorization," in NeurIPS, 2019, pp. 7413–7424.

[6] S. Arora, N. Cohen, and E. Hazan, "On the optimization of deep networks: Implicit acceleration by overparameterization," in International Conference on Machine Learning, PMLR, 2018, pp. 244–253.

[7] S. Azulay et al., "On the implicit bias of initialization shape: Beyond infinitesimal mirror descent," in International Conference on Machine Learning, PMLR, 2021, pp. 468–477.

[8] M. Bachmayr and M. Burger, "Iterative total variation schemes for nonlinear inverse problems," Inverse Problems, vol. 25, no. 10, pp. 105004, 26, 2009, ISSN: 0266-5611. DOI: 10.1088/0266-5611/25/10/105004.

[9] M. A. Bahraoui and B. Lemaire, "Convergence of diagonally stationary sequences in convex optimization," in 1-2, vol. 2, Set convergence in nonlinear analysis and optimization, Springer, 1994, pp. 49–61. DOI: 10.1007/BF01027092.

[10] A. B. Bakushinsky and M. Y. Kokurin, Iterative methods for approximate solution of inverse problems. Springer Science & Business Media, 2005, vol. 577.

[11] L. Barreira and C. Valls, Ordinary differential equations: Qualitative theory. American Mathematical Soc., 2012, vol. 137.

[12] P. L. Bartlett and M. Traskin, "Adaboost is consistent," the Journal of machine Learning research, vol. 8, pp. 2347–2368, 2007, ISSN: 1532-4435.

[13] F. Bauer, S. Pereverzev, and L. Rosasco, "On regularization algorithms in learning theory," Journal of complexity, vol. 23, no. 1, pp. 52–72, 2007.

[14] H. H. Bauschke, J. M. Borwein, and P. L. Combettes, "Essential smoothness, essential strict convexity, and legendre functions in banach spaces," Communications in Contemporary Mathematics, vol. 3, no. 04, pp. 615–647, 2001.

[15] H. H. Bauschke and P. L. Combettes, Convex analysis and monotone operator theory in Hilbert spaces (CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC), Second. Springer, Cham, 2017, pp. xix+619, With a foreword by Hédy Attouch, ISBN: 978-3-319-48311-5. DOI: 10.1007/978-3-319-48311-5.

[16] A. Beck and M. Teboulle, "Mirror descent and nonlinear projected subgradient methods for convex optimization," Oper. Res. Lett., vol. 31, no. 3, pp. 167–175, 2003, ISSN: 0167-6377. DOI: 10.1016/S0167-6377(02)00231-6.

[17] S. Becker, J. Bobin, and E. J. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," SIAM J. Imaging Sci., vol. 4, no. 1, pp. 1–39, 2011. DOI: 10.1137/090756855.

[18] M. Benning and M. Burger, "Error estimates for general fidelities," Electronic Transactions on Numerical Analysis, vol. 38, no. 44-68, p. 77, 2011.

[19] M. Benning and M. Burger, "Modern regularization methods for inverse problems," Acta Numer., vol. 27, pp. 1–111, 2018, ISSN: 0962-4929. DOI: 10.1017/s0962492918000016.

[20] G. Blanchard and N. Krämer, "Optimal learning rates for kernel conjugate gradient regression," in Proceedings of the 23rd International Conference on Neural Information Processing Systems-Volume 1, vol. 23, 2010, pp. 226–234.

[21] R. I. Boţ and T. Hein, "Iterative regularization with a general penalty term–theory and application to L1 and TV regularization," Inverse Problems, vol. 28, no. 10, p. 104 010, 2012, ISSN: 0266-5611. DOI: 10.1088/0266-5611/28/10/104010.

[22] L. M. Briceno-Arias, "Forward-douglas–rachford splitting and forward-partial inverse method for solving monotone inclusions," Optimization, vol. 64, no. 5, pp. 1239–1261, 2015.

[23] L. M. Briceno-Arias, P. L. Combettes, J.-C. Pesquet, and N. Pustelnik, "Proximal algorithms for multicomponent image recovery problems," Journal of Mathematical Imaging and Vision, vol. 41, no. 1-2, pp. 3–22, 2011.

[24] L. Briceño-Arias, J. Deride, and C. Vega, "Random activations in primal-dual splittings for monotone inclusions with a priori information," Journal of Optimization Theory and Applications, pp. 1–26, 2021.

[25] L. Briceño-Arias and S. López Rivera, "A projected primal-dual method for solving constrained monotone inclusions," J. Optim. Theory Appl., vol. 180, no. 3, pp. 907–924, 2019, ISSN: 0022-3239. DOI: 10.1007/s10957-018-1430-2.

[26] L. M. Briceño-Arias, "A Douglas-Rachford splitting method for solving equilibrium problems," Nonlinear Anal., vol. 75, no. 16, pp. 6053–6059, 2012, ISSN: 0362-546X. DOI: 10.1016/j.na.2012.06.014. [Online]. Available: https://doi.org/10.1016/j.na.2012.06.014.

[27] M. Burger, E. Resmerita, and L. He, "Error estimation for Bregman iterations and inverse scale space methods in image restoration," Computing, vol. 81, no. 2-3, pp. 109–135, 2007, ISSN: 0010-485X. DOI: 10.1007/s00607-007-0245-z.

[28] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," SIAM J. Optim., vol. 20, no. 4, pp. 1956–1982, 2010, ISSN: 1052-6234. DOI: 10.1137/080738970.

[29] J.-F. Cai, S. Osher, and Z. Shen, "Linearized Bregman iterations for frame-based image deblurring," SIAM J. Imaging Sci., vol. 2, no. 1, pp. 226–252, 2009. DOI: 10.1137/080733371.

[30] L. Calatroni, G. Garrigos, L. Rosasco, and S. Villa, "Accelerated iterative regularization via dual diagonal descent," SIAM J. Optim., vol. 31, no. 1, pp. 754–784, 2021, ISSN: 1052-6234. DOI: 10.1137/19M1308888.

[31] E. Candes, "Matrix completion with noise," Proc. IEEE, vol. 98, no. 6, pp. 925–936, 2010.

[32] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" IEEE Trans. Inform. Theory, vol. 52, no. 12, pp. 5406–5425, 2006, ISSN: 0018-9448. DOI: 10.1109/TIT.2006.885507.

[33] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," Found. Comput. Math., vol. 9, no. 6, pp. 717–772, 2009, ISSN: 1615-3375. DOI: 10.1007/s10208-009-9045-5.

[34] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Trans. Inform. Theory, vol. 52, no. 2, pp. 489–509, 2006, ISSN: 0018-9448. DOI: 10.1109/TIT.2005.862083.

[35] A. Cauchy et al., "Méthode générale pour la résolution des systemes d'équations simultanées," Comp. Rend. Sci. Paris, vol. 25, no. 1847, pp. 536–538, 1847.

[36] Y. Censor, P. P. Eggermont, and D. Gordon, "Strong underrelaxation in kaczmarz's method for inconsistent systems," Numerische Mathematik, vol. 41, pp. 83–92, 1983.

[37] A. Chambolle, "An algorithm for total variation minimization and applications," in 1-2, vol. 20, Special issue on mathematics and image analysis, 2004, pp. 89–97. DOI: 10.1023/B:JMIV.0000011320.81911.38.

[38] A. Chambolle, M. J. Ehrhardt, P. Richtárik, and C.-B. Schönlieb, "Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications," SIAM J. Optim., vol. 28, no. 4, pp. 2783–2808, 2018, ISSN: 1052-6234. DOI: 10.1137/17M1134834.

[39] A. Chambolle and P.-L. Lions, "Image recovery via total variation minimization and related problems," Numer. Math., vol. 76, no. 2, pp. 167–188, 1997, ISSN: 0029-599X. DOI: 10.1007/s002110050258.

[40] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," J. Math. Imaging Vision, vol. 40, no. 1, pp. 120–145, 2011, ISSN: 0924-9907. DOI: 10.1007/s10851-010-0251-1.

[41] X. Chen and A. M. Powell, "Almost sure convergence of the kaczmarz algorithm with random measurements," Journal of Fourier Analysis and Applications, vol. 18, no. 6, pp. 1195–1214, 2012.

[42] L. Chizat and F. Bach, "Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss," in Conference on Learning Theory, 2020, pp. 1305–1338.

[43] H.-H. Chou, J. Maly, and H. Rauhut, "More is less: Inducing sparsity via overparameterization," Information and Inference: A Journal of the IMA, vol. 12, no. 3, iaad012, 2023.

[44] H.-H. Chou, H. Rauhut, and R. Ward, "Robust implicit regularization via weight normalization," arXiv preprint arXiv:2305.05448, 2023.

[45] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," Multiscale modeling & simulation, vol. 4, no. 4, pp. 1168–1200, 2005.

[46] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," J. Optim. Theory Appl., vol. 158, no. 2, pp. 460–479, 2013, ISSN: 0022-3239. DOI: 10.1007/s10957-012-0245-9.

[47] C. De Mol, E. De Vito, and L. Rosasco, "Elastic-net regularization in learning theory," Journal of Complexity, vol. 25, no. 2, pp. 201–230, 2009.

[48] D. L. Donoho, "Compressed sensing," IEEE Trans. Inform. Theory, vol. 52, no. 4, pp. 1289–1306, 2006, ISSN: 0018-9448. DOI: 10.1109/TIT.2006.871582.

[49] S. S. Du, W. Hu, and J. D. Lee, "Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced," Advances in Neural Information Processing Systems, vol. 31, 2018.

[50] J. Duchi and Y. Singer, "Efficient online and batch learning using forward backward splitting," J. Mach. Learn. Res., vol. 10, pp. 2899–2934, 2009, ISSN: 1532-4435.

[51] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," The Annals of statistics, vol. 32, no. 2, pp. 407–499, 2004.

[52] H. W. Engl, M. Hanke, and A. Neubauer, Regularization of inverse problems. Springer Science & Business Media, 1996, vol. 375.

[53] S. Foucart, H. Rauhut, S. Foucart, and H. Rauhut, An invitation to compressive sensing. Springer, 2013.

[54] G. Garrigos, L. Rosasco, and S. Villa, "Iterative regularization via dual diagonal descent," J. Math. Imaging Vision, vol. 60, no. 2, pp. 189–215, 2018, ISSN: 0924-9907. DOI: 10.1007/s10851-017-0754-0.

[55] G. Gidel, F. Bach, and S. Lacoste-Julien, "Implicit regularization of discrete gradient dynamics in linear neural networks," Advances in Neural Information Processing Systems, vol. 32, 2019.

[56] D. Gissin, S. Shalev-Shwartz, and A. Daniely, "The implicit bias of depth: How incremental learning drives generalization," in International Conference on Learning Representations, 2019.

[57] G. H. Golub, M. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," Technometrics, vol. 21, no. 2, pp. 215–223, 1979.

[58] S. Gunasekar, J. Lee, D. Soudry, and N. Srebro, "Characterizing implicit bias in terms of optimization geometry," in International Conference on Machine Learning, PMLR, 2018, pp. 1832–1841.

[59] S. Gunasekar, J. D. Lee, D. Soudry, and N. Srebro, "Implicit bias of gradient descent on linear convolutional networks," Advances in Neural Information Processing Systems, vol. 31, 2018.

[60] S. Gunasekar, B. E. Woodworth, S. Bhojanapalli, B. Neyshabur, and N. Srebro, "Implicit regularization in matrix factorization," Advances in Neural Information Processing Systems, vol. 30, pp. 6151–6159, 2017.

[61] E. B. Gutiérrez, C. Delplancke, and M. J. Ehrhardt, "Convergence properties of a randomized primal-dual algorithm with applications to parallel mri," in International Conference on Scale Space and Variational Methods in Computer Vision, Springer, New York, 2021, pp. 254–266.

[62] M. Hanke and W. Niethammer, "On the acceleration of kaczmarz's method for inconsistent linear systems," Linear Algebra and its Applications, vol. 130, pp. 83–98, 1990.

[63] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "The entire regularization path for the support vector machine," Journal of Machine Learning Research, vol. 5, no. Oct, pp. 1391–1415, 2004.

[64] T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning. springer series in statistics," New York, NY, USA, 2001.

[65] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," Technometrics, vol. 12, no. 1, pp. 55–67, 1970.

[66] M. Jaggi, "Revisiting frank-wolfe: Projection-free sparse convex optimization," in International conference on machine learning, PMLR, 2013, pp. 427–435.

[67] Z. Ji and M. Telgarsky, "Gradient descent aligns the layers of deep linear networks," in International Conference on Learning Representations, 2018.

[68] Z. Ji and M. Telgarsky, "The implicit bias of gradient descent on nonseparable data," in Conference on Learning Theory, PMLR, 2019, pp. 1772–1798.

[69] B. Jin, D. A. Lorenz, and S. Schiffler, "Elastic-net regularization: Error estimates and active set methods," Inverse Problems, vol. 25, no. 11, p. 115 022, 2009.

[70] S. Kaczmarz, "English translation: S. kaczmarz, approximate solution of systems of linear equations," Int. J. Contr., vol. 57, pp. 355–357, 1937.

[71] B. Kaltenbacher, A. Neubauer, and O. Scherzer, Iterative regularization methods for nonlinear ill-posed problems (Radon Series on Computational and Applied Mathematics). Walter de Gruyter GmbH & Co. KG, Berlin, 2008, vol. 6, pp. viii+194, ISBN: 978-3-11-020420-9. DOI: 10.1515/9783110208276.

[72] L. Landweber, "An iteration formula for Fredholm integral equations of the first kind," Amer. J. Math., vol. 73, pp. 615–624, 1951, ISSN: 0002-9327. DOI: 10.2307/2372313.

[73] D. Leventhal and A. S. Lewis, "Randomized methods for linear constraints: Convergence rates and conditioning," Mathematics of Operations Research, vol. 35, no. 3, pp. 641–654, 2010.

[74] H. Li, N. Chen, and L. Li, "Error analysis for matrix elastic-net regularization algorithms," IEEE transactions on neural networks and learning systems, vol. 23, no. 5, pp. 737–748, 2012.

[75] Y. Li, T. Ma, and H. Zhang, "Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations," in Conference On Learning Theory, PMLR, 2018, pp. 2–47.

[76] Z. Li, T. Wang, and S. Arora, "What happens after sgd reaches zero loss?–a mathematical framework," in International Conference on Learning Representations, 2021.

[77] Z. Li, T. Wang, J. D. Lee, and S. Arora, "Implicit bias of gradient descent on reparametrized models: On equivalence to mirror descent," Advances in Neural Information Processing Systems, vol. 35, pp. 34 626–34 640, 2022.

[78] D. A. Lorenz, "Convergence rates and source conditions for tikhonov regularization with sparsity constraints," Journal of Inverse and Ill-Posed Problems, vol. 16, no. 5, pp. 463–478, 2008.

[79] K. Lyu and J. Li, "Gradient descent maximizes the margin of homogeneous neural networks," in International Conference on Learning Representations, 2019.

[80] J. Maly, "Robust sensing of low-rank matrices with non-orthogonal sparse decomposition," Applied and Computational Harmonic Analysis, 2023.

[81] B. Martinet, "Regularisation d'inequations variationelles par approximations successives," Revue Francaise d'informatique et de Recherche operationelle, vol. 4, pp. 154–159, 1970.

[82] S. Matet, L. Rosasco, S. Villa, and B. L. Vu, "Don't relax: Early stopping for convex regularization," arXiv preprint arXiv:1707.05422, 2017.

[83] C. Molinari, J. Liang, and J. Fadili, "Convergence rates of forward–douglas–rachford splitting method," Journal of Optimization Theory and Applications, vol. 182, pp. 606–639, 2019.

[84] C. Molinari, M. Massias, L. Rosasco, and S. Villa, "Iterative regularization for convex regularizers," in International Conference on Artificial Intelligence and Statistics, PMLR, 2021, pp. 1684–1692.

[85] C. Molinari, M. Massias, L. Rosasco, and S. Villa, "Iterative regularization for low complexity regularizers," Arxiv 2202.00420, pp. 1684–1692, 2022.

[86] C. Molinari and J. Peypouquet, "Lagrangian penalization scheme with parallel forward–backward splitting," Journal of Optimization Theory and Applications, vol. 177, pp. 413–447, 2018.

[87] C. Molinari, J. Peypouquet, and F. Roldan, "Alternating forward–backward splitting for linearly constrained optimization problems," Optimization Letters, vol. 14, pp. 1071–1088, 2020.

[88] E. Moroshko, B. E. Woodworth, S. Gunasekar, J. D. Lee, N. Srebro, and D. Soudry, "Implicit bias in deep linear classification: Initialization scale vs training accuracy," Advances in neural information processing systems, vol. 33, pp. 22 182–22 193, 2020.

[89] D. Morwani and H. G. Ramaswamy, "Inductive bias of gradient descent for weight normalized smooth homogeneous neural nets," in International Conference on Algorithmic Learning Theory, PMLR, 2022, pp. 827–880.

[90] E. Moulines and F. Bach, "Non-asymptotic analysis of stochastic approximation algorithms for machine learning," Advances in neural information processing systems, vol. 24, pp. 451–459, 2011.

[91] J. Munkres, "Topology james munkres second edition,"

[92] M. S. Nacson, J. Lee, S. Gunasekar, P. H. P. Savarese, N. Srebro, and D. Soudry, "Convergence of gradient descent on separable data," in The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 3420–3428.

[93] M. S. Nacson, K. Ravichandran, N. Srebro, and D. Soudry, "Implicit bias of the step size in linear diagonal neural networks," in International Conference on Machine Learning, PMLR, 2022, pp. 16 270–16 295.

[94] M. S. Nacson, N. Srebro, and D. Soudry, "Stochastic gradient descent on separable data: Exact convergence with a fixed learning rate," in The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 3051–3059.

[95] I. Necoara, "Faster randomized block kaczmarz algorithms," SIAM Journal on Matrix Analysis and Applications, vol. 40, no. 4, pp. 1425–1452, 2019.

[96] D. Needell, "Randomized kaczmarz solver for noisy linear systems," BIT Numerical Mathematics, vol. 50, pp. 395–403, 2010.

[97] D. Needell, R. Ward, and N. Srebro, "Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm," Advances in neural information processing systems, vol. 27, 2014.

[98] D. Needell, R. Zhao, and A. Zouzias, "Randomized block kaczmarz method with projection for solving least squares," Linear Algebra and its Applications, vol. 484, pp. 322–343, 2015.

[99] A. S. Nemirovski and D. B. Yudin, Problem complexity and method efficiency in optimization (A Wiley-Interscience Publication). New York: John Wiley & Sons Inc., 1983, pp. xv+388.

[100] A. S. Nemirovsky and D. B. a. Yudin, Problem complexity and method efficiency in optimization (Wiley-Interscience Series in Discrete Mathematics). John Wiley & Sons, Inc., New York, 1983, pp. xv+388, Translated from the Russian and with a preface by E. R. Dawson, ISBN: 0-471-10345-4.

[101] A. Neubauer, "On Nesterov acceleration for Landweber iteration of linear ill-posed problems," J. Inverse Ill-Posed Probl., vol. 25, no. 3, pp. 381–390, 2017, ISSN: 0928-0219. DOI: 10.1515/jiip-2016-0060.

[102] B. Neyshabur, R. Tomioka, and N. Srebro, "In search of the real inductive bias: On the role of implicit regularization in deep learning," arXiv preprint arXiv:1412.6614, 2014.

[103] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," Multiscale Model. Simul., vol. 4, no. 2, pp. 460–489, 2005, ISSN: 1540-3459. DOI: 10.1137/040605412.

[104] S. Osher, Y. Mao, B. Dong, and W. Yin, "Fast linearized Bregman iteration for compressive sensing and sparse denoising," Commun. Math. Sci., vol. 8, no. 1, pp. 93–111, 2010, ISSN: 1539-6746.

[105] S. Osher and L. I. Rudin, "Feature-oriented image enhancement using shock filters," SIAM Journal on numerical analysis, vol. 27, no. 4, pp. 919–940, 1990.

[106] S. Pesme, L. Pillaud-Vivien, and N. Flammarion, "Implicit bias of sgd for diagonal linear networks: A provable benefit of stochasticity," Advances in Neural Information Processing Systems, vol. 34, pp. 29 218–29 230, 2021.

[107] J. Peypouquet, Convex optimization in normed spaces (SpringerBriefs in Optimization). Springer, Cham, 2015, pp. xiv+124, Theory, methods and examples, With a foreword by Hedy Attouch, ISBN: 978-3-319-13710-0. DOI: 10.1007/978-3-319-13710-0.

[108] G. Peyré, "The numerical tours of signal processing-advanced computational signal and image processing," IEEE Computing in Science and Engineering, vol. 13, no. 4, pp. 94–97, 2011.

[109] T. Pock and A. Chambolle, "Diagonal preconditioning for first order primal-dual algorithms in convex optimization," in 2011 International Conference on Computer Vision, 2011, pp. 1762–1769.

[110] H. Raguet, J. Fadili, and G. Peyré, "A generalized forward-backward splitting," SIAM J. Imaging Sci., vol. 6, no. 3, pp. 1199–1226, 2013.

[111] J. Rasch and A. Chambolle, "Inexact first-order primal-dual algorithms," Comput. Optim. Appl., vol. 76, no. 2, pp. 381–430, 2020, ISSN: 0926-6003. DOI: 10.1007/s10589-020-00186-y.

[112] G. Raskutti, M. J. Wainwright, and B. Yu, "Early stopping and non-parametric regression: An optimal data-dependent stopping rule," The Journal of Machine Learning Research, vol. 15, no. 1, pp. 335–366, 2014.

[113] R. T. Rockafellar, Convex analysis (Princeton Landmarks in Mathematics). Princeton University Press, Princeton, NJ, 1997, pp. xviii+451, Reprint of the 1970 original, Princeton Paperbacks, ISBN: 0-691-01586-4.

[114] L. Rosasco and S. Villa, "Learning with incremental iterative regularization," NeurIPS, pp. 1630–1638, 2015.

[115] M. Rudelson and R. Vershynin, "Geometric approach to error-correcting codes and reconstruction of signals," International mathematics research notices, vol. 2005, no. 64, pp. 4019–4041, 2005, ISSN: 1073-7928. DOI: 10.1155/IMRN.2005.4019. [Online]. Available: https://doi.org/10.1155/IMRN.2005.4019.

[116] L. I. Rudin and S. Osher, "Total variation based image restoration with free local constraints," in Proceedings of 1st International Conference on Image Processing, IEEE, vol. 1, 1994, pp. 31–35.

[117] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," in 1-4, vol. 60, Experimental mathematics: computational issues in nonlinear science (Los Alamos, NM, 1991), 1992, pp. 259–268. DOI: 10.1016/0167-2789(92)90242-F.

[118] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," Advances in neural information processing systems, vol. 29, 2016.

[119] O. Scherzer, "A modified Landweber iteration for solving parameter estimation problems," Appl. Math. Optim., vol. 38, no. 1, pp. 45–68, 1998, ISSN: 0095-4616. DOI: 10.1007/s002459900081.

[120] F. Schöpfer and D. A. Lorenz, "Linear convergence of the randomized sparse kaczmarz method," Mathematical Programming, vol. 173, pp. 509–536, 2019.

[121] S. Shalev-Shwartz and S. Ben-David, Understanding machine learning: From theory to algorithms. Cambridge university press, 2014.

[122] A. Silveti-Falls, C. Molinari, and J. Fadili, "A stochastic bregman primal-dual splitting algorithm for composite optimization," arXiv preprint arXiv:2112.11928, 2021.

[123] A. Silveti-Falls, C. Molinari, and J. Fadili, "Generalized conditional gradient with augmented lagrangian for composite minimization," SIAM Journal on Optimization, vol. 30, no. 4, pp. 2687–2725, 2020.

[124] A. Silveti-Falls, C. Molinari, and J. Fadili, "Inexact and stochastic generalized conditional gradient with augmented lagrangian and proximal step," Journal of Nonsmooth Analysis and Optimization, vol. 2, no. Original research articles, 2021.

[125] D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro, "The implicit bias of gradient descent on separable data," The Journal of Machine Learning Research, vol. 19, no. 1, pp. 2822–2878, 2018.

[126] I. Steinwart and A. Christmann, Support vector machines (Information Science and Statistics). Springer, New York, 2008, pp. xvi+601, ISBN: 978-0-387-77241-7.

[127] T. Strohmer and R. Vershynin, "A randomized kaczmarz algorithm with exponential convergence," Journal of Fourier Analysis and Applications, vol. 15, no. 2, pp. 262–278, 2009.

[128] A. N. Tihonov, "Solution of incorrectly formulated problems and the regularization method," Soviet Math., vol. 4, pp. 1035–1038, 1963.

[129] Y. Tsaig and D. L. Donoho, "Extensions of compressed sensing," Signal processing, vol. 86, no. 3, pp. 549–571, 2006.

[130] G. Vardi and O. Shamir, "Implicit regularization in relu networks with the square loss," in Conference on Learning Theory, PMLR, 2021, pp. 4224–4258.

[131] T. Vaskevicius, V. Kanade, and P. Rebeschini, "Implicit regularization for optimal sparse recovery," Advances in Neural Information Processing Systems, vol. 32, 2019.

[132] C. Vega, C. Molinari, L. Rosasco, and S. Villa, "Fast iterative regularization by reusing data," Journal of Inverse and Ill-posed Problems, no. 0, 2023.

[133] B. C. Vũ, "A splitting algorithm for dual monotone inclusions involving cocoercive operators," Adv. Comput. Math., vol. 38, no. 3, pp. 667–681, 2013, ISSN: 1019-7168. DOI: 10.1007/s10444-011-9254-8.

[134] Y. Wang, M. Chen, T. Zhao, and M. Tao, "Large learning rate tames homogeneity: Convergence and balancing effect," in The International Conference on Learning Representations, 2022.

[135] B. Woodworth et al., "Kernel and rich regimes in overparametrized models," in Conference on Learning Theory, PMLR, 2020, pp. 3635–3673.

[136] F. Wu and P. Rebeschini, "A continuous-time mirror descent approach to sparse phase retrieval," Advances in Neural Information Processing Systems, vol. 33, pp. 20 192–20 203, 2020.

[137] F. Wu and P. Rebeschini, "Implicit regularization in matrix sensing via mirror descent," Advances in Neural Information Processing Systems, vol. 34, pp. 20 558–20 570, 2021.

[138] L. Wu, C. Ma, et al., "How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective," Advances in Neural Information Processing Systems, vol. 31, 2018.

[139] X. Wu et al., "Implicit regularization and convergence for weight normalization," Advances in Neural Information Processing Systems, vol. 33, pp. 2835–2847, 2020.

[140] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," J. Mach. Learn. Res., vol. 11, pp. 2543–2596, 2010, ISSN: 1532-4435.

[141] Y. Yao, L. Rosasco, and A. Caponnetto, "On early stopping in gradient descent learning," Constr. Approx., vol. 26, no. 2, pp. 289–315, 2007, ISSN: 0176-4276. DOI: 10.1007/s00365-006-0663-2.

[142] W. Yin, "Analysis and generalizations of the linearized bregman method," SIAM Journal on Imaging Sciences, vol. 3, no. 4, pp. 856–877, 2010.

[143] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for $\ell_1$-minimization with applications to compressed sensing," SIAM Journal on Imaging sciences, vol. 1, no. 1, pp. 143–168, 2008.

[144] C. You, Z. Zhu, Q. Qu, and Y. Ma, "Robust recovery via implicit bias of discrepant learning rates for double over-parameterization," Advances in Neural Information Processing Systems, vol. 33, pp. 17 733–17 744, 2020.

[145] C. Yun, S. Krishnan, and H. Mobahi, "A unifying view on implicit bias in training linear neural networks," in International Conference on Learning Representations, 2020.

[146] T. Zhang and B. Yu, "Boosting with early stopping: Convergence and consistency," Ann. Statist., vol. 33, no. 4, pp. 1538–1579, 2005, ISSN: 0090-5364. DOI: 10.1214/009053605000000255.

[147] P. Zhao, Y. Yang, and Q.-C. He, "Implicit regularization via hadamard product over-parametrization in high-dimensional linear regression," arXiv preprint arXiv:1903.09367, 2019.

[148] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," Journal of the royal statistical society: series B (statistical methodology), vol. 67, no. 2, pp. 301–320, 2005.

[149] H. Zou and H. H. Zhang, "On the adaptive elastic-net with a diverging number of parameters," Annals of statistics, vol. 37, no. 4, p. 1733, 2009.

[150] A. Zouzias and N. M. Freris, "Randomized extended kaczmarz for solving least squares," SIAM Journal on Matrix Analysis and Applications, vol. 34, no. 2, pp. 773–793, 2013.