UNIVERSITÀ
DEGLI STUDI
FIRENZE

PHD PROGRAM IN SMART COMPUTING
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE (DINFO)

Theory and Algorithms for Sparsity Constrained Optimization Problems

Matteo Lapucci

Dissertation presented in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Smart Computing

PhD Program in Smart Computing
University of Florence, University of Pisa, University of Siena

Theory and Algorithms for Sparsity Constrained Optimization Problems

Matteo Lapucci

Advisor:

Prof. Marco Sciandrone

Head of the PhD Program:

Prof. Paolo Frasconi

Evaluation Committee:
Prof. Christian Kanzow, *Universität Würzburg*
Prof. Veronica Piccialli, *Sapienza Università di Roma*

XXXIV ciclo — October 2022

Alla mia famiglia

Acknowledgments

I tried really hard to keep this Section short this time, I swear. But in the end, I miserably failed: there are too many people I have to thank, from the bottom of my heart.

First and foremost, I have had an extraordinary mentor for the last four years; a special teacher that made me get passionate about his subject, which is now also mine; a great researcher that led me to embrace the challenges of the academic world; most of all, an amazing person, a selfless gentleman that I dare to consider a friend. I want Prof. Marco Sciandrone to know that I am aware of how lucky I have been for all of that.

Then, my gratitude also goes to Prof. Fabio Schoen for having built and preserved for more than 25 years a stimulating environment where people can carry out high quality and interesting work in an incredibly positive atmosphere. Also, the enthusiasm and the trust he shows towards his research group do not go unnoticed.

I have to thank two international caliber researchers, namely Prof. Chih-Jen Lin and Prof. Francesco Rinaldi, that I have had the honor to work with and the privilege to learn from, during these three years. I also have to thank Prof. Stefano Lucidi and Dr. Giampaolo Liuzzi for spending their time as my Supervisory Committee.

Now, here comes the dream team I have been part of. I hope you guys know how much I am indebted to all of you. Tommaso L., Luca, Guido, Alessandro, Giulio and Leonardo G.: each one of you left me an invaluable legacy, both on the professional and, most importantly, on the human level. Pierluigi, Tommaso A. and Simone, I hope I repaid, at least a little bit, the undeserved trust you have often shown to have in me. Leonardo D., you're the best person I have had the opportunity to meet by doing the doctorate; even though you "are not educated", you have given me much more than I did to you. Enrico, my friend, I really don't care about your complaints: you have been the best pal for me both inside and outside the lab for these three years, so your PhD won't be "a complete waste of time" in any case. Last, but definitely not least, in order to complete the institutional credits I have to thank my "slimy" partner-in-crime, Alessio; we have faced a ton of challenges for eight long years at university, I certainly wouldn't have won most of them without you by my side.

Doing a PhD is not only a scientific and professional growth path, but it also tests you on the human level.

This journey has been heavily eased by Matteo, Alessandra, Pietro, Vishal, Luca and again Enrico and Alessio, who have been a formidable group of friends to share both the happy and the difficult times with. I surely believe that the bond that unites us will last long and strong.

I cannot forget of my volleyball court brothers, Niccolò, Tommaso, Tommaso,

Corrado, Andrea, Leonardo, Bernardo, Dario, Lorenzo, Carlo, Devid, Filippo and the rest of the gang at Sales Volley, that welcomed me from day one, making me feel at home. I owe them many successes, not just in sports competition but also in my life. I will always be grateful.

I was spending my time reasoning about optimization problems when, totally unexpected, a blond avalanche fell on me, changing my life for the better and somehow giving a sense to years of sacrifice and doubts. Giulia, there is no question, you are the highlight of my PhD years. I hope you'll be patient enough to keep bringing me joy for the times to come.

Finally, when I graduated, I thanked mom, dad, grandmas, aunt and uncles for everything they have done and represent for me. Their merits still stand high, but this time I feel I achieved the goal with my own strengths; in fact, being quite far away from them has been one of the hardest hurdles to overcome and I know that this feeling is mutual. Thus, instead of giving thanks, I feel free in this occasion to dedicate this manuscript to my wonderful family. I don't care if they don't understand my work, because of the language or the mathematics. Every single word is for them.

Oh, yeah, and many greetings of course to my two favorite youngsters, that have grown up already. I can see you laughing right now, not worried at all, as you watch me trying again to set the bar higher. I can't blame you for being rightfully conscious that you are already much better persons than I am.

Florence, October 31, 2021,

Matteo

Abstract

This dissertation is concerned with mathematical optimization problems where a sparsity constraint appears. The sparsity of the solution is a valuable requirement in many applications of operations research. Several classes of very different approaches have been proposed in the literature for this sort of problems; when the objective function is nonconvex, in presence of difficult additional constraints or in the high-dimensional case, the problem shall be addressed as a continuous optimization task, even though it naturally has an intrinsic combinatorial nature.

Within this setting, we first review the existing knowledge and the theoretical tools concerning the considered problem; we try to provide a unified view of parallel streams of research and we propose a new general stationarity condition, based on the concept of neighborhood, which somehow allows to take into account both the continuous and the combinatorial aspects of the problem.

Then, after a brief overview of the main algorithmic approaches in the related literature, we propose suitable variants of some of these schemes that can be effectively employed in complex settings, such as the nonconvex one, the derivative-free one or the multi-objective one. For each of the proposed algorithms we provide a detailed convergence analysis showing that these methods enjoy important theoretical guarantees, in line with the state-of-the-art algorithms.

Afterwards, exploiting the newly introduced concept of stationarity, we propose a completely novel algorithmic scheme that, combining continuous local searches and discrete moves, can be proved to guarantee stronger theoretical properties than most approaches from the literature and to exhibit strong exploration capabilities in a global optimization perspective.

All the proposed algorithms have finally been experimentally tested on a benchmark of relevant problems from machine learning and decision science applications. The computational results show the actual quality of the proposed methods when practically employed.

Contents

Chapter 1

Introduction

Mathematical optimization has represented, for more than half a century already, an indispensable tool for the amazing progress many disciplines have experienced. The most sensational example of this phenomenon is arguably given by the boom of data science and artificial intelligence, which would not have been possible without at least thirty years of optimization groundwork. Anyhow, many other fields could be cited where mathematical programming represented a decisive cog for the development of new technologies.

On the one hand, it is thus clear to the operations research community that the study of new models, algorithmic approaches and theoretical tools, even those that shall appear quite abstract and useless, may have an unpredictably great impact in the most different disciplines at a later time. On the other hand, once employing optimization techniques has become common practice for a particular application, researchers from that area constantly pose new demands and consequently new challenges to optimizers.

Nowadays, one of these challenges concerns the sparsity requirements that can be found in many applications of optimization models. Indeed, solutions to optimization problems with a low cardinality of the decision variables vector are often required, for example, in finance and decision science (Bertsimas and Cory-Wright, 2018; Cesarone et al., 2013; Di Lorenzo et al., 2012; Gao and Li, 2013; Teng et al., 2017), in signal processing (Blumensath and Davies, 2009; Candès and Wakin, 2008; Donoho, 2006; Foucart and Rauhut, 2013), in statistics (Bertsimas et al., 2016; Bertsimas and King, 2017; Civitelli et al., 2021; Di Gangi et al., 2019; Friedman et al., 2008; Guillot et al., 2012; Miller, 2002) and in machine learning (Bertsimas et al., 2017; Carlini and Wagner, 2017; Carreira-Perpinán and Idelbayev, 2018; d'Aspremont et al., 2008; Mairal et al., 2014; Zou et al., 2006). For a thorough review of applications of sparse optimization we refer the reader to the survey by Tillmann et al. (2021) and references therein.

In data science in particular, a sparse model shows important features such as high generalization capabilities, enhanced interpretability, efficiency and lower memory requirements (Bach et al., 2012; Weston et al., 2003).

The need for sparse solutions has been addressed in the most diverse ways by the optimization community. The cardinality of the variables vector can be modeled by using the ℓ_0 pseudo-norm, whose formal definition is recalled hereafter.

Definition 1.1 (ℓ_0 pseudo norm)**.** Let $x \in \mathbb{R}^n$. The *zero pseudo-norm* of x, also referred to as ℓ_0 pseudo-norm and denoted by the notation $\|x\|_0$, is defined as:

$$\|x\|_0 = |\{i \mid x_i \neq 0,\ i = 1, \dots, n\}| .$$

Based on the way the ℓ_0 term is inserted into the optimization problem to induce sparsity, we can distinguish three general classes of sparse optimization problems:

(i) *Cardinality Minimization Problems*, where the ℓ_0 pseudo-norm of the variables vector is minimized subject to some constraints;

(ii) *Cardinality Constrained Problems*, where an objective function is minimized subject to some constraints, among which there is an upper bound on the ℓ_0 pseudo-norm;

(iii) *Cardinality Regularized Problems*, where the objective function is a weighted sum of the ℓ_0 pseudo-norm and some other general function of the variables vector.

Now, optimization problems with ℓ_0 elements are well-known to be $\mathcal{NP}$-hard (Bienstock, 1996; Natarajan, 1995; Nguyen et al., 2019). This should not be surprising, as the sparsity requirement hides the combinatorial task of selecting the best subset of variables. The diversity of approaches that have been explored in recent years comes from the complexity of the task itself: indeed, various paths have been followed trying to tackle the problem hardness.

A wide stream of research has focused on possible ways to approximate the nonconvex discontinuous ℓ_0 element. Families of approaches are based on ℓ_1 (Bach et al., 2012; Beck and Teboulle, 2009; Chen et al., 2001; Donoho and Tsaig, 2008; Foucart and Rauhut, 2013; Malioutov et al., 2005; Tibshirani, 1996; Yin et al., 2008) and ℓ_p (Chartrand, 2007; Chen et al., 2010; Ge et al., 2011; Mourad and Reilly, 2010) surrogates, concave programming (Di Lorenzo et al., 2012; Mangasarian, 1999; Rinaldi et al., 2010), DC techniques (Gotoh et al., 2018; Le Thi et al., 2015). In addition, heuristic approaches based on evolutionary algorithms (Anagnostopoulos and Mamanis, 2010), particle swarm methods (Boudt and Wan, 2020; Deng et al., 2012), genetic algorithms, tabu search and simulated annealing (Chang et al., 2000) have been considered.

The aforementioned classes of methods suffer from drawbacks; in particular, the solutions retrieved either do not satisfy theoretical optimality conditions in the general case or are bad from a global optimization perspective.

The impressive recent advances of mixed-integer programming (MIP) algorithms allowed to define schemes designed to obtain exact solutions with a certificate of global optimality. Sparse optimization problems can be cast into MIP problems by introducing binary variables and big-M constraints (Belotti et al., 2016; Civitelli et al., 2021; Di Gangi et al., 2019; Ben Mhenni et al., 2021; Miyashiro and Takano, 2015) or complementarity-type constraints (Burdakov et al., 2016; Feng et al., 2013; Yu et al., 2019) and then using efficient branch-and-bound or branch-and-cut procedures; an alternative is given by the approach from Bertsimas et al. (2019) where a suitable change of variables makes the problem efficiently solvable by an outer approximation scheme.

In highly nonlinear or in nonconvex settings, however, exact approaches quickly become computationally unviable. For this reason, approaches to directly tackle problems with ℓ_0 terms based on classical tools from local continuous optimization theory have been proposed.

In this thesis, we specifically consider nonconvex nonlinear optimization problems with sparsity constraints, i.e., problems of the form

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & \|x\|_0 \leq s, \\ & x \in X, \end{aligned} \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function, $X \subseteq \mathbb{R}^n$ is a closed and convex set, and $s < n$ is a properly chosen integer value. When possible, the convex set X will also be analytically expressed as $X = \{x \in \mathbb{R}^n \mid g(x) \leq 0, h(x) = 0\}$, where h_i, $i = 1, \ldots, p$ are affine functions and g_i, $i = 1, \ldots, m$ are convex functions. We further use $\mathcal{X}$ to indicate the overall feasible set $X \cap \{x \in \mathbb{R}^n \mid \|x\|_0 \leq s\}$.

The interest of this thesis lies in the analysis of problem (1.1) as a continuous optimization problem, both on the theoretical and algorithmic sides. More in detail, the rest of the manuscript concerns the following novel contributions:

- In Chapter 2, we review the optimality conditions theory for problem (1.1), providing a unified view of the literature; moreover, a new theoretical point of view is proposed, based on a tailored concept of stationarity, that allows to recast most of the known theory as special cases of a more general and powerful framework.

- In Chapter 3, we briefly review and compare the main computational approaches from the literature to tackle problem (1.1). For each considered algorithm, we report the main theoretical convergence properties and highlight practical strengths and weaknesses.

- In Chapter 4, we focus on the approach of the Penalty Decomposition (PD) methods and propose a convergent algorithm of this family performing inexact minimizations by an Armijo-type line search along gradient-related directions. The algorithm is shown to enjoy the same convergence results as the base scheme, but can practically be applied without convexity assumptions on the objective function.

- In Chapter 5, we provide the definition of a derivative-free PD method for sparse black-box optimization. The algorithm is again shown to possess the convergence properties as its gradient-based counterparts.

- In Chapter 6, we propose an algorithmic framework, based on the concept of neighborhood, to tackle sparsity constrained problems and prove its convergence to points satisfying the previously introduced new concept of stationarity. We further show that, by suitably choosing the neighborhood, other well-known optimality conditions from the literature can be guaranteed for the limit points of the sequence produced by the algorithm.

- In Chapter 7, we focus on sparsity-constrained problems in the multi-objective setting. We carry out an analysis of optimality conditions for this family of problems. Then, we define a PD-type method to solve these problems and provide a thorough theoretical analysis showing that the algorithm possesses convergence properties to feasible points satisfying first-order optimality conditions.

- In Chapter 8, we report the results of computational experiments, carried out on a benchmark of real world relevant problems and aimed at assessing the empirical performance of all the algorithmic approaches proposed in this thesis.

- In Chapter 9, we finally draw some concluding remarks and suggest possible themes for future research.

Chapter 2

Optimality Conditions for Sparsity-Constrained Optimization Problems

Even though problem (1.1) is an optimization problem with continuous decision variables, it has an intrinsic combinatorial nature and in applications the interest often lies in finding a good, possibly globally optimal configuration of active variables.

Being (1.1) a continuous problem, $x^\star \in \mathcal{X}$ is a local minimizer if there exists an open ball $\mathcal{B}(x^\star, \epsilon)$ such that $f(x^\star) = \min\{f(x) \mid x \in \mathcal{X} \cap \mathcal{B}(x^\star, \epsilon)\}$. In some works from the literature (e.g., Burdakov et al. 2016; Lu and Zhang 2013) necessary conditions of local optimality have been proposed.

However, for this particular problem, every local minimizer for a fixed active set of s variables is also a local minimizer of the overall problem. Hence the number of local minimizers grows as fast as $\binom{n}{s}$ and is thus of low practical usefulness.

In other works (Beck and Eldar, 2013; Beck and Hallak, 2016), the authors propose necessary conditions for global optimality that go beyond the concept of local minimizer, thus allowing to consider possible changes to the structure of the solution and reducing the pool of optimal candidates. However, these conditions are either tailored to the "unconstrained case", or limited to moderate changes in the active set of nonzero variables, or involve hard operations, such as exact minimizations or projections onto nonconvex sets.

In this Chapter, we analyze necessary optimality conditions for sparsity constrained optimization problems. We begin by considering the simpler case where $X = \mathbb{R}^n$, which allows us to gradually make the reader familiar with basic concepts and issues of sparse optimization. We then turn to the more general case, reviewing in detail the necessary optimality conditions proposed in the literature for problem (1.1). As an important contribution of this work, we thoroughly clarify how these quite diverse conditions relate to each other. Next, we introduce a general and affordable necessary optimality condition that also takes into account the combinatorial nature of the problem. We finally show that most of the existing conditions can be recast as special cases of the newly introduced one. Before getting into the details, we begin by recalling terminology and basic definitions that will recurrently be used throughout the thesis.

2.1 Preliminaries

Basic Definitions and Notation

Concerning problem (1.1), one of the most basic concepts is given by the support set of a solution and its complement.

Definition 2.1 (Support set)**.** Let $\bar{x} \in \mathcal{X}$ be a feasible solution of problem (1.1). The *support set* of $\bar{x}$ is the index set $I_1(\bar{x}) \subset \{1, \dots, n\}$ of the nonzero variables of $\bar{x}$, i.e.,

$$I_1(\bar{x}) = \{i \mid \bar{x}_i \neq 0\}.$$

Similarly, we can define the complement $I_0(\bar{x})$ of $I_1(\bar{x})$, constituted by the indices of zero variables:

$$I_0(\bar{x}) = \{i \mid \bar{x}_i = 0\} = \{1, \dots, n\} \setminus I_1(\bar{x}).$$

A point $\bar{x} \in \mathcal{X}$ is a *full-support solution* for problem (1.1) if

$$\|\bar{x}\|_0 = |I_1(\bar{x})| = s,$$

whereas it has an *incomplete support* if $\|\bar{x}\|_0 < s$.

Example 2.1. Consider problem (1.1) with $n = 4$, $s = 2$ and $X = \mathbb{R}^4$. Let $x = (2, 0, 0, 2)$ and $y = (0, 3, 0, 0)$. We have

$$\begin{aligned} I_1(x) &= \{1, 4\} \quad & I_0(x) &= \{2, 3\}, \\ I_1(y) &= \{2\} \quad & I_0(y) &= \{1, 3, 4\}. \end{aligned}$$

The vector x has full support, whereas y has an incomplete support set.

In order to enrich the characterization of solutions of the problem, a further concept can be defined (Beck and Hallak, 2016).

Definition 2.2 (Super support set)**.** Let $\bar{x} \in \mathcal{X}$ be a feasible solution of problem (1.1). A set $J \subseteq \{1, \dots, n\}$ is referred to as a super support set for $\bar{x}$ if it is such that

- $I_1(\bar{x}) \subseteq J$,

- $|J| = s$.

We denote the set of all super support sets at $\bar{x}$ by $\mathcal{J}(\bar{x})$.

A super support set substantially identifies a subset of components of $\bar{x}$ that could be moved jointly without breaking the cardinality constraint. Clearly, if $\bar{x}$ has full support, then the only super support set for $\bar{x}$ is $I_1(\bar{x})$ itself.

Example 2.2. Consider the setting of Example 2.1. Then we have:

$$\mathcal{J}(x) = \{I_1(x)\},$$
$$\mathcal{J}(y) = \{J_A, J_B, J_C\}, \quad J_A = \{1,2\}, \quad J_B = \{2,3\}, \quad J_C = \{2,4\}.$$

Additional Notation

Throughout the entire work, we will denote by x_I the subvector of $x \in \mathbb{R}^n$ identified by the components contained in an index set I and by $\mathcal{X}_I$ the convex feasible set associated with the active set of variables identified by I:

$$\mathcal{X}_I = \{x \in X \mid x_i = 0 \text{ for all } i \notin I\}.$$

Projection onto Feasible Sets

In this work, we indicate by Π_X the classical orthogonal projection operator over the closed convex set $X \subset \mathbb{R}^n$; given $x \in \mathbb{R}^n$, we have

$$\Pi_X(x) = \operatorname*{arg\,min}_{z \in X} \|z - x\|_2^2.$$

In addition, we also define the sparse projection operator (Beck and Hallak, 2016):

Definition 2.3 (Sparse projection operator)**.** Consider the feasible set $\mathcal{X}$ of problem (1.1) and let $\bar{x} \in \mathbb{R}^n$. The *sparse projection operator* $\Pi_{\mathcal{X}} : \mathbb{R}^n \to \mathcal{X}$ maps $\bar{x}$ to a feasible solution of (1.1) as follows:

$$\Pi_{\mathcal{X}}(\bar{x}) \in \operatorname{arg\,min}\{\|z - \bar{x}\|^2 \mid z \in \mathcal{X}\}.$$

Since $\mathcal{X}$ is closed, the set $\Pi_{\mathcal{X}}(x)$ is always nonempty; however since $\mathcal{X}$ is nonconvex, it is not necessarily a singleton. In general, finding the sparse projection set is a difficult task.

However, in the case where $X = \mathbb{R}^n$, the sparse projection can be computed in closed form. To formally characterize the solution, let us define the index set $\mathcal{G}(x)$ of the largest nonzero variables (in absolute value) at a generic point $\hat{x} \in \mathbb{R}^n$:

$$\begin{aligned} \mathcal{G}(\hat{x}) \in \operatorname*{arg\,max}_{S \subseteq \{1,\dots,n\}} & |S| \\ \text{s.t.} \quad & |S| \le s, \quad S \subseteq I_1(\hat{x}), \\ & |\hat{x}_i| \ge |\hat{x}_j| \quad \forall\, i \in S, \forall\, j \notin S. \end{aligned} \tag{2.1}$$

In general, the index set $\mathcal{G}(\hat{x})$ is not uniquely defined. Also, note that $\mathcal{G}(\hat{x}) = I_1(\hat{x})$ if $\|x\|_0 \leq s$. Then, the sparse projection $x^\star = \Pi_{\mathcal{X}}$ of $\hat{x}$ is given by

$$x_i^\star = \hat{x}_i \text{ for } i \in \mathcal{G}(\hat{x}), \qquad x_i^\star = 0 \text{ for } i \notin \mathcal{G}(\hat{x}), \tag{2.2}$$

i.e., the sparse projection can be obtained by zeroing all the variables except for the s largest ones in absolute value.

Example 2.3. Consider the setting of Example 2.1 and let $\hat{x} = (-2, 1, 3, 0)$ and $\hat{z} = (-1, 0.5, 1, 1.5)$. Then we have

$$\Pi_{\mathcal{X}}(\hat{x}) = (-2, 0, 3, 0),$$

while for $\hat{z}$ we can choose either

$$\Pi_{\mathcal{X}}(\hat{z}) = (-1, 0, 0, 1.5) \qquad \text{or} \qquad \Pi_{\mathcal{X}}(\hat{z}) = (0, 0, 1, 1.5).$$

A Complementarity-Constrained Mixed-Integer Reformulation

An alternative way of characterizing problem (1.1) is based on an equivalent mixed-integer reformulation with complementarity-type constraints (Burdakov et al., 2016):

$$\begin{aligned} \min_{x,y}\ & f(x) \\ \text{s.t. } & e^\top y \geq n - s, \\ & x_i y_i = 0, \quad \forall\, i = 1, \dots, n, \\ & x \in X, \\ & y \in \{0,1\}^n. \end{aligned} \tag{2.3}$$

We will detailedly address later the properties of this reformulation. However, we can immediately observe that only variables x_i corresponding to a null y_i are allowed to be nonzero and that at most s elements of the binary vector y can be equal to zero.

Thus, given a feasible point $(\bar{x}, \bar{y})$ for problem (2.3), the components $I_0(\bar{y})$ give an *active subspace* for x, i.e., those components identify the subspace where the components of x are allowed to be nonzero. We thus have that $I_1(\bar{x}) \subseteq I_0(\bar{y})$.

Note that if $|I_0(\bar{y})| = s$, then $I_0(\bar{y})$ identifies a super support set for $\bar{x}$; on the other hand, if $|I_1(\bar{x})| = |I_0(\bar{y})|$, then $I_0(\bar{y})$ is obviously equal to the support of $\bar{x}$.

Example 2.4. Consider the setting of Example 2.1. To exploit reformulation (2.3), we introduce variables $y \in \{0,1\}^4$ and the constraints

$$e^\top y \geq 2, \qquad x_i y_i = 0,\ \forall i = 1, \dots, n.$$

So, let

$$v = (2, 0, 0, 2), \qquad z = (0, 3, 0, 0).$$

Corresponding feasible values for the binary variables are respectively given by

$$y_v = (0,1,1,0)$$

and

$$y_{z_a} = (1,0,1,1),\ y_{z_b} = (0,0,1,1),\ y_{z_c} = (1,0,0,1),\ y_{z_d} = (1,0,1,0).$$

We can observe that, since z has incomplete support, many vectors y_z exist such that (z, y_z) is feasible for the mixed-integer reformulation. Moreover, we note that $|I_0(y_{z_b})| = |I_0(y_{z_c})| = |I_0(y_{z_d})| = s$, so that the three sets identify super support sets for z, whereas $|I_0(y_{z_a})| = |I_1(z)|$, thus y_{z_a} defines the support of z.

In the following, when referring to reformulation (2.3), we will use the following notation:

$$\mathcal{Y} = \{y \mid y \in \{0,1\}^n,\ e^\top y \geq n - s\},$$
$$\mathcal{X}(y) = \{x \in X \mid x_i y_i = 0\ \forall\, i = 1, \dots, n\}.$$

Note that if $y \in \mathcal{Y}$, then $\mathcal{X}(y) = X_{I_0(y)}$.

2.2 Conditions for Optimality

As typically done with continuous optimization problems, necessary conditions of local optimality have also been analyzed for problem (1.1).

In this Section we first analyze the properties of optimal solutions of the problem in the case $X = \mathbb{R}^n$, i.e., when the cardinality constraint is the only constraint. This setting allows us to present in depth the crucial aspects of sparse optimization problems. Afterwards, we turn to the general, yet less intuitive setting, which can be analyzed from diverse perspectives.

The Case $X = \mathbb{R}^n$

As we have outlined at the beginning of this Chapter, the concept of local minimizer itself is rather weak from a practical point of view when delaing with problems of the form (1.1). For this reason, necessary conditions of global optimality, that are not necessarily conditions of local optimality Beck has directly analyzed, have directly been analyzed (Beck and Eldar, 2013).

The simplest optimality condition for problem (1.1) is the following.

Definition 2.4 (BF vectors, case $X = \mathbb{R}^n$)**.** Let $X = \mathbb{R}^n$. We say that a point $\bar{x} \in \mathcal{X}$ is a *basic feasible (BF) vector*, if:

- when $\|\bar{x}\|_0 = s$, it holds $\nabla_i f(\bar{x}) = 0$ for all $i \in I_1(\hat{x})$;
- when $\|\bar{x}\|_0 < s$, it holds $\nabla_i f(\bar{x}) = 0$ for all $i = 1, \dots, n$.

The basic feasibility property is a local optimality condition; in practice, for a basic feasible point, there is not a feasible descent direction: basic feasibility is the most direct extension of the classical stationarity concept to problems with a sparsity constraint. However, being a local optimality condition, the BF property does not characterize the quality of the specific support. In order to get over this limitation, the following concept can firstly be introduced.

Definition 2.5 (L-stationarity, case $X = \mathbb{R}^n$). Let $X = \mathbb{R}^n$. A vector $x^\star \in \mathcal{X}$ is called an *L-stationary point* if it satisfies the relation

$$x^\star \in \Pi_{\mathcal{X}}\left(x^\star - \frac{1}{L}\nabla f(x^\star)\right).$$

L-stationarity resembles the stationarity condition commonly employed with convexly-constrained problem (see Appendix A), where the sparse projection operator is used in place of the standard projection. Since, as remarked in Section 2.1, the operator $\Pi_{\mathcal{X}}$ is easily accessible when $X = \mathbb{R}^n$, checking for L-stationarity is a simple way to assess whether small steps along the full gradients allow to identify a better support or not.

However, L-stationarity is again a very local property; moreover, as we will remark in the following, it requires Lipschitz-continuity assumptions to be employable. A stronger optimality condition, that can be practically employed with convex objectives, is the following.

Definition 2.6 (CW-minimum). Let $X = \mathbb{R}^n$. A vector $x^\star$ is a *component-wise (CW) minimum* if one of the following cases holds true:

- $\|x^\star\|_0 < s$ and for every $i = 1, \dots, n$ one has
$$f(x^\star) = \min_{t \in \mathbb{R}} f(x^\star + te_i);$$
- $\|x^\star\|_0 = s$ and for every $i \in I_1(x^\star)$ and $j = 1, \dots, n$ one has
$$f(x^\star) \leq \min_{t \in \mathbb{R}} f(x^\star - x^\star_i e_i + te_j).$$

For a CW-optimal solution, there is no way of improving the objective function changing the value of a single variable, or, when the support is full, performing a swap operation, i.e., one variable in the support is zeroed while a variable out of the support is moved away from zero. It is clear that a globally optimal solution is necessarily a CW point, but this is not necessarily true for a local optimizer; indeed, CW-optimality allows to take into account changes in the support in the form of swaps. Moreover, the condition is based on global information on the objective function, as it requires to check global optimality w.r.t. single variables. For this reason, it is indeed a quite strong condition, but it is not practically employable without convexity assumptions on the objective function.

The formal relationships existing between the three above optimality conditions are summarized in the following proposition.

Proposition 2.1 (Beck and Eldar 2013)**.** *Let $X = \mathbb{R}^n$ and $x^\star \in \mathcal{X}$. The following statements hold:*

(i) If $x^\star$ is a global minimizer of (1.1)*, then $x^\star$ is a CW-minimum.*

(ii) If $x^\star$ is a CW-minimum, then it is a BF vector. Moreover, if $\nabla f(x)$ is Lipschitz continuous over $\mathbb{R}^n$, then $x^\star$ is L-stationary for all $L > L(f)$, where $L(f)$ is the Lipschitz constant of ∇f.

(iii) If $x^\star$ is an L-stationary point for some $L > 0$, then $x^\star$ is a BF point.

Example 2.5. Consider the optimization problem

$$\begin{aligned}\min_{x\in\mathbb{R}^2}\ & (x_1-2)^2 + \left(\frac{1}{4}x_2^4 - \frac{1}{3}x_2^3 - \frac{9}{2}x_2^2 + 9x_2\right)\\ \text{s.t. } & \|x\|_0 \le 1.\end{aligned}$$

The gradients of the objective function are given by

$$\nabla_{x_1} f(x) = 2x_1 - 4, \qquad \nabla_{x_2} f(x) = (x_2-3)(x_2+3)(x_2-1).$$

We thus have 4 BF points:

$$\bar{x}_a = (2,0), \quad \bar{x}_b = (0,3), \quad \bar{x}_c = (0,-3), \quad \bar{x}_d = (0,1).$$

$\bar{x}_a$ is a local minimizer ($f(\bar{x}_a) = 0$); however, we can observe that it is not a CW point, since $\|\bar{x}_a\|_0 = 1$ and by a single swap of variables in the support we can obtain $\bar{x}_c = (0,-3)$ with $f(\bar{x}_c) = -153/4$. In fact, $\bar{x}_c$ is the global minimizer and is the only CW optimal point: $\bar{x}_b$ is a local minimizer but not a CW point, while $\bar{x}_d$ is a local maximizer.

Note that $\bar{x}_a$, for instance, is L-stationary for all $L > 9/2$ while it is not for $L < 9/2$; indeed we have

$$\bar{x}_a - \frac{1}{L}\nabla f(\bar{x}_a) = (2,0) - \frac{1}{L}(0,9) = (2,-9/L),$$

so that the sparse projection operator returns

$$\Pi_{\mathcal{X}}((2, 9/L)) = \begin{cases} (0, -9/L) & \text{if } L < 9/2, \\ (2, 0) & \text{if } L > 9/2. \end{cases}$$

Similarly, it is easy to check that x_b and x_c are L-stationary for $L > 4/3$ and x_d is L-stationary for $L > 4$. Hence, we have

- x_a, x_b, x_c, x_d are L-stationary if $L > 9/2$;
- x_b, x_c, x_d are L-stationary if $4 < L \leq 9/2$;
- x_b, x_c are L-stationary if $4/3 < L \leq 4$;
- no point is L-stationary if $L < 4/3$.

We can thus observe how crucial is the value of L for L-stationarity: if L is too large, all BF points are L-stationary; on the other hand, if L is too small, L-stationarity may not be a necessary condition of optimality. In order to properly select L, in cases (unlike the above example) where ∇f is Lipschitz-continuous, the knowledge of $L(f)$ would be beneficial.

The three above conditions, even though significant, might however be difficult to enforce with algorithms with nonconvex objectives. Therefore, it is also reasonable to introduce a less restrictive condition for optimality.

Definition 2.7 (Lu-Zhang conditions, case $X = \mathbb{R}^n$)**.** Let $X = \mathbb{R}^n$. We say that a point $\bar{x} \in \mathcal{X}$ satisfies *Lu-Zhang (LZ) first-order optimality conditions* if there exists a super support set $J \in \mathcal{J}(\bar{x})$ $\nabla_i f(\bar{x}) = 0$ for all $i \in J$.

In the case $\|\bar{x}\|_0 = s$, the only super support set is the support itself and Lu-Zhang trivially coincide with basic feasibility. In general, however, the BF property is stronger than LZ conditions. A BF point satisfies LZ condition for all $J \in \mathcal{J}(\bar{x})$. We show this by the following example.

Example 2.6. Consider problem (1.1), letting

$$f(x) = (x_1 - 1)^2 + x_2^2 + (x_3 - 1)^2$$

and $s = 2$. The point $\bar{x} = (1, 0, 0)$ satisfies Lu-Zhang conditions, but it is not a BF-vector. Indeed, let $J = \{1, 2\}$. We have that $\bar{x}_j = 0$ for all $j \in \bar{J}$, i.e., J is a super support set, and $\nabla_i f(\bar{x}) = 0$ for all $i \in J$. Thus $\bar{x}$ satisfies Lu-Zhang conditions. On the other hand, $\|\bar{x}\|_0 < 2$, and $\nabla_3 f(\bar{x}) \neq 0$, i.e., it is not a BF-vector (LZ conditions are not satisfied by the super support set $\{1, 3\}$).

The General Case with Convex Constraints

We are now ready to turn to the more general case where $X \subset \mathbb{R}^n$, i.e., when the sparsity constraint is considered in conjunction with another set of constraints which we assume to be convex.

Optimality can be characterized based on first order information. In particular, when X can be expressed as a set of equality and inequalities, KKT-fashioned conditions can be expressed as follows:

Definition 2.8 (Lu-Zhang Conditions)**.** A point $\bar{x} \in \mathcal{X}$ satisfies *Lu-Zhang (LZ) conditions* for problem (1.1) if there exist a super support set $J \in \mathcal{J}(\bar{x})$ and multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$, $\gamma \in \mathbb{R}^n$ such that

$$\begin{aligned} \nabla f(\bar{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\bar{x}) + \sum_{i=1}^{p} \mu_i \nabla h_i(\bar{x}) + \sum_{i=1}^{n} \gamma_i e_i = 0,& \\ \lambda_i \geq 0,\ \lambda_i g_i(\bar{x}) = 0,\ \forall i = 1, \dots, m,& \\ \gamma_i = 0,\ \forall i \in J.& \end{aligned} \tag{2.4}$$

The following proposition holds:

Proposition 2.2 (Lu and Zhang 2013)**.** *Let $x^\star \in \mathcal{X}$ be a local minimizer for problem* (1.1) *and $J \in \mathcal{J}(x^\star)$ be a super support set. Assume Slater's condition is satisfied by the constraints*

$$h(x) = 0, \qquad g(x) \leq 0, \qquad x_{\bar{J}} = 0, \tag{2.5}$$

where $\bar{J} = \{1, \dots, n\} \setminus J$. Then $x^\star$ satisfies conditions (2.4) *for some λ, μ and γ, i.e., $x^\star$ satisfies Lu-Zhang conditions for problem* (1.1).

Basically, Lu-Zhang conditions extend to a more general case Definition 2.7, being necessary conditions of local optimality for problem (1.1) under constraint qualifications (CQs). In the original work from Lu and Zhang (2013), the Robinson condition is considered as CQ, but in fact we deduce from Ruszczynski (2011, Chapter 3) that, since we consider a convex set X, Slater's condition associated with constraints (2.5) leads to the same result.

As in the case $X = \mathbb{R}^n$, LZ conditions are a rather weak property, since only one out of many possible super support sets is sufficient to make them hold true. A more restrictive property is what we refer to as strong Lu-Zhang conditions.

Definition 2.9 (Strong LZ Conditions)**.** A point $\bar{x} \in \mathcal{X}$ satisfies *strong Lu-Zhang (SLZ) conditions* for problem (1.1) if for every super support set $J \in \mathcal{J}(\bar{x})$ there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$, $\gamma \in \mathbb{R}^n$ such that conditions (2.4) hold.

Strong Lu-Zhang conditions substantially require that basic Lu-Zhang conditions are satisfied for each possible super support set. Note that in the case $\|\bar{x}\|_0 = s$ there is only one super support set, coinciding with $I_1(\bar{x})$, and hence SLZ and LZ conditions are equivalent.

If the problem is convex, except for the ℓ_0 term, i.e., if we assume that f is convex, then strong LZ conditions are sufficient and necessary conditions of (local) optimality.

Proposition 2.3 (Lu and Zhang 2013). *Assume that f is a convex function and that $x^\star \in \mathcal{X}$ satisfies strong Lu-Zhang conditions for problem* (1.1)*. Then, $x^\star$ is a local minimizer of problem* (1.1)*.*

In case X cannot be defined as a set of equality ($h(x) = 0$) and inequality ($g(x) \leq 0$) constraints, or constraints do not satisfy suitable CQs, stationarity can be characterized, as in the smooth convex case, by means of the projection operators.

Definition 2.10 (BF vectors)**.** We say that a point $\bar{x} \in \mathcal{X}$ is a *basic feasible (BF) vector* for problem (1.1) if, for every super support set $J \in \mathcal{J}(x^\star)$, there exists $L > 0$ such that:

$$x^\star = \Pi_{\mathcal{X}_J}(x^\star + d), \qquad d_i = \begin{cases} -\frac{1}{L}\nabla_i f(x^\star) & \text{if } i \in J \\ d_i = 0 & \text{otherwise.} \end{cases}$$

Definition 2.11 (L-stationarity)**.** A vector $x^\star \in \mathcal{X}$ is called an *L-stationary point* of problem (1.1) if it satisfies the relation

$$x^\star \in \Pi_{\mathcal{X}}\left(x^\star - \frac{1}{L}\nabla f(x^\star)\right).$$

As can be easily observed, Definition 2.10 extends Definition 2.4 to the case $X \subset \mathbb{R}^n$, even though this more general case is somewhat more complicated, relying on super support sets. On the contrary, L-stationarity is defined in the same exact way as in Definition 2.5.

The following results hold for BF and L-stationary points.

Proposition 2.4 (Beck and Hallak 2016)**.** *Let $x^\star \in \mathcal{X}$. Then the following statements hold:*

(i) If $x^\star$ is a global minimizer of problem (1.1)*, then $x^\star$ is basic feasible.*

(ii) If $x^\star$ is an L-stationary point for problem (1.1) *for some $L > 0$, then $x^\star$ is basic feasible.*

(iii) If $x^\star$ is a global minimizer of problem (1.1) *and $\nabla f(x)$ is a Lipschitz-continuous function, then $x^\star$ is L-stationary for any $L > L(f)$, being $L(f)$ the Lipschitz constant of ∇f.*

We should highlight that, similarly as in the case $X = \mathbb{R}^n$, the BF notion does not say anything about the optimality of the support set; basic feasibility is indeed a necessary condition of local optimality: the reasoning of the proof of Beck and Hallak (2016, Theorem 5.1), where $x^\star$ is assumed to be a global minimizer, identically holds if the point is a local minimizer. L-stationarity is instead necessary for global minimizers, but not for local ones.

However, the sparse projection operation onto the nonconvex set $\mathcal{X}$, which is easy when $X = \mathbb{R}^n$, is a complex operation, practically available only in particular cases (Beck and Hallak, 2016). Moreover, L-stationarity again requires gradients Lipschitz-continuity and the knowledge of the Lipschitz constant $L(f)$ to be useful in practice.

On the other hand, the orthogonal projection onto the convex set $\mathcal{X}_J$ is doable, although often expensive. Hence, basic feasibility is a generally much more affordable condition to consider than L-stationarity.

Remark 2.1. CW-optimality is not extendable to the more general case $X \subset \mathbb{R}^n$. The reason is that such condition is designed to consider the variables of the problem independently; unless X defines bounds, the constraints are not completely separable and thus the single variable optimization problems in Definition 2.6 are pointless.

An alternative path can be followed to characterize solutions of problem (1.1), exploiting reformulation (2.3) (Burdakov et al., 2016). The mixed-integer programming problem can be relaxed into the following smooth problem:

$$\begin{aligned}\min_{x,y}\ & f(x)\\ \text{s.t. } & e^\top y \geq n - s,\\ & x_i y_i = 0, \quad \forall\, i = 1, \dots, n,\\ & x \in X,\\ & 0 \leq y_i \leq 1, \quad \forall\, i = 1, \dots, n.\end{aligned} \tag{2.6}$$

Equivalence relationships between problems (1.1) and (2.6) have been proved.

Proposition 2.5 (Burdakov et al. 2016). *Let $x^\star \in \mathbb{R}^n$. $(x^\star, y^\star)$. Then the following statements hold:*

1. *$x^\star$ is feasible for problem* (1.1) *if and only if there exists $y^\star \in \mathbb{R}^n$ such that $(x^\star, y^\star)$ is feasible for problem* (2.6)*.*
2. *$x^\star$ is a global minimizer of problem* (1.1) *if and only if there exists $y^\star \in \mathbb{R}^n$ such that $(x^\star, y^\star)$ is a global minimizer of problem* (2.6)*.*
3. *If $x^\star$ is a local minimizer for problem* (1.1)*, then there exists $y^\star \in \mathbb{R}^n$ such that $(x^\star, y^\star)$ is a local minimizer of problem* (2.6)*. Conversely, if $(x^\star, y^\star)$ is a local minimizer for problem* (2.6) *and $\|x^\star\|_0 = s$, then $x^\star$ is a local minimizer of problem* (1.1)*.*

There is thus a full correspondence between global optima $x^\star$ of (1.1) and global optima $(x^\star, y^\star)$ of (2.6). Local minimizers of problem (1.1) are local minimizers of (2.6), whereas the vice versa is necessarily true only for points such that $\|x^\star\|_0 = s$.

Since the relaxed problem (2.6) is a continuous optimization problem, we are also able to define suitable stationarity concepts to characterize its solutions and, consequently, solutions of the original problem (1.1). For example, it can be shown that the feasible set of problem (2.6), in the specific case when X is polyhedral convex, satisfies a suitable constraint qualification. In other words, KKT stationarity is indeed a necessary optimality condition. In general, however, the feasible set of problem (2.6) may violate each standard constraint qualification.

For this reason, it is more appropriate to introduce tailored definitions of stationarity.

Definition 2.12 (S/M-stationarity)**.** Let $(x^\star, y^\star)$ be feasible for the relaxed problem (2.6). Then $(x^\star, y^\star)$ is called

(a) *S-stationary* (S = strong) if there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that the following conditions hold:

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^m \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^p \mu_i \nabla h_i(x^\star) + \sum_{i=1}^n \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0,\ \forall i \in I_0(y^\star). \end{aligned} \tag{2.7}$$

(b) *M-stationary* (M = Mordukhovich) if there exist multipliers $\lambda \in \mathbb{R}^m, \mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that the following conditions hold:

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^m \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^p \mu_i \nabla h_i(x^\star) + \sum_{i=1}^n \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0,\ \forall i \in I_1(x^\star). \end{aligned} \tag{2.8}$$

Note that M-stationarity is a weaker condition than S-stationarity, as it does not impose conditions on the components for which both $x_i^\star$ and $y_i^\star$ are equal to 0. S-stationarity can be shown to be equivalent to KKT stationarity. As a consequence, M-stationarity is a weaker concept than KKT stationarity.

Note that, while S-stationarity depends on $y^\star$, M-stationarity is a condition that actually only depends on the original variable $x^\star$. In other words, a "wrong" vector $y^\star$ can destroy S-stationarity, while an M-stationary point $x^\star$ remains M-stationary independently of the vector $y^\star$ that is associated to it, as long as $(x^\star, y^\star)$ is feasible.

M-stationarity can be further understood realizing that $(x^\star, y^\star)$ is M-stationary if and only if $x^\star$ is a KKT point for the problem

$$\begin{aligned} \min_x \; & f(x) \\ \text{s.t. } & x \in X, \\ & x_i = 0 \quad \forall\, i \in I_0(x^\star). \end{aligned} \tag{2.9}$$

Hence, we can directly talk about $x^\star$ as an M-stationary point for problem (1.1).

The strength of the mentioned stationarity conditions is explicitly stated in the next Proposition.

Proposition 2.6 (Burdakov et al. 2016)**.** *Let $x^\star \in \mathcal{X}$ be a feasible point for problem* (1.1) *and let $y^\star \in \mathbb{R}^n$ such that $(x^\star, y^\star)$ is feasible for problem* (2.6)*. Then, the following statements hold:*

(i) If $(x^\star, y^\star)$ is a local minimizer for problem (2.6) *and X is polyhedral, i.e., functions g_i are affine, then $(x^\star, y^\star)$ satisfies KKT conditions for problem* (2.6)*.*

(ii) $(x^\star, y^\star)$ satisfies KKT conditions for problem (2.6) *if and only if $(x^\star, y^\star)$ is S-stationary for* (2.6)*.*

(iii) If $(x^\star, y^\star)$ is S-stationary for (2.6)*, then $x^\star$ is M-stationary for problem* (1.1)*.*

(iv) If $x^\star$ is a local optimizer for problem (1.1) *and some suitable CQ holds, then $x^\star$ is M-stationary for problem* (1.1)*.*

2.3 A Unified View

In the previous Section, we introduced a set of diverse properties to characterize optimizers of problem (1.1). These conditions have been well established in the specialized literature; however, to the best of our knowledge, a thorough analysis of how some of them relate to others has not been carried out.

Here, we aim at building a bridge between the various points of view. In the next Proposition, we state and prove, when needed, the missing pieces to build, together with Propositions 2.2-2.6, a complete hierarchy of optimality conditions for problem (1.1).

Proposition 2.7. *Consider problem* (1.1) *and a point $x^\star \in \mathcal{X}$. The following statements hold:*

1. *If $x^\star$ satisfies strong Lu-Zhang conditions for problem* (1.1)*, then $x^\star$ is basic feasible for* (1.1)*;*

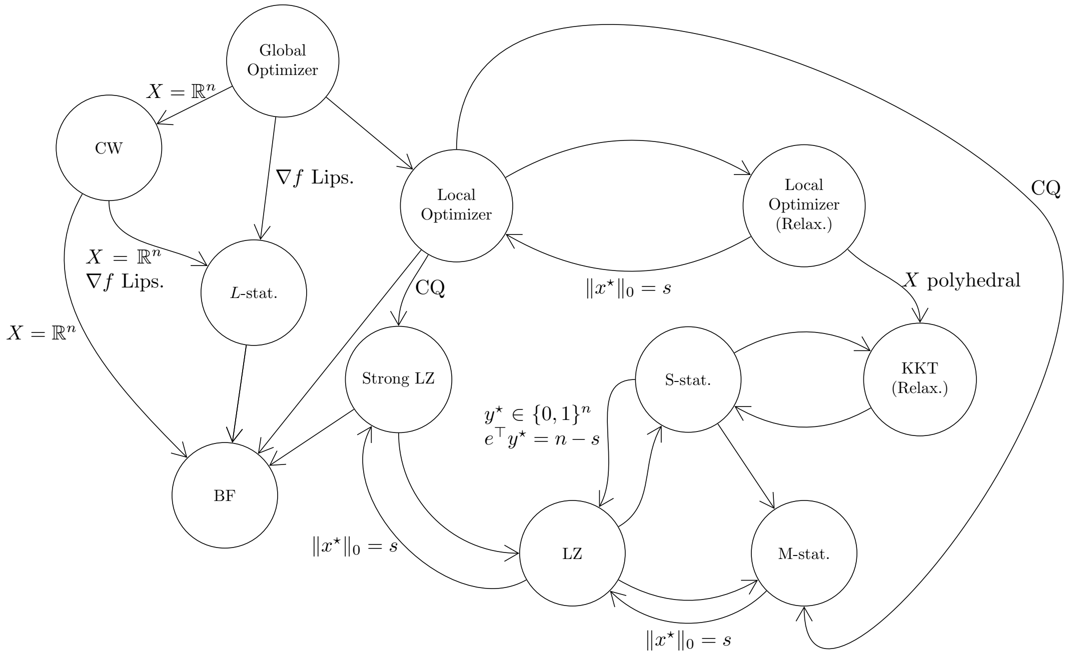


Figure 2.1: Chain of implications between necessary optimality conditions for problem (1.1).

2. *If $x^\star$ satisfies Lu-Zhang conditions for problem* (1.1)*, then $x^\star$ is M-stationary for problem* (1.1)*;*

3. *If $x^\star$ satisfies Lu-Zhang conditions for problem* (1.1)*, then there exists $y^\star$ such that $(x^\star, y^\star)$ is S-stationary for problem* (2.6)*;*

4. *If $(x^\star, y^\star)$ is S-stationary for problem* (2.6)*, $y^\star \in \{0,1\}^n$ and $e^\top y = n - s$, then $x^\star$ satisfies Lu-Zhang conditions for problem* (1.1)*.*

5. *If $x^\star$ is M-stationary for problem* (1.1) *and $\|x^\star\|_0 = s$, then $x^\star$ satisfies (strong) Lu-Zhang conditions.*

Proof. We provide the proof of each statement:

1. Let $J \in \mathcal{J}(x^\star)$ be any super support set. Since $x^\star$ satisfies strong Lu-Zhang

conditions, there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$, $\gamma \in \mathbb{R}^n$ such that

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0\ \forall i \in J. \end{aligned}$$

Therefore, there exist multipliers such that

$$\begin{aligned} \nabla_J f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla_J g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla_J h_i(x^\star) = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall\, i \in J. \end{aligned}$$

Now, let $X(J) \subseteq \mathbb{R}^s$ be the feasible set for the super support set J. From the above equation and Proposition A.1, we get $x_J^\star = \Pi_{X(J)}[x_J^\star - \nabla_J f(x^\star)]$. Therefore, letting d be such that $d_J = -\nabla_J f(x^\star)$ and $d_{\bar{J}} = 0$, recalling $x_{\bar{J}}^\star = 0$, we get $x^\star = \Pi_{\mathcal{X}_J}[x^\star + d]$, which completes the proof, since J is an arbitrary super support set.

2. Let $x^\star \in \mathcal{X}$ satisfy Lu-Zhang conditions for problem (1.1), i.e., there exist a super support set $J \in \mathcal{J}(x^\star)$ and multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0\ \forall i \in J. \end{aligned}$$

Since $I_1(x^\star) \subseteq J$, we have that λ, μ and γ satisfy

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0\ \forall i \in I_1(x^\star), \end{aligned}$$

i.e., $x^\star$ is M-stationary.

3. By the assumptions, $x^\star$ satisfies Lu-Zhang conditions, hence there exists a super support set $J \in \mathcal{J}(x^\star)$ and multipliers such that (2.4) holds. Now, let $y^\star \in \{0,1\}^n$ be such that $y_i^\star = 0$ for all $i \in J$, and $y_i^\star = 1$ otherwise. Then, $I_0(y^\star) = J$ and $x^\star$ satisfies (2.7).

4. Let $(x^\star, y^\star)$ be an S-stationary point such that $y^\star \in \{0,1\}^n$ and $e^\top y^\star = n - s$. Clearly, $|I_0(y^\star)| = s$, hence $I_0(y^\star) \in \mathcal{J}(x^\star)$; by definition of S-stationarity, there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$, $\gamma \in \mathbb{R}^n$ such that

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0\ \forall i \in I_0(y^\star), \end{aligned}$$

which completes the proof since $I_0(y^\star)$ is a super support set.

5. Since, when $\|x^\star\| = s$, $I_1(x^\star)$ is the (unique) super support set, we have that conditions (2.8) and (2.4) coincide. The proof straightforwardly follows.

□

We also graphically show the full chain of implications in Figure 2.1.

The stationarity conditions summarized above can be interpreted as follows; M-stationarity can be seen as KKT-stationarity with respect to the variables in the support; S-stationarity is KKT-stationarity w.r.t. a specific active subspace; Lu-Zhang conditions represent KKT-stationarity with respect to at least one super support set; strong Lu-Zhang conditions are KKT-stationarity with respect to any possible super support set; these four conditions coincide when $\|x^\star\|_0 = s$.

When $y^\star$ has integer components and $e^\top y^\star = n - s$, S-stationarity is substantially equivalent to Lu-Zhang conditions, as the active set identified by $I_0(y^\star)$ corresponds to a super support set. Basic feasibility denotes classical continuous stationarity over convex sets (see Appendix A) w.r.t. any possible super support set; similarly as in the continuous case, KKT-stationarity w.r.t. a super support set implies stationarity w.r.t. that super support set, hence strong Lu-Zhang conditions imply basic feasibility. The converse is true under suitable constraints qualification (see Appendix A).

2.4 A General Condition: $\mathcal{N}$-stationarity

The necessary conditions for global optimality described in the previous Sections hardly go beyond the weak concept of local minimum. In fact, even conditions that allow to consider possible changes to the structure of the support set and reduce the pool of optimal candidates are either tailored to the "unconstrained case", or limited to moderate changes in the support, or involve difficult operations, such as exact minimizations or projections onto nonconvex sets. In order to introduce a general and affordable necessary optimality condition that also takes into account the combinatorial nature of the problem, we exploit in our analysis the mixed-integer reformulation (2.3).

Nonlinear mixed-integer programs can be characterized exploiting the notion of *neighborhood* (Lucidi et al., 2005; Li and Sun, 2006).

Definition 2.13. Let $(\bar{x}, \bar{y}) \in \mathcal{X}(\bar{y}) \times \mathcal{Y}$ a feasible point for problem (2.3). A *discrete neighborhood* $\mathcal{N}(\bar{x}, \bar{y})$ is a set of points such that:

- $(\bar{x}, \bar{y}) \in \mathcal{N}(\bar{x}, \bar{y})$;
- $(\hat{x}, \hat{y}) \in \mathcal{X}(\hat{y}) \times \mathcal{Y}$ for all $(\hat{x}, \hat{y}) \in \mathcal{N}(\bar{x}, \bar{y})$;
- $|\mathcal{N}(\bar{x}, \bar{y})| < \infty$.

Basically, given a feasible point (x, y), a discrete neighborhood $\mathcal{N}(x, y)$ is a finite set of feasible points that contains (x, y) itself. Of course, in order for the concept of neighborhood to be practically meaningful, the points in it should be close, to some extent, to the center (x, y); however, the formalization of this feature will be deferred to the definition of each specific neighborhood.

We introduce here an example of a tailored neighborhood for problem (2.3) that can be implemented at a reasonable computational cost. Such a neighborhood will also help us to relate our analysis to the other theoretical tools available in the literature.

Definition 2.14 ($\mathcal{N}_\rho$ neighborhood)**.** Let $d_H : \{0,1\}^n \times \{0,1\}^n \to \mathbb{N}$ denote the Hamming distance. Moreover, let $\Delta(y, \hat{y}) = \{i \mid y_i \neq \hat{y}_i\}$ and let $H_{\Delta(y,\hat{y})}(\cdot)$ be a function such that $\hat{x} = H_{\Delta(y,\hat{y})}(x)$ is defined as

$$(H_{\Delta(y,\hat{y})}(x))_h = \begin{cases} 0 & \text{if } h \in \Delta(y, \hat{y}), \\ x_h & \text{otherwise.} \end{cases}$$

Then, given $\rho \in \mathbb{N}$, the neighborhood $\mathcal{N}_\rho$ is defined as

$$\mathcal{N}_\rho(x, y) = \left\{ (\hat{x}, \hat{y}) \mid \hat{y} \in \mathcal{Y}, \mathcal{X}(\hat{y}) \neq \emptyset,\ d_H(\hat{y}, y) \leq \rho,\ \hat{x} = \Pi_{\mathcal{X}(\hat{y})}(H_{\Delta(y,\hat{y})}(x)) \right\}. \tag{2.10}$$

Basically, the neighborhood contains points $(\hat{x}, \hat{y})$ with at most ρ components of $\hat{y}$ differing from y; $\hat{x}$ is obtained by zeroing components of x as needed to maintain feasibility w.r.t. the complementarity constraints and then by projecting the result onto the (convex) active feasible set $X(\hat{y})$. In other words, this particular definition of neighborhood allows to take into account the potential "change of status" of up to ρ variables in the vector $\hat{y}$ defining an active subspace.

Example 2.7. Consider the problem (2.3) with $n = 3$, $s = 2$, $X = \mathbb{R}^n$ and let $\rho = 2$. Let (x, y) be a feasible point defined as follows

$$(x, y) = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

The neighborhood $\mathcal{N}_\rho(x, y)$ is given by

$$\mathcal{N}_2(x,y) = \left\{ \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\}.$$

Now, a notion of local optimality for the mixed-integer problem (2.3), depending on the neighborhood $\mathcal{N}(x, y)$, can be introduced:

Definition 2.15 ($\mathcal{N}$-local minimizer for (2.3))**.** A point $(x^\star, y^\star) \in \mathcal{X}(y^\star) \times \mathcal{Y}$ is an *$\mathcal{N}$-local minimizer of problem* (2.3) if there exists an $\epsilon > 0$ such that for all $(\hat{x}, \hat{y}) \in \mathcal{N}(x^\star, y^\star)$ it holds

$$f(x^\star) \leq f(x) \quad \forall\, x \in \mathcal{B}(\hat{x}, \epsilon) \cap \mathcal{X}(\hat{y}).$$

Note that in the above definition the continuous nature of the problem, expressed by the variables x, is taken into account by means of the standard ball $\mathcal{B}(\hat{x}, \epsilon)$. The given definition clearly depends on the choice of the neighborhoods. A larger neighborhood $\mathcal{N}(x^\star, y^\star)$ should give a better local minimizer, but the computational effort needed to locate the solution may increase.

Inspired by the definition of local optimality for problem (2.3), we introduce a necessary condition of global optimality for problem (1.1) that allows to take into account possible, beneficial changes of the support and that hence properly captures, from an applied point of view, the essence of the problem.

Such a condition relies on the use of stationary points related to continuous problems obtained by fixing the binary variables in problem (2.3), i.e., for a fixed $\bar{y} \in \mathcal{Y}$,

$$\begin{aligned} &\min f(x) \\ &\ \text{s.t. } x \in \mathcal{X}(\bar{y}). \end{aligned} \tag{2.11}$$

Definition 2.16 ($\mathcal{N}$-stationarity)**.** A point $x^\star \in \mathcal{X}$ is called an *$\mathcal{N}$-stationary point* for problem (1.1) if there exists an $y^\star \in \mathcal{Y}$ such that

(i) $(x^\star, y^\star)$ is feasible for problem (2.3), i.e., $(x^\star, y^\star) \in \mathcal{X}(y^\star) \times \mathcal{Y}$;

(ii) the point $x^\star$ is a stationary point of the continuous problem

$$\begin{aligned} &\min f(x) \\ &\ \text{s.t. } x \in \mathcal{X}(y^\star); \end{aligned}$$

(iii) every $(\hat{x}, \hat{y}) \in \mathcal{N}(x^\star, y^\star)$ satisfies $f(\hat{x}) \geq f(x^\star)$ and if $f(\hat{x}) = f(x^\star)$, the point $\hat{x}$ is a stationary point of the continuous problem

$$\begin{aligned} &\min f(x) \\ &\ \text{s.t. } x \in \mathcal{X}(\hat{y}). \end{aligned}$$

It is easy to see that the following result stands:

Proposition 2.8. *Let $x^\star$ be a minimum point of problem* (1.1)*. Then $x^\star$ is an $\mathcal{N}$-stationary point.*

In Definition 2.16 we generically refer to stationary points of problem (2.11), namely, to points satisfying suitable optimality conditions. Then, concerning the assumptions on the feasible set $\mathcal{X}(\bar{y})$, we may distinguish the two cases:

(a) no constraint qualifications hold;

(b) constraint qualifications are satisfied and the usual KKT theory can be applied.

In case (a), we will refer to the following definition of stationary point of problem (2.11).

Definition 2.17. Given $\bar{y} \in \mathcal{Y}$ and $\bar{x} \in \mathcal{X}(\bar{y})$, we say that $\bar{x}$ is a stationary point of problem (2.11) if and only if

$$\bar{x} = \Pi_{\mathcal{X}(\bar{y})}\left[\bar{x} - \nabla f(\bar{x})\right].$$

We notice that $\mathcal{X}(\bar{y})$ is a convex set when X is convex, then the condition given above is the classic stationarity condition for the problem (2.11), also discussed in Appendix A.

In case (b) instead, KKT-stationarity will be considered. The relation between KKT-stationarity and projection-based stationarity is thoroughly addressed in Appendix A.

In the next few subsections, we will analyze the relationships between $\mathcal{N}$-stationarity and the optimality conditions previously discussed in this Chapter. In particular, we will show how the definition of $\mathcal{N}$-stationarity allows to retrieve in a unified view most of the known optimality conditions, if a suitable neighborhood $\mathcal{N}$ is employed.

$\mathcal{N}$-stationarity and S-stationarity

It is fairly easy to see that, when KKT-stationarity is considered in Definition (2.16), $\mathcal{N}$-stationarity implies S-stationarity. Indeed, we can prove the following proposition.

Proposition 2.9. *Let $x^\star$ be an $\mathcal{N}$-stationary point of problem* (1.1)*, assuming KKT-stationarity for continuous problems is considered. Then, there exists $y^\star \in \mathcal{Y}$ such that $(x^\star, y^\star)$ is an S-stationary point.*

Proof. If $x^\star$ is $\mathcal{N}$-stationary, there exists $y^\star \in \mathcal{Y}$ such that $(x^\star, y^\star)$ is feasible for problem (2.3) (point (i)) and KKT-stationary w.r.t. the following problem (point (ii)):

$$\begin{aligned} \min_x \; & f(x) \\ \text{s.t. } & h_i(x) = 0, \quad \forall i = 1, \dots, p, \\ & g_i(x) \le 0, \quad \forall i = 1, \dots, m, \\ & x_i y_i^\star = 0, \quad \forall i = 1, \dots, n. \end{aligned}$$

Rearranging, the previous problem can be rewritten as

$$\begin{aligned} \min_x \; & f(x) \\ \text{s.t. } & h_i(x) = 0, \quad \forall i = 1, \dots, p, \\ & g_i(x) \le 0, \quad \forall i = 1, \dots, m, \\ & x_i = 0, \qquad \forall i \in I_1(y^\star). \end{aligned}$$

This means that there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that the following conditions hold:

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^m \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^p \mu_i \nabla h_i(x^\star) + \sum_{i \in I_1(y^\star)} \gamma_i e_i = 0, \\ \lambda_i \ge 0, \; \lambda_i g_i(x^\star) = 0, \; \forall i = 1, \dots, m. \end{aligned}$$

That is:

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^m \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^p \mu_i \nabla h_i(x^\star) + \sum_{i=1}^n \gamma_i e_i = 0, \\ \lambda_i \ge 0, \; \lambda_i g_i(x^\star) = 0, \; \forall i = 1, \dots, m, \\ \gamma_i = 0, \; \forall \; i \in I_0(y^\star). \end{aligned}$$

Therefore, $(x^\star, y^\star)$ is an S-stationary point. □

In fact, we can observe that S-stationarity and $\mathcal{N}$-stationarity substantially coincide when the neighborhood

$$\mathcal{N}(x, y) = \{(x, y)\}$$

is considered.

$\mathcal{N}$-stationarity and M-stationarity

Since, when considering KKT-continuous stationarity, $\mathcal{N}$-stationarity implies S-stationarity which is in turn stronger than M-stationarity, we get that even when we use the simplest possible neighborhood

$$\mathcal{N}(x, y) = \{(x, y)\}$$

$\mathcal{N}$-stationarity implies M-stationarity.

To be more precise, with the above neighborhood, the three conditions are basically equivalent, as we show in the following proposition.

Proposition 2.10. *Let $x^\star \in \mathcal{X}$ be an M-stationary point for problem* (1.1)*. Then, $x^\star$ is $\mathcal{N}$-stationary with $\mathcal{N}(x,y) = \{(x,y)\}$.*

Proof. Assume $x^\star \in \mathcal{X}$ is M-stationary. Let $y^\star$ be defined as follows:

$$y_i^\star = \begin{cases} 0 & \text{if } i \in I_1(x^\star), \\ 1 & \text{otherwise.} \end{cases}$$

We have that $x_i^\star y_i^\star = 0$ for all $i = 1, \dots, n$, $x \in X$, $y \in \mathcal{Y}$ and $e^\top y = n - \|x\|_0 \geq n - s$; hence $(x^\star, y^\star) \in \mathcal{X}(y) \times \mathcal{Y}$ and point (i) of Definition 2.16 is satisfied. Since $x^\star$ is M-stationary, it satisfies conditions (2.8). Noting that $I_0(x^\star) = I_1(y^\star)$ by the definition of $y^\star$, we deduce that conditions (2.7) are satisfied and thus, by tracing backwards the reasoning in the proof of Proposition 2.9 we obtain that point (ii) of Definition 2.16 is satisfied. Since $\mathcal{N}(x,y) = \{(x,y)\}$, we have that point (iii) collapses to only check again the condition of point (ii), hence the proof is complete. □

$\mathcal{N}$-stationarity and Lu-Zhang Conditions

As we have shown in Proposition 2.7, S-stationarity and LZ conditions are equivalent when the considered vectors y are binary and satisfy $e^\top y = n - s$. We can also recall that $\mathcal{N}$-stationarity and S-stationarity are equivalent when

$$\mathcal{N}(x,y) = \{(x,y)\}$$

and KKT-stationarity is considered.

We can thus deduce that Lu-Zhang conditions can be retrieved if the above neighborhood is employed and, in Definition 2.16, $y^\star$ is required to satisfy $e^\top y^\star = n - s$, i.e., $I_0(y^\star)$ identifies a super support set for $x^\star$.

On the other hand, strong Lu-Zhang conditions can be retrieved by using the following neighborhood:

$$\mathcal{N}(x^\star, y^\star) = \left\{ (x,y) \mid x = x^\star,\ y \in \{0,1\}^n,\ e^\top y = n - s,\ y_i x_i^\star = 0\ \forall\, i = 1, \dots, n \right\}.$$

Indeed, we can observe that, in the above neighborhood, $x = x^\star$ for all $(x,y) \in \mathcal{N}(x^\star, y^\star)$ and thus $f(x^\star) = f(x)$; therefore, by point (iii) of Definition 2.16, (KKT) stationarity has to be checked for all points in $\mathcal{N}(x^\star, y^\star)$.

The different points in the discrete neighborhood are obtained by changing the y part of the solution in all the possible ways so that the new binary vector identifies a super support set.

Hence, with this neighborhood, a point is $\mathcal{N}$-stationary if and only if it is KKT-stationary w.r.t. any possible super support set, i.e., strong LZ conditions hold.

$\mathcal{N}$-stationarity and Basic Feasibility

By analogous reasonings as those seen in the previous section for strong LZ conditions, we can see that Basic feasibility is obtainable by using, again, the neighborhood

$$\mathcal{N}(x^\star, y^\star) = \Big\{(x,y) \mid x = x^\star,\ y \in \{0,1\}^n,\ e^\top y = n-s,\ y_i x_i^\star = 0\ \forall\, i = 1,\dots,n\Big\},$$

while considering the projection-based concept of continuous stationarity as in Definition 2.17.

We have to note, however, that the BF property requires that, for any super support set $J \in \mathcal{J}(x^\star)$, it holds

$$x^\star = \Pi_{\mathcal{X}_J}[x^\star + d],$$

where $d_J = -\frac{1}{L}\nabla_J f(x^\star)$ and $d_{\bar{J}} = 0$, whereas the condition in Definition 2.17 requires

$$x^\star = \Pi_{\mathcal{X}_J}[x^\star - \nabla f(x^\star)].$$

In fact, in the case of our problem the two conditions are equivalent, as we show below.

Lemma 2.1. *Let $y \in \mathcal{Y}$ and $x^\star \in \mathcal{X}(y)$. Then $x^\star$ satisfies*

$$x^\star = \Pi_{\mathcal{X}(y)}(x^\star + d),$$

where $d_{I_0(y)} = -\frac{1}{L}\nabla_{I_0(y)} f(x^\star)$ and $d_{I_1(y)} = 0$, if and only if it satisfies

$$x^\star = \Pi_{\mathcal{X}(y)}(x^\star - \nabla f(x^\star)).$$

Proof. By the definition of projection, we have for all $z \in \mathbb{R}^n$ that

$$\Pi_{\mathcal{X}(y)}(z) = \underset{\substack{(x_{I_0(y)}, x_{I_1(y)}) \in X \\ x_{I_1(y)} = 0}}{\arg\min} \left\| \begin{matrix} x_{I_0(y)} - z_{I_0(y)} \\ x_{I_1(y)} - z_{I_1(y)} \end{matrix} \right\|^2 = \begin{bmatrix} \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \|x_{I_0(y)} - z_{I_0(y)}\|^2 \\ 0 \end{bmatrix}$$

Hence, we have

$$\Pi_{\mathcal{X}(y)}(x^* - \nabla f(x^*)) = \begin{bmatrix} \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \|x_{I_0(y)} - (x^*_{I_0(y)} - \nabla_{I_0(y)} f(x^*)\|^2 \\ 0 \end{bmatrix}$$

and

$$\Pi_{\mathcal{X}(y)}(x^* + d) = \begin{bmatrix} \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \|x_{I_0(y)} - (x^*_{I_0(y)} - \frac{1}{L}\nabla_{I_0(y)} f(x^*)\|^2 \\ 0 \end{bmatrix}$$

To prove the statement, it is sufficient to show that if

$$x^*_{I_0(y)} = \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \left\| x_{I_0(y)} - (x^*_{I_0(y)} - \frac{1}{L}\nabla_{I_0(y)} f(x^*) \right\|^2$$

for some $L > 0$, then

$$x^*_{I_0(y)} = \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \left\| x_{I_0(y)} - (x^*_{I_0(y)} - \frac{1}{L_2}\nabla_{I_0(y)} f(x^*) \right\|^2$$

for all $L_2 > 0$. Thus, let us assume by contradiction that there exists $L_2 > 0$, $L_2 \neq L$, such that

$$\hat{x}_{I_0(y)} = \underset{x_{I_0(y)}:(x_{I_0(y)},0)\in X}{\arg\min} \left\| x_{I_0(y)} - (x^*_{I_0(y)} - \frac{1}{L_2}\nabla_{I_0(y)} f(x^*) \right\|^2,$$

with $\hat{x}_{I_0(y)} \neq x^*_{I_0(y)}$. By the properties of the projection operator over a convex set, we get:

$$\left(x^*_{I_0(y)} - \left(x^*_{I_0(y)} - \frac{1}{L}\nabla_{I_0(y)} f(x^*)\right)\right)^\top (x^*_{I_0(y)} - x_{I_0(y)}) \leq 0 \quad \forall\, x_{I_0(y)} : (x_{I_0(y)}, 0) \in X$$

and

$$\left(\hat{x}_{I_0(y)} - \left(x^*_{I_0(y)} - \frac{1}{L_2}\nabla_{I_0(y) f(x^*)}\right)\right)^\top (\hat{x}_{I_0(y)} - x_{I_0(y)}) \leq 0 \quad \forall\, x_{I_0(y)} : (x_{I_0(y)}, 0) \in X.$$

From the first of the above equations we then obtain

$$\nabla_{I_0(y)} f(x^*)^\top (x^*_{I_0(y)} - \hat{x}_{I_0(y)}) \leq 0,$$

whereas from the second we can write

$$\left(\hat{x}_{I_0(y)} - \left(x^*_{I_0(y)} - \frac{1}{L_2}\nabla_{I_0(y) f(x^*)}\right)\right)^\top (\hat{x}_{I_0(y)} - x^*_{I_0(y)}) \leq 0,$$

and then

$$\|\hat{x}_{I_0(y)} - x^*_{I_0(y)}\|^2 \leq \frac{1}{L_2}\nabla_{I_0(y)} f(x^*)^\top (x^*_{I_0(y)} - \hat{x}_{I_0(y)}) \leq 0,$$

which is absurd. □

$\mathcal{N}$-stationarity, L-stationarity and CW-optimality

The L-stationarity property can hardly be obtained in our framework, since it is based on the sparse projection operation.

As for the concept of CW-optimality for the case $X = \mathbb{R}^n$, the definition is based on argmin operators, i.e., on global information, and thus cannot be directly encapsulated in a concept of stationarity.

However, if we relax the definition, we can think of a CW-stationarity concept, i.e., we can replace the argmin operation w.r.t. a variable by stationarity w.r.t. the same variable.

When $\|x^\star\|_0 < s$, we require $\nabla_i f(x^\star) = 0$ for all $i = 1, \dots, n$ (otherwise, there would be a descent coordinate direction and the objective could be decreased by moving one single variable).

When $\|x^\star\|_0 = s$, we shall instead require $\nabla f(\hat{x}) = 0$ for all points in the set

$$R(x^\star) = \{\hat{x} \mid \hat{x} = x^\star - x_i^\star e_i, i = 1, \dots, n\}$$

and $\nabla_{I_1(x^\star)} f(x^\star) = 0$. The set $R(x^\star)$ contains all points obtained by zeroing one single variable of $x^\star$ (these points thus have incomplete support). We thus want each of these points to be stationary, otherwise there would exist swaps allowing to decrease the objective value. We also want $x^\star$ to be stationary w.r.t. the variables in the support set. Substantially, we want all points in

$$R(x^\star) \cup \{x^\star\}$$

to be basic feasible.

Recalling the way the BF property can be written in terms of discrete neighborhoods, the definition of CW-stationarity can thus be obtained, in the mixed-integer setting, by using the discrete neighborhood

$$\mathcal{N}(x^\star, y^\star) = \begin{cases} \{(x,y) \mid x = x^\star,\ e^T y = n - s,\ y_i x_i^\star = 0\ \forall\, i\} & \text{if } \|x^\star\|_0 < s, \\ \{(x,y) \mid x \in R(x^\star) \cup \{x^\star\},\ e^T y = n - s,\ y_i x_i = 0\ \forall\, i\} & \text{if } \|x^\star\|_0 = s. \end{cases}$$

Chapter 3

Review of State-of-the-art Algorithms

Together with the analysis of solutions of problem (1.1) and the definition of optimality conditions, tailored algorithmic schemes have been developed to tackle nonconvex sparsity-constrained problems in a continuous optimization fashion. These algorithms are, in general, specifically designed to produce solutions that satisfy some of the optimality conditions discussed in Section 2.2.

In this Chapter, we provide a brief overview of the main algorithmic proposals that can be found in the related literature.

3.1 Iterative Hard Thresholding Method

The *Iterative Hard Thresholding* (IHT) approach (Beck and Eldar, 2013) to solve problem (1.1) when $X = \mathbb{R}^n$ basically consists of employing a fixed point method aimed at enforcing the L-stationarity condition. Specifically, iterations of the form

$$x^{k+1} \in \Pi_{\mathcal{X}} \left(x^k - \frac{1}{L} \nabla f(x^k) \right)$$

are performed. The cost of a single iteration of IHT is moderate since, as discussed in Section 2.1, the projection of a solution onto the sparse set is readily available when $X = \mathbb{R}^n$.

Under suitable regularity assumptions, convergence properties can be stated for the IHT algorithm.

Proposition 3.1 (Beck and Eldar 2013). *Let ∇f be Lipschitz-continuous with constant $L(f)$. Let $\{x^k\}$ be the sequence generated by the IHT algorithm with constant stepsize $1/L$ where $L > L(f)$. Then*

(i) $\{f(x^k)\}$ *is a monotone non increasing and thus convergent sequence;*

(ii) $f(x^{k+1}) < f(x^k)$ *if* $x^{k+1} \neq x^k$*;*

(iii) $\|x^{k+1} - x^k\| \to 0$ *as* $k \to \infty$*;*

(iv) any accumulation point $\bar{x}$ *of* $\{x^k\}$ *is an* L*-stationary point.*

The main shortcoming of the IHT algorithm is that its performance strongly depends on the choice of the stepsize $1/L$. An excessively low value of $1/L$ leads to slow convergence and increases the chance not to identify the global optimizer. On the other hand, large stepsizes might not guarantee the convergence of the algorithm. Moreover, the Lipschitz-continuity assumption on the gradients is quite restrictive.

The IHT is also not particularly well suited to be used in the case $X \subset \mathbb{R}^n$. Although the algorithmic scheme can be directly mirrored to the general case (Beck and Hallak, 2016), the sparse projection operation required at each iteration quickly becomes unviable as the feasible set becomes more articulated. Indeed, the approach is practically useful under strong symmetry assumptions on the feasible set.

3.2 Greedy-Sparse Simplex Method

Similarly as the IHT algorithm, that is specifically designed to produce L-stationary points, the *Greedy Sparse-Simplex* (GSS) method (Beck and Eldar, 2013) is devised to generate a sequence of points converging to CW-optima for problem (1.1) in the case $X = \mathbb{R}^n$.

In analogy with the CW-optimality condition, the iterations of the algorithm depend on the cardinality of the current iterate:

- if $\|x^k\|_0 < s$:
    - for all $i = 1, \dots, n$, compute $x^i = x^k + t_i e_i$ where
    $$t_i \in \arg\min_t f(x^k + te_i);$$
    - $x^{k+1} \in \arg\min_{x^i} f(x)$;
- if $\|x^k\|_0 = s$:
    - for all $i = 1, \dots, n$, and $j \in I_1(x^k)$, compute $x^{i,j} = x^k + t_i e_i - x^k_j e_j$ where
    $$t_i \in \arg\min_t f(x^k + te_i - x^k_j e_j);$$
    - $x^{k+1} \in \arg\min_{x^{i,j}} f(x)$.

When the support is incomplete, there is still room for adding variables to it: the algorithm performs the best possible step among all those involving the change of a single variable. When, on the other hand, the support is complete, the pool of possibly considered moves includes those consisting of the zeroing of a variable in the support and the optimal change of another one, hence allowing swap of variables that modify the composition of the active set. The procedure stops when the current iterate is kept fixed after an iteration.

By the definition of the algorithm itself, the following properties should not appear surprising.

Proposition 3.2 (Beck and Eldar 2013)**.** *Let $\{x^k\}$ the sequence produced by the GSS algorithm. Then,*

(i) $f(x^{k+1}) \leq f(x^k)$ *for all* $k = 0, 1, \ldots$*;*

(ii) $f(x^{k+1}) = f(x^k)$ *if and only if* $x^{k+1} = x^k$ *and* x^k *is CW-optimal;*

(iii) If $\{x^k\}$ *is an infinite sequence, any accumulation point* $\bar{x}$ *of* $\{x^k\}$ *is a CW-optimal solution.*

The benefits of the GSS method over the IHT are evident. Indeed, the GSS produces points that satisfy the CW-optimality property, which is stronger than L-stationarity, and does not require the accurate setting of a parameter (L) nor Lipschitz-continuity assumptions.

However, this apparently overall superiority comes not for free. Specifically, the GSS algorithm requires to perform steps of global optimization of subproblems. This operation hides the convexity requisites, at least component-wise. In addition, the cost of iterations is much higher for the GSS than for the IHT: the number of one-variable optimization problems in the case $\|x^k\|_0 = s$ grows as fast as $\binom{n}{s}$, thus the algorithm does not scale well with the problem dimensionality. Moreover, even once the final support has been identified, the algorithm performs a very long tail of (costly) iterations that are needed for obtaining convergence by moving one variable at a time.

The algorithm can be substantially extended to the case $X \subset \mathbb{R}^n$ (Beck and Hallak, 2016). However, in order to do that, strong symmetry assumptions on the feasible set have to be made; in particular, any swap of variables values and any sign change shall lead from a feasible point to another feasible point.

3.3 Regularization Method

A regularization approach has been proposed to tackle problem (1.1) by exploiting the relaxed reformulation (2.6). In particular, to get rid of the hardly manageable complementarity type constraints, the following constraints are introduced:

$\varphi(x_i, y_i; t) \leq 0$, $\tilde{\varphi}(x_i, y_i; t) \leq 0$ for all $i = 1, \dots, n$, where

$$\varphi(a,b;t) = \begin{cases} (a-t)(b-t) & \text{if } a+b \geq 2t, \\ -\frac{1}{2}[(a-t)^2 + (b-t)^2] & \text{if } a+b < 2t, \end{cases}$$

$$\varphi(a,b;t) = \begin{cases} (-a-t)(b-t) & \text{if } -a+b \geq 2t, \\ -\frac{1}{2}[(-a-t)^2 + (b-t)^2] & \text{if } -a+b < 2t. \end{cases}$$

A sequence of regularized problems can be considered for values of t^k such that $t^k \to 0$. Each subproblem is then solved to KKT stationarity. Under rather weak constraints qualification, the sequence of obtained solutions is such that any limit point is an M-stationary solution.

Proposition 3.3 (Burdakov et al. 2016)**.** *Let $\{t^k\}$ be a sequence such that $t_k > 0$ for all k, $t^k \to 0$ as $k \to \infty$. Let $\{(x^k, y^k)\}$ be a sequence of KKT points for the regularized subproblem of parameter t^k such that $(x^k, y^k) \to (\bar{x}, \bar{y})$. Assume that $(\bar{x}, \bar{y})$ satisfies some suitable constraint qualification for problem* (2.6)*. Then $\bar{x}$ is an M-stationary point of problem* (1.1)*.*

The convergence result for the regularization method is of course weaker than those of IHT and GSS, being M-stationarity the weakest condition among those analyzed in Section 2.2. However, this approach can be generally employed in presence of additional constraints.

From a computational perspective, this approach proves to be quite efficient, but evidence of its good performance is limited to a simple and limited benchmark and the quality of the retrieved solutions, from a global optimization point of view, appears to be lacking.

In fact, it has been recently shown (Kanzow et al., 2021) that a probably more effective way to deal with problem (1.1), with still the same convergence guarantees, is to directly tackle the relaxed subproblem (2.6) with a standard Augmented Lagrangian Method (ALM) with multipliers safeguarding (Birgin and Martínez, 2014; Galvan and Lapucci, 2019).

3.4 Penalty Decomposition Approach

Applying the classical variable splitting technique (Jörnsten et al., 1985), Problem (1.1) can be equivalently expressed as

$$\begin{aligned} \min_{x,z \in \mathbb{R}^n} \; & f(x) \\ \text{s.t. } & \|z\|_0 \leq s, \\ & x \in X, \\ & x = z. \end{aligned} \tag{3.1}$$

For simplicity, in the following, we will denote $Z = \{z \in \mathbb{R}^n : \|z\|_0 \leq s\}$.

The quadratic penalty function associated to Problem (3.1) is

$$q_\tau(x,z) = f(x) + \frac{\tau}{2}\left(\|x - z\|^2 + \|h(x)\|^2 + \|g_+(x)\|^2\right),$$

where $\tau > 0$ is the penalty parameter and $g_+(x)$ denotes the component-wise maximum $\max\{0, g(x)\}$.

The Penalty Decomposition (PD) method (Lu and Zhang, 2013), formally defined in Algorithm 1, can be used to solve Problem (1.1) by tackling Problem (3.1). In particular, the approach consists of approximately solving a sequence of penalty subproblems by a two-block decomposition method.

The algorithm starts from a point (x^0, z^0) that is feasible for problem (3.1). At every iteration, the algorithm performs the Block Coordinate Descent (BCD) method (Bertsekas and Tsitsiklis, 1989; Beck and Tetruashvili, 2013) w.r.t. the two blocks of variables x and z, until an approximate stationary point of the penalty function w.r.t. the x block is attained. Then, the penalty parameter τ_k is increased for the successive iteration, where a higher degree of accuracy is required to approximate a stationary point.

Note that, as discussed in Section 2.1, the z-update step can be performed by computing the closed-form solution of the related subproblem. At the beginning of each iteration, before starting the BCD loop, a test is performed to ensure that the points of the generated sequence belong to a compact level set. This is done in order to guarantee that the sequence generated by the PD method is bounded, so that it admits limit points.

Convergence properties can be proved for the PD algorithm.

Proposition 3.4 (Lu and Zhang 2013). *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 1. Then*

- *for all $k = 0, 1, \dots,$ there exists ℓ such that $\|\nabla_x q_{\tau_k}(u^\ell, v^\ell)\| \leq \varepsilon_k$;*
- *the sequence $\{x^k, z^k\}$ admits accumulation points;*
- *each accumulation point $(\bar{x}, \bar{y})$ is such that:*
    - *$\bar{x} = \bar{y}$ and $\bar{x}$ is feasible for Problem* (1.1)*;*
    - *$\bar{x}$ satisfies LZ conditions for problem* (1.1)*.*

Compared to the other algorithms described so far, the PD approach has the advantage of being practically employable with additional constraints ($X \subset \mathbb{R}^n$), with stronger convergence properties than the regularization method (LZ conditions are stronger than M-stationarity).

Algorithm 1: `PenaltyDecomposition`

```
1  Input: $\tau_0 > 0, \theta > 1, x^0 = z^0 \in \mathbb{R}^n$ s.t. $\|x^0\|_0 \leq s$, a sequence $\{\varepsilon_k\}$ s.t. $\varepsilon_k \to 0$,
   $\Gamma \geq \max\{f(x^0), \min_x q_{\tau_0}(x, z^0)\}$.
2  for $k = 0, 1, \dots$ do
3      $\ell = 0$
4      $u^0 = x^k$
5      if $\min_x q_{\tau_k}(x, z^k) \leq \Gamma$ then
6          $v^0 = z^k$
7      else
8          $v^0 = z^0$
9      while $\|\nabla_x q_{\tau_k}(u^\ell, v^\ell)\| > \varepsilon_k$ do
10         $u^{\ell+1} \in \arg\min_u q_{\tau_k}(u, v^\ell)$
11         $v^{\ell+1} \in \arg\min_{v \in Z} q_{\tau_k}(u^{\ell+1}, v)$
12         $\ell = \ell + 1$
13     $\tau_{k+1} = \theta\tau_k$
14     $x^{k+1}, z^{k+1} = u^\ell, v^\ell$
15 Output: The sequence $\{x^k\}$
```

However, in the case $X = \mathbb{R}^n$, it has on the contrary weaker optimality guarantees than IHT or GSS; indeed LZ points are not even guaranteed to be BF solutions. Moreover, the practical performance of the PD scheme strongly depend, both in terms of efficiency and effectiveness, on the setting of the penalty parameters sequence $\{\tau^k\}$. Finally, the argmin operations required at step 10 implicitly require convexity assumptions on the problem.

We will address some of the shortcomings highlighted for this approach in the case $X = \mathbb{R}^n$ in the following Section.

Chapter 4

A Convergent Inexact Penalty Decomposition Method for Cardinality Constrained Optimization

The Penalty Decomposition algorithm has been shown to be effective in practice (Lu and Zhang, 2013). However, it requires to compute, in the inner iterations of the block decomposition method, the exact solution of a sequence of subproblems in the x variables (see steps 5 and 10 of Algorithm 1). This may be prohibitive when either the objective function is nonconvex or the finite termination of an algorithm applied to a convex subproblem cannot be guaranteed (this latter issue typically occurs when the convex function is not quadratic.).

On the other hand, the convergence analysis for the PD scheme is strongly based on the assumption that the global minima of the subproblems in the x variables are determined. In order to overcome this not trivial issue by preserving global convergence properties, we propose in this Chapter a modified version of the algorithm, suitable even for problems with nonconvex objective function in the case when $X = \mathbb{R}^n$.

4.1 An Inexact Penalty Decomposition Method

Throughout the Chapter, we assume that $X = \mathbb{R}^n$, i.e., here we are concerned with the optimization problem

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & \|x\|_0 \leq s. \end{aligned} \tag{4.1}$$

Moreover, we make the following hypothesis.

Assumption 4.1. The function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and coercive on $\mathbb{R}^n$, i.e., for all sequences $\{x^k\}$ such that $x^k \in \mathbb{R}^n$ and $\lim_{k\to\infty} \|x^k\| = \infty$ we have $\lim_{k\to\infty} f(x^k) = \infty$.

The above assumption implies that problem (4.1) admits solution.

Algorithm 2: `InexactPenaltyDecomposition`

```
Input: $\tau_0 > 0, \theta > 1, x^0 = z^0 \in \mathbb{R}^n$ s.t. $\|x^0\|_0 \leq s$, a sequence $\{\varepsilon_k\}$ s.t. $\varepsilon_k \to 0$,
    $\gamma \in (0,1), \beta \in (0,1)$.
for $k = 0, 1, \dots$ do
    $\ell = 0$
    $\alpha = \texttt{ArmijoLineSearch}(q_{\tau_k}, x^k, z^k, -\nabla_x q_{\tau_k}(x^k, z^k), \gamma, \beta)$
    $x_{\text{trial}} = x^k - \alpha \nabla_x q_{\tau_k}(x^k, z^k)$
    if $q_{\tau_k}(x_{trial}, z^k) \leq f(x^0)$ then
        $u^0, v^0 = x^k, z^k$
    else
        $u^0, v^0 = x^0, z^0$
    while $\|\nabla_x q_{\tau_k}(u^\ell, v^\ell)\| > \varepsilon_k$ do
        $\alpha_\ell = \texttt{ArmijoLineSearch}(q_{\tau_k}, u^\ell, v^\ell, -\nabla_x q_{\tau_k}(u^\ell, v^\ell), \gamma, \beta)$
        $u^{\ell+1} = u^\ell - \alpha_\ell \nabla_x q_{\tau_k}(u^\ell, v^\ell)$
        $v^{\ell+1} \in \arg\min_{v \in Z} q_{\tau_k}(u^{\ell+1}, v)$
        $\ell = \ell + 1$
    $\tau_{k+1} = \theta \tau_k$
    $x^{k+1} = u^\ell$
    $z^{k+1} = v^\ell$
Output: The sequence $\{x^k\}$
```

The proposed procedure is described in Algorithm 2. The exact minimization with respect to the x variables is replaced by an Armijo-type line search along the steepest descent direction of the penalty function, similarly as what is done in other decomposition schemes (Grippo and Sciandrone, 1999, 2000; Galvan et al., 2020). The line search procedure along a descent direction d is shown in Algorithm 3.

We recall some well-known properties for the Armijo-type line search, later used in the convergence analysis. These results can be found, for instance, in Bertsekas (1997) book.

It can be easily seen that the algorithm is well-defined, i.e., there exists a finite integer j such that β^j satisfies the acceptability condition (4.2). Moreover the following result holds.

```
Algorithm 3: ArmijoLineSearch
Input: $g : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, $x, z \in \mathbb{R}^n$, $d \in \mathbb{R}^n$, $\gamma \in (0,1)$, $\beta \in (0,1)$.
Compute
    $\alpha = \max_{j \in \mathbb{N}} \{\beta^j : g(x + \beta^j d, z) \leq g(x,z) + \gamma \beta^j \nabla_x g(x,z)^T d\}$    (4.2)
return $\alpha$
```

Proposition 4.1. *Let $g : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function and $\{x^t, z^t\} \subseteq \mathbb{R}^n \times \mathbb{R}^n$. Let $T \subseteq \{0, 1, \dots, \}$ be an infinite subset such that*

$$\lim_{\substack{t \to \infty \\ t \in T}} (x^t, z^t) = (\bar{x}, \bar{z}).$$

Let $\{d^t\}$ be a sequence of directions such that $\nabla_x g(x^t, z^t)^T d^t < 0$ and assume that $\|d^t\| \leq M$ for some $M > 0$ and for all $t \in T$. If

$$\lim_{\substack{t \to \infty \\ t \in T}} g(x^t, z^t) - g(x^t + \alpha_t d^t, z^t) = 0,$$

then we have

$$\lim_{\substack{t \to \infty \\ t \in T}} \nabla_x g(x^t, z^t)^T d^t = 0.$$

Remark 4.1. Step 12 of Algorithm 2 can be modified in order to make the algorithm more general. More specifically, the steepest descent direction $-\nabla_x q_{\tau_k}(u^\ell, v^\ell)$ could be replaced by any gradient-related direction d^ℓ. In this sense, we have the possibility of arbitrarily defining the updated point $u^{\ell+1}$, provided that $q_{\tau_k}(u^{\ell+1}, v^\ell) \leq q_{\tau_k}(u^\ell + \alpha_\ell d^\ell, v^\ell)$, where α_ℓ is computed by Armijo line search along the descent direction d^ℓ that, in particular, may be $-\nabla_x q_{\tau_k}(u^\ell, v^\ell)$. It can be easily seen that this modification does not spoil the theoretical analysis we are going to carry out hereafter, while it may bring significant benefits from a computational perspective.

Remark 4.2. As outlined by Lu and Zhang (2013), the stopping condition at line 10 of Algorithm 2 is useful for establishing the convergence properties of the algorithm, but, in practice, different rules could be employed with benefits in terms of efficiency. For example, the progress of the decreasing sequence $\{q_{\tau_k}(u^\ell, v^\ell)\}$ might be taken into account. As for the main loop, the whole algorithm can be stopped in practice as soon as x^k and z^k are sufficiently close.

In the following Section, we address the properties of the Inexact Penalty Decomposition Method.

4.2 Convergence Analysis

Let us introduce the level set

$$\mathcal{L}_0(f) = \{x : f(x) \leq f(x^0)\}.$$

Note that $\mathcal{L}_0(f)$ is compact, being f continuous and coercive on $\mathbb{R}^n$. First we show that also $q_\tau(x,z)$ is a coercive function.

Lemma 4.1. *The function $q_\tau(x,z)$ is coercive on $\mathbb{R}^n \times \mathbb{R}^n$.*

Proof. Let us consider any pair of sequences $\{x^k\}$ and $\{z^k\}$ such that at least one of the following conditions holds

$$\lim_{k\to\infty} \|x^k\| = \infty, \tag{4.3}$$

$$\lim_{k\to\infty} \|z^k\| = \infty. \tag{4.4}$$

Assume by contradiction that there exists an infinite subset $K \subseteq \{0,1,\dots,\}$ such that

$$\limsup_{\substack{k\to\infty \\ k\in K}} q_\tau(x^k,z^k) \neq \infty. \tag{4.5}$$

Suppose first that there exists an infinite subset $K_1 \subseteq K$ such that

$$\|x^k - z^k\| \leq M, \tag{4.6}$$

for some $M > 0$ and for all $k \in K_1$. Recalling that f is coercive on $\mathbb{R}^n$, from (4.3), (4.4) we have that $f(x^k) \to \infty$ for $k \to \infty, k \in K_1$. From (4.6) we obtain

$$\lim_{\substack{k\to\infty \\ k\in K_1}} q_\tau(x^k,z^k) = \lim_{\substack{k\to\infty \\ k\in K_1}} f(x^k) + \frac{\tau}{2}\|x^k - z^{k^2}\| = \infty,$$

and this contradicts (4.5). Then we must have

$$\lim_{\substack{k\to\infty \\ k\in K}} \|x^k - z^k\| = \infty.$$

As f is coercive and continuous, it admits minimum over $\mathbb{R}^n$. Let $f^\star$ the minimum value of f. Thus, we have

$$q_\tau(x^k,z^k) \geq f^\star + \frac{\tau}{2}\|x^k - z^k\|^2,$$

which implies that $q_\tau(x^k,z^k) \to \infty$ for $k \to \infty, k \in K$.

Then, we can conclude that, for any infinite set K, we have

$$\lim_{\substack{k\to\infty \\ k\in K}} q_\tau(x^k,z^k) = \infty,$$

and this contradicts (4.5). □

Now, we can prove that Algorithm 2 is well-defined, i.e., that the cycle between step 10 and step 14 terminates in a finite number of inner iterations.

Proposition 4.2. *Algorithm 2 cannot infinitely cycle between step 10 and step 14, i.e., for each outer iteration $k \geq 0$, the algorithm determines in a finite number of inner iterations a point (x^{k+1}, z^{k+1}) such that*

$$\|\nabla_x q_{\tau_k}(x^{k+1}, z^{k+1})\| \leq \epsilon. \tag{4.7}$$

Proof. Suppose by contradiction that, at a certain iteration k, the sequence $\{u^\ell, v^\ell\}$ is infinite. From the instructions of the algorithm, we have

$$q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq q_{\tau_k}(u^0, v^0).$$

Hence, for all $\ell \geq 0$, the point (u^ℓ, v^ℓ) belongs to the level set

$$\mathcal{L}_0(q_{\tau_k}) = \{(u, v) \in \mathbb{R}^n \times \mathbb{R}^n : \; q_{\tau_k}(u, v) \leq q_{\tau_k}(u^0, v^0)\}.$$

Lemma 4.1 implies that $\mathcal{L}_0(q_{\tau_k})$ is a compact set. Therefore, the sequence $\{u^\ell, v^\ell\}$ admits cluster points. Let $K \subseteq \{0, 1, \dots\}$ be an infinite subset such that

$$\lim_{\substack{\ell \to \infty \\ \ell \in K}} (u^\ell, v^\ell) = (\bar{u}, \bar{v}).$$

Recalling the continuity of the gradient, we have

$$\lim_{\substack{\ell \to \infty \\ \ell \in K}} \nabla_x q_{\tau_k}(u^\ell, v^\ell) = \nabla_x q_{\tau_k}(\bar{u}, \bar{v}).$$

We now show that $\nabla_x q_{\tau_k}(\bar{u}, \bar{v}) = 0$. Setting $d^\ell = -\nabla_x q_{\tau_k}(u^\ell, v^\ell)$ and taking into account the instructions of the algorithm we can write

$$q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq q_{\tau_k}(u^{\ell+1}, v^\ell) = q_{\tau_k}(u^\ell + \alpha_\ell d^\ell, v^\ell) < q_{\tau_k}(u^\ell, v^\ell). \tag{4.8}$$

Recalling again the continuity of the gradient, we have that $d^\ell \to \nabla_x q_{\tau_k}(\bar{u}, \bar{v})$ for $\ell \in K$ and $\ell \to \infty$, and hence $\|d^\ell\| \leq M$ for some $M > 0$ and for all $\ell \in K$.

The sequence $\{q_{\tau_k}(u^\ell, v^\ell)\}$ is monotonically decreasing, $q_{\tau_k}(u, v)$ is continuous, and hence we have that

$$\lim_{\ell \to \infty} q_{\tau_k}(u^\ell, v^\ell) = q_{\tau_k}(\bar{u}, \bar{v}).$$

From (4.8) it follows $\lim\limits_{\ell \to \infty} q_{\tau_k}(u^\ell, v^\ell) - q_{\tau_k}(u^\ell + \alpha_\ell d^\ell, v^\ell) = 0$. Then, the hypothesis of Proposition 4.1 are satisfied and we can write

$$\lim_{\substack{\ell \to \infty \\ \ell \in K}} \nabla_x q_{\tau_k}(u^\ell, v^\ell)^T d^\ell = \lim_{\substack{\ell \to \infty \\ \ell \in K}} -\|\nabla_x q_{\tau_k}(u^\ell, v^\ell)\|^2 = 0,$$

which implies that, for $\ell \in K$ sufficiently large, we have $\|\nabla_x q_{\tau_k}(u^\ell, v^\ell)\| \leq \epsilon$, i.e., that the stopping criterion of step 10 is satisfied in a finite number of iterations, and this contradicts the fact that $\{u^\ell, v^\ell\}$ is an infinite sequence. □

Before stating the global convergence result, we prove that the sequence generated by the algorithm admits limit points and that every limit point $(\bar{x}, \bar{z})$ is such that $\bar{x}$ is feasible for the original problem (4.1).

Proposition 4.3. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 2. Then $\{x^k, z^k\}$ admits cluster points and every cluster point $(\bar{x}, \bar{y})$ is such that $\bar{x} = \bar{z}$, and $\|\bar{x}\|_0 \leq s$.*

Proof. Consider a generic iteration k. The instructions of the algorithm imply for all $\ell \geq 0$

$$q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq q_{\tau_k}(u^{\ell+1}, v^{\ell}) = q_{\tau_k}(u^{\ell} - \alpha_\ell \nabla_x q_{\tau_k}(u^{\ell}, v^{\ell}), v^{\ell}) \leq q_{\tau_k}(u^{\ell}, v^{\ell}),$$

and hence we can write

$$q_{\tau_k}(x^{k+1}, z^{k+1}) \leq q_{\tau_k}(u^0 - \alpha_0 \nabla_x q_{\tau_k}(u^0, v^0), v^0). \tag{4.9}$$

From the definition of (u^0, v^0), we either have $(u^0, v^0) = (x^k, z^k)$ or $(u^0, v^0) = (x^0, z^0)$. In the former case we have, by the definition of x_{trial}, that

$$q_{\tau_k}(u^0 - \alpha_0 \nabla_x q_{\tau_k}(u^0, v^0), v^0) = q_{\tau_k}(x_{\text{trial}}, z^k) \leq f(x^0),$$

where the last inequality holds, as in this case the condition at line 6 is satisfied. In the latter case, we have

$$\begin{aligned} q_{\tau_k}(u^0 - \alpha_0 \nabla_x q_{\tau_k}(u^0, v^0), v^0) &\leq q_{\tau_k}(u^0, v^0) = q_{\tau_k}(x^0, z^0) \\ &= f(x^0) + \frac{\tau_k}{2}\|x^0 - z^0\|^2 = f(x^0). \end{aligned}$$

Then, in both cases from (4.9) it follows

$$q_{\tau_k}(x^{k+1}, z^{k+1}) \leq f(x^0). \tag{4.10}$$

We also have

$$f(x^{k+1}) \leq q_{\tau_k}(x^{k+1}, z^{k+1}) = f(x^{k+1}) + \frac{\tau_k}{2}\|x^{k+1} - z^{k+1}\|^2 \leq f(x^0), \tag{4.11}$$

and hence we can conclude that for all $k \geq 0$ we have $f(x^{k+1}) \leq f(x^0)$. Therefore, the points of the sequence $\{x^k\}$ belong to the compact set $\mathcal{L}_0(f)$, and this implies that $\{x^k\}$ is a bounded sequence and that, for all $k \geq 0$, $f(x^k) \geq f^\star > -\infty$, being $f^\star$ the minimum value of f over $\mathbb{R}^n$.

From (4.11), dividing by τ_k, we get

$$\|x^{k+1} - z^{k+1}\|^2 \leq 2\frac{f(x^0) - f(x^{k+1})}{\tau_k} \leq 2\frac{f(x^0) - f^\star}{\tau_k}.$$

Taking the limits for $k \to \infty$, recalling that $\tau_k \to \infty$ for $k \to \infty$, we obtain

$$\lim_{k\to\infty} \|x^{k+1} - z^{k+1}\| = 0. \tag{4.12}$$

Therefore, since $\{x^k\}$ is a bounded sequence, from (4.12), it follows that $\{z^k\}$ is bounded, and hence the sequence $\{(x^k, z^k)\}$ admits cluster points. Let $(\bar{x}, \bar{z})$ be any cluster point of $\{(x^k, z^k)\}$, i.e., there exists an infinite subset $K \subseteq \{0, 1, \dots\}$ such that

$$\lim_{\substack{k\to\infty \\ k\in K}} (x^k, z^k) = (\bar{x}, \bar{z}).$$

Again from (4.12) it follows $\bar{x} = \bar{z}$.

Finally, as $\|z^k\|_0 \le s$ for all k, recalling the lower semicontinuity of the ℓ_0-norm $\|\cdot\|_0$, we can conclude that $\|\bar{x}\|_0 = \|\bar{z}\|_0 \le s$. □

We are ready to state the global convergence result.

Theorem 4.1. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 2. Then $\{x^k, z^k\}$ admits cluster points and every cluster point $(\bar{x}, \bar{z})$ is such that $\bar{x}$ satisfies the Lu-Zhang conditions for problem* (4.1).

Proof. Proposition 4.3 implies that the sequence $\{x^k, z^k\}$ admits cluster points. Let $K \subseteq \{0, 1, \dots\}$ be an infinite subsequence such that

$$\lim_{\substack{k\to\infty \\ k\in K}} (x^{k+1}, z^{k+1}) = (\bar{x}, \bar{z}).$$

From Proposition 4.3, it follows $\bar{x} = \bar{z}$ and

$$\|\bar{x}\|_0 \le s. \tag{4.13}$$

Using (4.7) of Proposition 4.2, for all $k \ge 0$, we have

$$\|\nabla f(x^{k+1}) + \tau_k (x^{k+1} - z^{k+1})\| \le \varepsilon_k,$$

so that, taking the limits for $k \in K$ and $k \to \infty$, as $\varepsilon_k \to 0$, we can write

$$\lim_{\substack{k\to\infty \\ k\in K}} \|\nabla f(x^{k+1}) + \tau_k (x^{k+1} - z^{k+1})\| = 0. \tag{4.14}$$

From the instructions of the algorithm, we have $z^{k+1} \in \arg\min_{z\in Z} q_{\tau_k}(x^{k+1}, z)$, i.e., z^{k+1} is a solution of the problem

$$\min_z \; \|z - x^{k+1}\|^2 \quad \text{s.t.} \quad \|z\|_0 \le s.$$

From (2.2) it follows

$$z_i^{k+1} = x_i^{k+1} \quad \text{for } i \in \mathcal{G}(x^{k+1}), \qquad z_i^{k+1} = 0 \quad \text{for } i \notin \mathcal{G}(x^{k+1}),$$

where we recall that the index set $\mathcal{G}(x^{k+1})$ contains at most s elements, those corresponding to the not null components of x^{k+1} with the largest absolute value.

Note that $|\mathcal{G}(x^{k+1})| < s$ implies $\|x^{k+1}\|_0 < s$ and hence $z^{k+1} = x^{k+1}$. Therefore, we can write

$$-\tau_k(x_i^{k+1} - z_i^{k+1}) = 0 \begin{cases} \forall\, i \in \mathcal{G}(x^{k+1}), & \text{if } |\mathcal{G}(x^{k+1})| = s, \\ \forall\, i \in \{1,\dots,n\}, & \text{if } |\mathcal{G}(x^{k+1})| < s. \end{cases} \tag{4.15}$$

The index set $\mathcal{G}(x^{k+1})$ belongs to the finite set $\{1,\dots,n\}$, therefore there exists an infinite subset $K_1 \subseteq K$ such that $\mathcal{G}(x^{k+1}) = \mathcal{G}$ for all $k \in K_1$.

Let $\mathcal{G}^\star = \mathcal{G}(\bar{x}) = I_1(\bar{x})$, being $\bar{x}$ feasible. We show that $\mathcal{G}^\star \subseteq \mathcal{G}$. Indeed, assume by contradiction that there exists $i \in \mathcal{G}^\star$ such that $i \notin \mathcal{G}$. Hence, $\bar{y}_i = \bar{x}_i \neq 0$, while $z_i^{k+1} = 0$ for all $k \in K$. This is a contradiction, since $z^{k+1} \to \bar{z}$ for $k \to \infty, k \in K$.

Therefore, we have the following possible cases:

$$\text{(i) } |\mathcal{G}| = s,\ \mathcal{G} = \mathcal{G}^\star; \qquad \text{(ii) } |\mathcal{G}| < s; \qquad \text{(iii) } |\mathcal{G}| = s,\ \mathcal{G} \supset \mathcal{G}^\star.$$

We now prove each case separately:

(i) Let $i \in \mathcal{G} = \mathcal{G}^\star$; from (4.14) we have

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \nabla_i f(x^{k+1}) + \tau_k(x_i^{k+1} - z_i^{k+1}) = 0,$$

and, using the first condition of (4.15), it follows $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $k \in K_1$. Therefore, recalling the continuity of the gradient, we can write

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \nabla_i f(x^{k+1}) = \nabla_i f(\bar{x}) = 0 \quad \forall\, i \in \mathcal{G}^\star,$$

i.e., Lu-Zhang conditions hold with the (super) support set $\mathcal{G} = \mathcal{G}^\star$.

(ii) Let $i \in \{1,\dots,n\}$; similarly to the previous case, we have that

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \nabla_i f(x^{k+1}) + \tau_k(x_i^{k+1} - z_i^{k+1}) = 0,$$

and using the second condition of (4.15) it follows $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $k \in K_1$. Therefore, we obtain

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \nabla_i f(x^{k+1}) = \nabla_i f(\bar{x}) = 0 \quad \forall\, i \in \{1,\dots,n\},$$

i.e., Lu-Zhang conditions hold taking any super support set.

(iii) Let $i \in \mathcal{G}$. By the same reasonings of case (i), we can write

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \nabla_i f(x^{k+1}) = \nabla_i f(\bar{x}) = 0 \quad \forall\, i \in \mathcal{G},$$

i.e., Lu-Zhang conditions hold with the super support set $\mathcal{G} \supseteq I_1(\bar{x})$.

Putting everything together, we have from (i), (ii) and (iii) that Lu-Zhang conditions are always satisfied. □

As we can see, the proposed inexact version of the algorithm enjoys the same convergence properties as the original, exact one described in Section 3.4. In the following remark, we provide a better characterization of the algorithm, with an ex-post result that shows that the limit points are often BF-vectors (equivalently, satisfy strong LZ conditions).

Remark 4.3. We note that, in both case (i) and case (ii) we have that $\bar{x}$ satisfies the BF optimality conditions. Moreover, note also that:

- If there exists a subsequence $\hat{K} \subseteq K$ s.t. $\|x^k\|_0 = \|\bar{x}\|_0$ for all $k \in \hat{K}$, the only possible cases are case (i) and (ii). Indeed, let us consider a further subsequence $K_2 \subseteq \hat{K}$, such that $\mathcal{G}(x^{k+1}) = \mathcal{G}$ for every $k \in K_2$, for some $\mathcal{G} \subset \{1, \dots, n\}$. We know that K_2 exists and that $\mathcal{G} \supseteq \mathcal{G}^\star$. Since $\|x^{k+1}\|_0 = \|\bar{x}\|_0 \le s$ for every $k \in K_2$, $\mathcal{G} = I_1(x^{k+1})$ and $\mathcal{G}^\star = I_1(\bar{x})$ respectively, and they have the same cardinality. Therefore, it cannot be $\mathcal{G} \supset \mathcal{G}^\star$. It follows that $\mathcal{G} = \mathcal{G}^\star$, so we fall into either case (i) or case (ii), and thus $\bar{x}$ satisfies BF conditions.

- If there exists a subsequence $\hat{K} \subseteq K$ such that $\|x^{k+1}\|_0 < s$ for all $k \in \hat{K}$, we can again define $K_2 \subseteq \hat{K}$ such that $\mathcal{G}(x^{k+1}) = \mathcal{G}$ for every $k \in K_2$, for some $\mathcal{G} \subset \{1, \dots, n\}$. In this case, we have $|\mathcal{G}| = \|x^{k+1}\|_0 < s$ and case (ii) applies. It follows that $\bar{x}$ is a BF-vector.

In case (i) from the proof, the algorithm is substantially imposing optimality w.r.t. the support, which is the only super support set. In case (ii), the algorithm looks at all possible super support sets; in case (iii), instead, the algorithm only considers one super support set among many.

Basically, the unfortunate case happens when the support of the solution asymptotically becomes incomplete. In this case, the algorithm somewhat enforces optimality only w.r.t. variables in the support of iterates $\{z^k\}$, ignoring some super support sets that should be considered at the limit point.

4.3 Future Work

Further work shall regard the extension of the presented algorithm to the case of problem (1.1) when $X \subset \mathbb{R}^n$, which, similarly to what is done in the exact PD method, might be handled by moving the additional constraints into the quadratic penalty term.

Another interesting theoretical investigation might concern the substitution of the line search step by a trust-region framework. Such a modification, which appears to be reasonable, would in fact require nontrivial changes to the convergence analysis.

Chapter 5

A Derivative-Free Penalty Decomposition Algorithm for Black-Box Sparse Optimization

First-order information about the objective function is fundamental for the PD class of methods. However, there are applications where the objective function is obtained by direct measurements or it is the result of a complex system of calculations, so that its analytical expression is not available and the computation of its values may be affected by the presence of noise. Hence, in these cases the gradient cannot be explicitly calculated or approximated.

Such lack of information has an impact on the applicability of Algorithm 2. In particular, the x update step and the inner loop stopping criterion are no more employable as they are.

In this Chapter, we provide the definition of a derivative-free PD method for sparse black-box optimization. We remark that, to our knowledge, convergent derivative-free methods for cardinality constrained problems are not known, and this makes the derivative-free algorithm proposed here particularly attractive.

5.1 A Derivative-Free Penalty Decomposition Method

Similarly as in Chapter 4, we consider the problem without additional constraints

$$\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & \|x\|_0 \leq s, \end{array} \tag{5.1}$$

and we also make the coercivity assumption on the objective function.

Assumption 5.1. The function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and coercive on $\mathbb{R}^n$, i.e., for all sequences $\{x^k\}$ such that $x^k \in \mathbb{R}^n$ and $\lim_{k\to\infty} \|x^k\| = \infty$ we have $\lim_{k\to\infty} f(x^k) = \infty$.

The derivative-free PD method is described by Algorithm 5. At the x update step, we employ as search directions the coordinate directions and their opposites. A tentative step length $\tilde{\alpha}_i$ is associated with each of these directions. At every iteration, all search directions are considered one at a time; a derivative-free line search is performed along each direction, according to Algorithm 4.

If the tentative step size does not provide a sufficient decrease, it will be reduced for the next iteration. If, on the other hand, the tentative step size is of sufficient decrease, an extrapolation procedure is carried out; the tentative step size for that same direction at the successive iteration will be the longest one tried in the extrapolation phase that provides a sufficient decrease.

That same step length is also used to move along the considered direction, provided it is large at least ε_k; otherwise, no movement is done along the direction. The inner loop then stops when all tentative step sizes have become smaller than ε_k.

Algorithm 4: `LineSearch`

```
1  Input: $f : \mathbb{R}^n \to \mathbb{R}$, $d \in \mathbb{R}^n$, $\alpha_0 \in \mathbb{R}^+$, $x \in \mathbb{R}^n$, $\gamma \in (0,1)$, $\sigma > 1$.
2  $\alpha = \alpha_0$
3  if $f(x + \alpha d) \leq f(x) - \gamma\alpha^2\|d\|^2$ then
4      Let $\beta = \alpha$
5      repeat
6          Set $\alpha = \beta$
7          Set $\beta = \sigma\alpha$
8      until $f(x + \beta d) > f(x) - \gamma\beta^2\|d\|^2$;
9      return $\alpha$
10 Set $\alpha = 0$
11 return $\alpha$
```

5.2 Convergence Analysis

Hereafter, we show that Algorithm 5 enjoys the same convergence properties as Algorithm 2 and hence of the original PD Algorithm 1.

First, we prove that the line search procedure does not loop infinitely inside our procedure.

Proposition 5.1. *Algorithm 4 cannot infinitely cycle between steps 5 and 8.*

Algorithm 5: `DerivativeFreeInexactPenaltyDecomposition`

```
Input: $\tau_0 > 0, \theta > 1, \delta \in (0,1), \gamma \in (0,1), \sigma > 1, x^0 = z^0 \in \mathbb{R}^n$ s.t. $\|x^0\|_0 \leq s$,
  $\{\varepsilon_k\}$ s.t. $\varepsilon_k < 1$ for all $k$ and $\varepsilon_k \to 0$,
  $\mathcal{D} = \{d_1, \dots, d_{2n}\} = \{e_1, \dots, e_n, -e_1, \dots, -e_n\}$.
for $k = 0, 1, \dots$ do
    $\tilde{\alpha}^0 = e \in \mathbb{R}^{2n}$
    $\ell = 0$
    $x_{\text{trial}} = x^k$
    for $i = 1, \dots, 2n$ do
        $\hat{\alpha}_i = \texttt{LineSearch}(q_{\tau_k}(x, z^k), d_i, 1, x^k, \gamma, \sigma)$
        if $\hat{\alpha}_i > \varepsilon_k$ then
            $x_{\text{trial}} = x^k + \hat{\alpha}_i d_i$
            break
    if $q_{\tau_k}(x_{trial}, z^k) \leq f(x^0)$ then
        $u^0, v^0 = x^k, z^k$
    else
        $u^0, v^0 = x^0, z^0$
    while $\max_{i=1,\dots,2n} \{\tilde{\alpha}_i^\ell\} > \varepsilon_k$ do
        $u^\ell(0) = u^\ell$
        for $i = 1, \dots, 2n$ do
            $\alpha_i^\ell = \texttt{LineSearch}(q_{\tau_k}(u, v^\ell), d_i, \tilde{\alpha}_i^\ell, u^\ell(i-1), \gamma, \sigma)$
            if $\alpha_i^\ell = 0$ then
                $\tilde{\alpha}_i^{\ell+1} = \delta \tilde{\alpha}_i^\ell$
            else
                $\tilde{\alpha}_i^{\ell+1} = \alpha_i^\ell$
            if $\alpha_i^\ell > \varepsilon_k$ then
                $u^\ell(i) = u^\ell(i-1) + \alpha_i^\ell d_i^\ell$
            else
                $u^\ell(i) = u^\ell(i-1)$
        $u^{\ell+1} = u^\ell(2n)$
        $v^{\ell+1} \in \arg\min_{v \in Z} q_{\tau_k}(u^{\ell+1}, v)$
        $\ell = \ell + 1$
    $\tau_{k+1} = \theta \tau_k$
    $x^{k+1} = u^\ell$
    $z^{k+1} = v^\ell$
Output: The sequence $\{x^k\}$
```

Proof. Assume by contradiction that Algorithm 4 does not terminate. Then, for $j = 0, 1, \ldots$, we have $f(x + \sigma^j \alpha_0 d) \leq f(x) - \gamma \sigma^{2j} \alpha_0^2 \|d\|^2$. Taking the limits for $j \to \infty$, we obtain that $f(x + \sigma^j \alpha_0 d) \to -\infty$, and this contradicts the fact that f is bounded below, being continuous and coercive. □

Note that, as shown by Proposition 5.1, q_{τ_k} is coercive on $\mathbb{R}^n \times \mathbb{R}^n$. We prove that Algorithm 5 is well-defined, i.e., the inner loop terminates in a finite number of iterations.

Proposition 5.2. *Algorithm 5 cannot infinitely cycle between steps 15 and 29.*

Proof. Assume by contradiction that the algorithm loops infinitely. Then, for every $\ell = 0, 1, \ldots$, there exists $i \in \{1, \ldots, 2n\}$ such that $\tilde{\alpha}_i^\ell > \varepsilon_k$, i.e.,

$$\max_{i=1,\ldots,2n} \{\tilde{\alpha}_i^\ell\} > \varepsilon_k. \tag{5.2}$$

The instructions of the algorithm imply

$$q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq q_{\tau_k}(u^{\ell+1}, v^\ell) \leq q_{\tau_k}(u^\ell(i), v^\ell) \leq q_{\tau_k}(u^\ell(i-1), v^\ell) \leq q_{\tau_k}(u^\ell, v^\ell).$$

Then, the decreasing sequence $\{q_{\tau_k}(u^\ell, v^\ell)\}$ tends to a finite value, being q_{τ_k} continuous and coercive and hence bounded below. For any $i \in \{1, \ldots, 2n\}$, we can split the sequence of iterations $\{0, 1, \ldots\}$ into two subsequences K_1 and K_2 such that $K_1 \cup K_2 = \{0, 1, \ldots\}$, $K_1 \cap K_2 = \emptyset$. In particular, we denote by:

- K_1 the set of iterations where $\tilde{\alpha}_i^{\ell+1} = \alpha_i^\ell = \tilde{\alpha}_i^\ell \sigma^t > 0$ for some $t \geq 0$, $t \in \mathbb{N}$;
- K_2 the set of iterations where $\tilde{\alpha}_i^{\ell+1} = \delta \tilde{\alpha}_i^\ell$ and $\alpha_i^\ell = 0$.

Note that K_1 and K_2 cannot both be finite. Then we analyze the following two cases, K_1 infinite (Case I) and K_2 infinite (Case II).
Case (I). We have

$$\begin{aligned} q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) &\leq q_{\tau_k}(u^{\ell+1}, v^\ell) \leq q_{\tau_k}(u^\ell(i), v^\ell) \leq q_{\tau_k}(u^\ell(i-1), v^\ell) - \gamma(\tilde{\alpha}_i^\ell \sigma^t)^2 \\ &\leq q_{\tau_k}(u^\ell(0), v^\ell) - \gamma(\tilde{\alpha}_i^\ell)^2 = q_{\tau_k}(u^\ell, v^\ell) - \gamma(\tilde{\alpha}_i^\ell)^2. \end{aligned}$$

Taking the limits for $\ell \in K_1$, $\ell \to \infty$, recalling that $\{q_{\tau_k}(u^\ell, v^\ell)\}$ tends to a finite limit, we get

$$\lim_{\substack{\ell \to \infty \\ \ell \in K_1}} \tilde{\alpha}_i^\ell = 0, \tag{5.3}$$

and hence, for $\ell \in K_1$ sufficiently large, we have $\tilde{\alpha}_i^\ell \leq \varepsilon_k$.
Case (II). For every $\ell \in K_2$, let m_ℓ be the maximum index on $\{0, 1, \ldots\}$ such that $m_\ell \in K_1$, $m_\ell < \ell$ (m_ℓ is the index of the last iteration in K_1 preceding ℓ). We can

assume $m_\ell = 0$ if the index m_ℓ does not exist, that is, K_1 is empty. Then we can write $\tilde{\alpha}_i^\ell = \delta^{\ell - m_\ell} \alpha_i^{m_\ell}$. As $\ell \in K_2$ and $\ell \to \infty$, either $m_\ell \to \infty$ (if K_1 is an infinite subset) or $\ell - m_\ell \to \infty$ (if K_1 is finite). Therefore, (5.3) and the fact that $\delta \in (0,1)$ imply

$$\lim_{\substack{\ell \to \infty \\ \ell \in K_2}} \tilde{\alpha}_i^\ell = 0.$$

Thus, for $\ell \in K_2$ sufficiently large, we have $\tilde{\alpha}_i^\ell \leq \varepsilon_k$.

We can conclude that $\lim_{\ell \to \infty} \tilde{\alpha}_i^\ell = 0$, so that, recalling that i is arbitrary, we get $\max_{i=1,\dots,n}\{\tilde{\alpha}_i^\ell\} \leq \varepsilon_k$ for ℓ sufficiently large, and this contradicts (5.2). □

Next, we prove a technical result used later.

Proposition 5.3. *Assume that the initial step sizes $\tilde{\alpha}_i^0$, with $i = 1, \dots, n$, are such that $\tilde{\alpha}_i^0 > \varepsilon_k$ for all k. Then, for every k and for every $i = 1, \dots, 2n$, there exists $\rho_i^k \in (0, c\varepsilon_k)$ such that*

$$\nabla_x q_{\tau_k}(x^{k+1} + \rho_i^k d_i, y^{k+1})^T d_i > -c\varepsilon_k,$$

with $c = \max\{\sigma, 1/\delta\}$.

Proof. Given any iteration k, let ℓ be the index of the last inner iteration. By definition of ℓ, we must have that $\tilde{\alpha}_i^{\ell+1} \leq \varepsilon_k$ for all $i = 1, \dots, n$. From the instructions of the algorithm this implies that we have $u^{\ell+1} = u^\ell(2n) = \dots = u^\ell(0) = u^\ell$, and consequently $v^{\ell+1} = v^\ell$. Consider any $i \in \{1, \dots, 2n\}$. We have two cases:

1. $\tilde{\alpha}_i^{\ell+1} = \delta \tilde{\alpha}_i^\ell$; in this case, $\tilde{\alpha}_i^\ell$ did not satisfy the sufficient decrease condition in the `LineSearch` procedure, i.e.

$$q_{\tau_k}(u^\ell + \tilde{\alpha}_i^\ell d_i, v^\ell) - q_{\tau_k}(u^\ell, v^\ell) > -\gamma(\tilde{\alpha}_i^\ell)^2. \tag{5.4}$$

Using the Mean Value Theorem we can write

$$q_{\tau_k}(u^\ell + \tilde{\alpha}_i^\ell d_i, v^\ell) - q_{\tau_k}(u^\ell, v^\ell) = \tilde{\alpha}_i^\ell \nabla_x q_{\tau_k}(u^\ell + \rho_i^\ell d_i, v^\ell)^T d_i, \tag{5.5}$$

where $\rho_i^\ell \in (0, \tilde{\alpha}_i^\ell)$. From (5.4) and (5.5), it follows:

$$\nabla_x q_{\tau_k}(u^\ell + \rho_i^\ell d_i, v^\ell)^T d_i > -\gamma \tilde{\alpha}_i^\ell = -\frac{\gamma}{\delta} \tilde{\alpha}_i^{\ell+1} \geq -\frac{\gamma}{\delta} \varepsilon_k.$$

Observe that $\tilde{\alpha}_i^\ell \leq \varepsilon_k/\delta$ and hence $\rho_i^\ell \in (0, \varepsilon_k/\delta)$.

2. $\tilde{\alpha}_i^{\ell+1} = \alpha_i^\ell$; from the instructions of the `LineSearch` procedure, we get

$$q_{\tau_k}(u^\ell + \sigma \alpha_i^\ell d_i, v^\ell) - q_{\tau_k}(u^\ell, v^\ell) > -\gamma(\sigma \alpha_i^\ell)^2. \tag{5.6}$$

Using the Mean Value Theorem, we can write

$$q_{\tau_k}(u^\ell + \sigma \alpha_i^\ell d_i, v^\ell) - q_{\tau_k}(u^\ell, v^\ell) = \sigma \alpha_i^\ell \nabla_x q_{\tau_k}(u^\ell + \rho_i^\ell d_i, v^\ell)^T d_i, \tag{5.7}$$

where $\rho_i^\ell \in (0, \sigma\alpha_i^\ell)$. From (5.6) and (5.7), it follows

$$\nabla_x q_{\tau_k}(u^\ell + \rho_i^\ell d_i, v^\ell)^T d_i > -\gamma\sigma\alpha_i^\ell = -\gamma\sigma\tilde{\alpha}_i^{\ell+1} \geq -\gamma\sigma\varepsilon_k.$$

Observe that $\sigma\alpha_i^\ell = \sigma\tilde{\alpha}_i^{\ell+1} \leq \sigma\varepsilon_k$ and hence $\rho_i^\ell \in (0, \sigma\varepsilon_k)$.

Thus, in both cases we can write

$$\nabla_x q_{\tau_k}(u^\ell + \rho_i^\ell d_i, v^\ell)^T d_i > -c\varepsilon_k, \tag{5.8}$$

for some $\rho_i^\ell \in (0, c\varepsilon_k)$ and $c = \max\{\sigma, 1/\delta\}$.

Since $\tilde{\alpha}_i^{\ell+1} \leq \varepsilon_k$ for all $i = 1, \dots, 2n$, from the instructions of the algorithm, we have $u^{\ell+1} = u^\ell$ and consequently $v^{\ell+1} = v^\ell$. Hence, equation (5.8) holds with $u^\ell = x^{k+1}$, and $v^\ell = z^{k+1}$. □

Now, we prove that the sequence generated by the algorithm admits limit points and that every limit point is feasible for the original problem.

Proposition 5.4. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 5. Then, $\{x^k, z^k\}$ admits cluster points and every cluster point $(\bar{x}, \bar{z})$ is such that $\bar{x} = \bar{z}$, and $\|\bar{x}\|_0 \leq s$.*

Proof. Consider a generic iteration k. The instructions of the algorithm imply, for all $\ell \geq 0$,

$$q_{\tau_k}(x^{k+1}, z^{k+1}) = q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq q_{\tau_k}(u^{\ell+1}, v^\ell) \leq q_{\tau_k}(u^\ell, v^\ell).$$

From the definition of (u^0, v^0), we either have $(u^0, v^0) = (x^k, z^k)$ or $(u^0, v^0) = (x^0, z^0)$. In the former case, for some $i \in \{1, \dots, 2n\}$ we have, by the definition of x_{trial}, that

$$q_{\tau_k}(u^1, v^0) \leq q_{\tau_k}(u^0 + \hat{\alpha}_i d_i, v^0) = q_{\tau_k}(x_{\text{trial}}, z^k) \leq f(x^0).$$

In the latter case, we have

$$q_{\tau_k}(u^0, v^0) = q_{\tau_k}(x^0, z^0) = f(x^0) + \frac{\tau_k}{2}\|x^0 - z^0\|^2 = f(x^0).$$

Then, in both cases it follows

$$q_{\tau_k}(x^{k+1}, z^{k+1}) \leq f(x^0). \tag{5.9}$$

The rest of the proof follows the same reasonings used in the proof of Proposition 4.3, starting from the condition corresponding to (5.9), i.e., condition (4.10). □

Theorem 5.1. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 5. Then $\{x^k, z^k\}$ admits cluster points and every cluster point $(\bar{x}, \bar{z})$ is such that $\bar{x}$ satisfies the Lu-Zhang conditions for problem* (5.1).

Proof. Proposition 5.4 implies that the sequence $\{x^k, z^k\}$ admits cluster points. Let $K \subseteq \{0, 1, \dots\}$ be an infinite subsequence such that

$$\lim_{\substack{k\to\infty \\ k\in K}} (x^{k+1}, z^{k+1}) = (\bar{x}, \bar{z}).$$

From Proposition 5.4 it follows $\bar{x} = \bar{z}$ and $\|\bar{x}\|_0 \leq s$. From the instructions of the algorithm, we have $z^{k+1} \in \arg\min_{z\in Z} q_{\tau_k}(x^{k+1}, z)$, i.e., z^{k+1} is solution of the problem

$$\min_z \ \|z - x^{k+1}\|^2 \quad \text{s.t.} \quad \|z\|_0 \leq s.$$

From (2.2) it follows

$$z_i^{k+1} = x_i^{k+1} \quad \text{for } i \in \mathcal{G}(x^{k+1}), \qquad z_i^{k+1} = 0 \quad \text{for } i \notin \mathcal{G}(x^{k+1}),$$

where we recall that the index set $\mathcal{G}(x^{k+1})$ contains at most s elements, those corresponding to the not null components of x^{k+1} with the largest absolute value.

Note that $|\mathcal{G}(x^{k+1})| < s$ implies $\|x^{k+1}\|_0 < s$ and hence $z^{k+1} = x^{k+1}$. Therefore, we can write

$$-\tau_k(x_i^{k+1} - z_i^{k+1}) = 0 \begin{cases} \forall\, i \in \mathcal{G}(x^{k+1}), & \text{if } |\mathcal{G}(x^{k+1})| = s, \\ \forall\, i \in \{1, \dots, n\}, & \text{if } |\mathcal{G}(x^{k+1})| < s. \end{cases} \tag{5.10}$$

The index set $\mathcal{G}(x^{k+1})$ belongs to the finite set $\{1, \dots, n\}$, therefore there exists an infinite subset $K_1 \subseteq K$ such that $\mathcal{G}(x^{k+1}) = \mathcal{G}$ for all $k \in K_1$.

Let $\mathcal{G}^\star = \mathcal{G}(\bar{x}) = I_1(\bar{x})$, being $\bar{x}$ feasible. We have already shown in the proof of Theorem 4.1 that $\mathcal{G}^\star \subseteq \mathcal{G}$. We consider the following possible cases:

$$\text{(i) } |\mathcal{G}| = s,\ \mathcal{G} = \mathcal{G}^\star; \qquad \text{(ii) } |\mathcal{G}| < s; \qquad \text{(iii) } |\mathcal{G}| = s,\ \mathcal{G} \supset \mathcal{G}^\star.$$

We now prove each case separately:

(i) Let $i \in \mathcal{G} = \mathcal{G}^\star$; using the first condition of (5.10), we get $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $k \in K_1$. From Proposition 5.3, recalling that

$$\mathcal{D} = \{d_1, \dots, d_{2n}\} = \{e_1, \dots e_n, -e_1, \dots, -e_n\},$$

we have that

$$\nabla f(x^{k+1} + \rho_i^k e_i)^T e_i = \nabla_x q_{\tau_k}(x^{k+1} + \rho_i^k e_i, z^{k+1})^T e_i > -c\varepsilon_k,$$
$$-\nabla f(x^{k+1} + \rho_{i+n}^k e_i)^T e_i = -\nabla_x q_{\tau_k}(x^{k+1} - \rho_{i+n}^k e_i, z^{k+1})^T e_i > -c\varepsilon_k,$$

with $c = \max\{\sigma, 1/\delta\}$. Taking limits for $k \to \infty, k \in K_1$, recalling that $\varepsilon_k \to 0$, $\rho_i^k, \rho_{i+n}^k \in (0, c\varepsilon_k)$ and the continuity of the gradient, we get

$$\lim_{k\in K_1, k\to\infty} \nabla f(x^{k+1} + \rho_i^k e_i)^T e_i = \nabla_i f(\bar{x}) \geq 0,$$
$$\lim_{k\in K_1, k\to\infty} -\nabla f(x^{k+1} - \rho_{i+n}^k e_i)^T e_i = -\nabla_i f(\bar{x}) \geq 0,$$

from which it follows that $\nabla_i f(\bar{x}) = 0$ for all $i \in \mathcal{G}^\star$, i.e., Lu-Zhang conditions hold with the (super) support set $\mathcal{G} = \mathcal{G}^\star$.

(ii) Let $i \in \{1,\dots,n\}$; the second condition of (5.10) implies $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $k \in K_1$. Similarly to the previous case, we can write

$$\begin{aligned}\nabla f(x^{k+1} + \rho_i^k e_i)^T e_i &= \nabla_x q_{\tau_k}(x^{k+1} + \rho_i^k e_i, z^{k+1})^T e_i > -c\varepsilon_k,\\ -\nabla f(x^{k+1} + \rho_{i+n}^k e_i)^T e_i &= -\nabla_x q_{\tau_k}(x^{k+1} - \rho_{i+n}^k e_i, z^{k+1})^T e_i > -c\varepsilon_k,\end{aligned}$$

with $c = \max\{\sigma, 1/\delta\}$, and we can prove

$$\lim_{\substack{k\to\infty\\ k\in K_1}} \nabla_i f(x^{k+1}) = \nabla_i f(\bar{x}) = 0 \quad \forall\, i \in \{1,\dots,n\},$$

i.e., Lu-Zhang conditions hold taking any super support set.

(iii) Let $i \in \mathcal{G}$. By the same reasonings of case (i), we can write

$$\lim_{\substack{k\to\infty\\ k\in K_1}} \nabla_i f(x^{k+1}) = \nabla_i f(\bar{x}) = 0 \quad \forall\, i \in \mathcal{G},$$

i.e., Lu-Zhang conditions hold with the super support set $\mathcal{G}$.

Putting everything together, we have, from (i), (ii) and (iii), that Lu-Zhang conditions are always satisfied. □

Remark 5.1. As in Remark 4.3, if there exists a subsequence $\hat{K} \subset K$ s.t. $\|x^k\|_0 = \|\bar{x}\|_0$ for all $k \in \hat{K}$ or $\|x^k\|_0 < s$ for all $k \in \hat{K}$, $\bar{x}$ is a BF-vector.

Chapter 6

A General Algorithm for Sparsity-Constrained Optimization Problems based on Discrete Neighborhoods

In this Chapter, we discuss an algorithmic framework for the solution of sparsity constrained problems

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & \|x\|_0 \leq s, \\ & x \in X, \end{aligned} \tag{6.1}$$

that exploits the reformulation given by problem

$$\begin{aligned} \min_{x,y} \; & f(x) \\ \text{s.t.} \; & e^\top y \geq n - s, \\ & x_i y_i = 0, \quad \forall\, i = 1, \dots, n, \\ & x \in X, \\ & y \in \{0,1\}^n. \end{aligned} \tag{6.2}$$

In particular, the approach aims at finding points satisfying the $\mathcal{N}$-stationarity condition newly defined in Chapter 2 (Definition 2.16). The algorithm combines inexact minimizations with a strategy that explores discrete neighborhoods of a given feasible point. Those features make it easy to handle the nonconvexity in both the objective function and the feasible set also from a practical point of view. We prove the convergence of the algorithmic scheme, establishing that its limit points are $\mathcal{N}$-stationary. We then show that most of the conditions reviewed in Chapter 2 can be easily guaranteed.

6.1 The Algorithm

The proposed approach tackles the mixed integer reformulation (6.2) and is somehow related to classical methods for mixed variable programming proposed in the literature (Li and Sun, 2006; Lucidi et al., 2005).

The core piece of the proposed method lies in the exploration of discrete neighborhoods. It is thus useful recalling the corresponding definition.

Definition 6.1. Let $(\bar{x},\bar{y}) \in \mathcal{X}(\bar{y}) \times \mathcal{Y}$ a feasible point for problem (6.2). A *discrete neighborhood* $\mathcal{N}(\bar{x},\bar{y})$ is a set of points such that:

- $(\bar{x},\bar{y}) \in \mathcal{N}(\bar{x},\bar{y})$;
- $(\hat{x},\hat{y}) \in \mathcal{X}(\hat{y}) \times \mathcal{Y}$ for all $(\hat{x},\hat{y}) \in \mathcal{N}(\bar{x},\bar{y})$;
- $|\mathcal{N}(\bar{x},\bar{y})| < \infty$.

Roughly speaking, the whole approach is based at each iteration on the computation of a discrete neighborhood $\mathcal{N}(x^k,y^k)$ of the current point (x^k,y^k) and on local exploratory moves with respect to the continuous variables around the points of the neighborhood.

Specifically, the continuous exploration move consists of a local search performed by an Armijo-type line search along the projected gradient direction, where the feasible set $\mathcal{X}(y)$ for the continuous variables is induced by the binary variables y that implicitly define an active set. The procedure is formalized in Algorithm 6.

Algorithm 6: `Projected-Gradient Line Search (PGLS)`

```
Input: $y \in \mathcal{Y}, x \in \mathcal{X}(y), \gamma \in (0,\frac{1}{2}), \delta \in (0,1), \alpha = 1$.
Set $\hat{x} = \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]$
Set $d = \hat{x} - x$
while $f(x+\alpha d) > f(x) + \gamma\alpha\nabla f(x)^\top d$ do
    set $\alpha = \delta\alpha$
Set $\tilde{x} = x + \alpha d$;
return $\tilde{x}$
```

In brief, the instructions of the algorithm are carried out as follows:

(i) starting from the current iterate (x^k,y^k), the `PGLS` is performed to obtain the point $\tilde{x}^k$;

(ii) for any point $(\hat{x}^k,\hat{y}^k) \in \mathcal{N}(\tilde{x}^k,y^k)$ that is not significantly worse (in terms of the objective value) than the current candidate, we perform a local continuous search around $\hat{x}^k$;

(iii) the local search can be constituted by several steps of `PGLS`;

(iv) we skip to the following iteration as soon as a point providing a sufficient decrease of the objective value is found (successful iteration) or when no point is left in the neighborhood to be explored;

(v) in the latter case, the success of the iteration will be established by the decrease in the objective value attained by $\tilde{x}$.

The algorithm, which we refer to as Sparse Neighborhood Search (SNS) is formally defined in Algorithm 7.

6.2 Neighborhood Continuity

In the following sections, we will prove a set of results concerning the properties of the sequences produced by Algorithm 7. Note that, unless stated otherwise, we employ the classical concept of stationarity (A.2) for convex optimization, based on the projection operator. First, however, we need to state some suitable assumptions.

Assumption 6.1. The gradient $\nabla f(x)$ is Lipschitz-continuous, i.e., there exists a constant $L > 0$ such that

$$\|\nabla f(x) - \nabla f(\bar{x})\| \leq L \|x - \bar{x}\|$$

for all $x, \bar{x} \in \mathbb{R}^n$.

Assumption 6.2. Given $y^0 \in \mathcal{Y}$, $x^0 \in \mathcal{X}(y^0)$ and a scalar $\xi > 0$, the level set

$$\mathcal{L}(x^0, y^0) = \{(x, y) \in \mathcal{X}(y) \times \mathcal{Y} \mid f(x) \leq f(x^0) + \xi\}$$

is compact.

The crucial point in the proposed framework is choosing suitable discrete neighborhoods. First, note that when we deal with both continuous and integer variables, the usual notion of convergence to a point needs to be tweaked. In particular, we have the following definition.

Definition 6.2. A sequence $\{(x^k, y^k)\}$ converges to a point $(\bar{x}, \bar{y})$ if for any $\epsilon > 0$ there exists an index k_ϵ such that for all $k \geq k_\epsilon$ we have that $y^k = \bar{y}$ and $\|x^k - \bar{x}\| < \epsilon$.

To ensure convergence to meaningful points, we need a "continuity" assumption on the discrete neighborhoods we explore.

Assumption 6.3. Let $\{(x^k, y^k)\}$ be a sequence converging to $(\bar{x}, \bar{y})$ and let $\mathcal{N}$ be a discrete neighborhood. Then, for any $(\hat{x}, \hat{y}) \in \mathcal{N}(\bar{x}, \bar{y})$, there exists a sequence $\{(\hat{x}^k, \hat{y}^k)\}$ converging to $(\hat{x}, \hat{y})$ such that $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}(x^k, y^k)$ for all k.

Algorithm 7: Sparse Neighborhood Search (SNS)

```
Input: $y^0 \in \mathcal{Y}, x^0 \in \mathcal{X}(y^0), \xi \geq 0, \theta \in (0,1), \eta_0 > 0, \mu_0 > 0, \delta \in (0,1)$.
for $k = 0, 1, \ldots$ do
    Compute $\tilde{x}^k = \texttt{PGLS}(x^k, y^k)$
    Define $W_k = \{(x, y) \in \mathcal{N}(\tilde{x}^k, y^k) \mid f(x) \leq f(\tilde{x}^k) + \xi\}$
    Set *success* = False
    while $W_k \neq \emptyset$ *and* *success* = False do
        select $(x', y') \in W_k$
        Set $z^1 = x'$
        for $j = 1, 2, \ldots$ do
            Compute $z^{j+1} = \texttt{PGLS}(z^j, y')$
            if $f(z^{j+1}) \leq f(\tilde{x}^k) - \eta_k$ then
                Set $x^{k+1}, y^{k+1} = z^{j+1}, y'$
                Set $\eta_{k+1} = \eta_k$
                Set *success* = *True*
                break
            if $\left\| z^j - \Pi_{\mathcal{X}(y')} \left[ z^j - \nabla f(z^j) \right] \right\| \leq \left\| x^k - \Pi_{\mathcal{X}(y^k)} \left[ x^k - \nabla f(x^k) \right] \right\| + \mu_k$
             then
                Set $W_k = W_k \setminus \{(x', y')\}$
                break
    if *success* = *False* then
        Set $x^{k+1}, y^{k+1} = \tilde{x}^k, = y^k$
        if $f(x^{k+1}) \leq f(x^k) - \eta_k$ then
            Set $\eta_{k+1} = \eta_k$
            *success* = True
        else
            Set $\eta_{k+1} = \theta \eta_k$
    Set $\mu_{k+1} = \delta \mu_k$
Output: The sequence $\{(x^k, y^k)\}$
```

The assumption above is a mild continuity assumption on the discrete neighborhoods and is equivalent to the lower semicontinuity of a point-to-set function (Berge, 1963).

We show that, for example, the neighborhood $\mathcal{N}_\rho$ defined in Definition 2.14 satisfies Assumption 6.3 in the case $X = \mathbb{R}^n$, as stated here below.

Proposition 6.1. *The point-to-set map $\mathcal{N}_\rho(x,y)$ defined in Definition 2.14 satisfies Assumption 6.3 when $X = \mathbb{R}^n$.*

Proof. Let $\{x^k, y^k\}$ be a sequence convergent to $\{\bar{x}, \bar{y}\}$. Then, for any $\epsilon > 0$, there exists k_ϵ such that $y^k = \bar{y}$ and $\|x^k - \bar{x}\| \leq \epsilon$ for all $k > k_\epsilon$.

Let $(\hat{x}, \hat{y}) \in \mathcal{N}_\rho(\bar{x}, \bar{y})$. For k sufficiently large, since $y^k = \bar{y}$, we have $\{y \mid y \in \mathcal{Y},\ d_H(y, y^k) \leq \rho\} = \{y \mid y \in \mathcal{Y},\ d_H(y, \bar{y}) \leq \rho\}$, hence $\hat{y} \in \{y \mid d_H(y, y^k) \leq \rho\}$ for all k.

Let us then consider the sequence $\{\hat{x}^k, \hat{y}^k\}$ where $\hat{y}^k = \hat{y}$ and $\hat{x}^k = H_{\Delta(y^k, \hat{y})}(x^k)$. We can observe that $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}_\rho(x^k, y^k)$. Now, let $j \in \{1, \dots, n\}$. The set $\Delta(y^k, \hat{y}^k) = \Delta(\bar{y}, \hat{y}) = \Delta$ is constant for k sufficiently large.

Noting that, being $X = \mathbb{R}^n$, $\Pi_{\mathcal{X}(\hat{y})}(H_\Delta(x)) = H_\Delta(x)$, we have for $j \notin \Delta$

$$\lim_{k\to\infty} \hat{x}_j^k = \lim_{k\to\infty} x_j^k = \bar{x}_j = \hat{x}_j.$$

On the other hand, if $j \in \Delta$, $\hat{x}_j^k = 0$ and $\hat{x}_j = 0$. Hence

$$\lim_{k\to\infty} \hat{x}^k = \hat{x}$$

and we thus get the thesis. □

The result still holds in the case $X \subset \mathbb{R}^n$.

Proposition 6.2. *Let $\{(x^k, y^k)\}$ be a sequence converging to $(\bar{x}, \bar{y})$. Then, the point-to-set map $\mathcal{N}_\rho(x,y)$ defined in Definition 2.14 satisfies Assumption 6.3.*

Proof. The proof follows exactly as in Proposition 6.1, recalling the continuity of the projection operator $\Pi_{\mathcal{X}(\hat{y})}$. □

Before turning to the convergence analysis of the algorithm, we prove a further useful preliminary result concerning the neighborhood $\mathcal{N}_\rho$.

Lemma 6.1. *Let $y \in \mathcal{Y}$ and $x \in \mathcal{X}(y)$ with $\delta = \|x\|_0$. Let us consider the set*

$$\bar{\mathcal{N}}(x) = \{(\hat{x}, \hat{y}) \mid y \in \{0,1\}^n,\ \hat{x} = x,\ e^\top \hat{y} = n - s,\ I_0(\hat{y}) \supseteq I_1(x)\ \}.$$

We have that

$$\bar{\mathcal{N}}(x) \subseteq \mathcal{N}_\rho(x, y),$$

when $\rho \geq 2(s - \delta)$.

Proof. Let $(\hat{x}, \hat{y})$ be any point in $\bar{\mathcal{N}}(x)$. From the feasibility of (x, y) we have

$$\delta \leq |I_0(y)| \leq s \qquad n - s \leq |I_1(y)| \leq n - \delta. \tag{6.3}$$

Moreover, from the definition of $\bar{\mathcal{N}}(x)$, we have

$$|I_0(\hat{y})| = s \qquad |I_1(\hat{y})| = n - s.$$

Now, it is easy to see that

$$d_H(y, \hat{y}) = n - |I_0(y) \cap I_0(\hat{y})| - |I_1(y) \cap I_1(\hat{y})|. \tag{6.4}$$

We can note that, since $I_0(y) \supseteq I_1(x)$ and $I_0(\hat{y}) \supseteq I_1(x)$, it has to be $I_0(y) \cap I_0(\hat{y}) \supseteq I_1(x)$. Therefore

$$|I_0(y) \cap I_0(\hat{y})| \geq |I_1(x)| = \delta. \tag{6.5}$$

We can now turn to $I_1(y) \cap I_1(\hat{y})$. Since the latter set can be equivalently written, by De Morgan's law, as $\{1, \dots, n\} \setminus (I_0(y) \cup I_0(\hat{y}))$, we can obtain

$$\begin{aligned}
|I_1(y) \cap I_1(\hat{y})| &= |\{1, \dots, n\} \setminus (I_0(y) \cup I_0(\hat{y}))| \\
&= n - |I_0(y) \cup I_0(\hat{y})| \\
&= n - (|I_0(y)| + |I_0(\hat{y})| - |I_0(y) \cap I_0(\hat{y})|) \\
&= n - |I_0(y)| - s + |I_0(y) \cap I_0(\hat{y})| \\
&\geq n - s - s + \delta \\
&= n - 2s + \delta,
\end{aligned}$$

where the second last inequality comes from (6.3) and (6.5). Putting everything together back in (6.4), we get

$$d_H(y, \hat{y}) \leq n - \delta - n + 2s - \delta = 2(s - \delta).$$

Taking into account that $\rho \geq 2(s - \delta)$ in the definition of $\mathcal{N}_\rho(x, y)$, we obtain

$$(\hat{x}, \hat{y}) \in \mathcal{N}_\rho(x, y),$$

thus getting the desired result. □

6.3 Convergence Analysis

We can now focus on the algorithms. First, we prove a property of Algorithm 6 that will play an important role in the convergence analysis of Algorithm 7.

Proposition 6.3. *Given a feasible point $(x, y) \in \mathcal{X}(y) \times \mathcal{Y}$, Algorithm 6 produces a feasible point $(\tilde{x}, y)$ such that*

$$f(\tilde{x}) \leq f(x) - \sigma\left(\left\|x - \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]\right\|\right),$$

where the function $\sigma(\cdot) \geq 0$ is such that if $\sigma\left(t^h\right) \to 0$ then $t^h \to 0$.

Proof. By definition, $d = \hat{x} - x$, where $\hat{x} = \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]$. By the properties of the projection operator, we can write

$$(x - \nabla f(x) - \hat{x})^\top (x - \hat{x}) \leq 0,$$

which, with simple manipulations, implies that

$$\nabla f(x)^\top d \leq -\|d\|^2 = -\left\|x - \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]\right\|^2. \tag{6.6}$$

By the instructions of the algorithm, either $\alpha = 1$ or $\alpha < 1$.

If $\alpha = 1$, then $\tilde{x} = x + d$ satisfies

$$f(\tilde{x}) \leq f(x) + \gamma \nabla f(x)^\top d \leq f(x) - \gamma\left\|x - \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]\right\|^2. \tag{6.7}$$

If $\alpha < 1$, we must have that

$$f(x + \alpha d) \leq f(x) + \gamma\alpha \nabla f(x)^\top d, \tag{6.8}$$

$$f\left(x + \frac{\alpha}{\delta} d\right) > f(x) + \gamma \frac{\alpha}{\delta} \nabla f(x)^\top d. \tag{6.9}$$

Applying the mean value theorem to equation (6.9), we get

$$\nabla f\left(x + \theta \frac{\alpha}{\delta} d\right)^\top d > \gamma \nabla f(x)^\top d,$$

where $\theta \in (0, 1)$. Adding and subtracting $\nabla f(x)^\top d$, and rearranging, we get

$$(1 - \gamma) \nabla f(x)^\top d > \left[\nabla f(x) - \nabla f\left(x + \theta \frac{\alpha}{\delta} d\right)\right]^\top d.$$

By the Lipschitz-continuity of $\nabla f(x)$, we can write

$$\left[\nabla f(x) - \nabla f\left(x + \theta \frac{\alpha}{\delta} d\right)\right]^\top d \geq -L \frac{\alpha}{\delta} \|d\|^2,$$

which means that

$$(1 - \gamma) \nabla f(x)^\top d > -L \frac{\alpha}{\delta} \|d\|^2,$$

Rearranging, we get

$$\frac{\delta}{L}(1-\gamma)\nabla f(x)^{\top}d > -\alpha \left\|d\right\|^2.$$

This last inequality, together with (6.6), yields

$$\frac{\delta}{L}(1-\gamma)\nabla f(x)^{\top}d > \alpha\nabla f(x)^{\top}d,$$

and substituting in equation (6.8) we finally get

$$f(\tilde{x}) < f(x) + \gamma\frac{\delta}{L}(1-\gamma)\nabla f(x)^{\top}d \leq f(x) - \gamma\frac{\delta}{L}(1-\gamma)\left\|x - \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]\right\|^2.$$

This last inequality, together with (6.7), implies that

$$f(\tilde{x}) \leq f(x) - \sigma\left(\left\|x - \Pi_{\mathcal{X}(y)}\left[x - \nabla f(x)\right]\right\|\right)$$

where

$$\sigma(t) = \gamma\min\left\{1, \frac{\delta}{L}(1-\gamma)\right\}t^2.$$

□

We can now state a couple of preliminary theoretical results. We first show that Algorithm 7 is well-posed.

Proposition 6.4. *For each iteration k, the loop between steps 9 and 18 of Algorithm 7 terminates in a finite number of steps.*

Proof. Suppose by contradiction that Steps 9-18 generate an infinite loop, so that an infinite sequence of points $\{z^j\}$ is produced for which

$$\left\|z^j - \Pi_{\mathcal{X}(y')}\left[z^j - \nabla f(z^j)\right]\right\| > \left\|x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\| + \mu_k > 0 \quad \forall j. \quad (6.10)$$

By Proposition 6.3, for each j we have that

$$f(z^{j+1}) - f(z^j) \leq -\sigma\left(\left\|z^j - \Pi_{\mathcal{X}(y')}\left[z^j - \nabla f(x^j)\right]\right\|\right), \quad (6.11)$$

where $\sigma(\cdot) \geq 0$. The sequence $\{f(z^j)\}$ is therefore nonincreasing. Moreover, equation (6.11) implies that

$$\left|f(z^{j+1}) - f(z^j)\right| \geq \sigma\left(\left\|z^j - \Pi_{\mathcal{X}(y')}\left[z^j - \nabla f(z^j)\right]\right\|\right). \quad (6.12)$$

By Assumption 6.2, $\{f(x^j)\}$ is lower bounded. Therefore, recalling that $\{f(z^j)\}$ is nonincreasing, we get that $\{f(z^j)\}$ converges, which implies that

$$\left|f(z^{j+1}) - f(z^j)\right| \to 0.$$

By (6.12), we get that $\sigma\left(\left\|z^j - \Pi_{\mathcal{X}(y')}\left[z^j - \nabla f(z^j)\right]\right\|\right) \to 0$, and, by the properties of $\sigma(\cdot)$, we finally get that $\left\|z^j - \Pi_{\mathcal{X}(y')}\left[z^j - \nabla f(z^j)\right]\right\| \to 0$, and this contradicts (6.10). □

The next proposition shows some properties of the sequences generated by the algorithm, which will play an important role in the subsequent analysis.

Proposition 6.5. *Let $\{(x^k, y^k)\}$, $\{\mu_k\}$ and $\{\eta_k\}$ be the sequences produced by Algorithm 7. Then:*

(i) *the sequence $\{f(x^k)\}$ is nonincreasing and convergent;*

(ii) *the sequence $\{(x^k, y^k)\}$ is bounded;*

(iii) *the set $K_u = \{k \mid \eta_k < \eta_{k-1}\}$ of unsuccessful iterates is infinite;*

(iv) $\lim_{k\to\infty} \mu_k = 0$*;*

(v) $\lim_{k\to\infty} \eta_k = 0$*;*

(vi) $\lim_{k\to\infty} \left\| x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\| = 0.$

Proof. (i) The instructions of the algorithm and Proposition 6.3 imply that $\{f(x^k)\}$ is nonincreasing, and Assumption 6.2 implies that $\{f(x^k)\}$ is lower bounded. Hence, $\{f(x^k)\}$ converges.

(ii) The instructions of the algorithm imply that each point (x^k, y^k) belongs to the level set $\mathcal{L}(x^0, y^0)$, which is compact by Assumption 6.2. Therefore, $\{(x^k, y^k)\}$ is bounded.

(iii) Suppose that K_u is finite. Then there exists $\bar{k} > 0$ such that all iterates satisfying $k > \bar{k}$ are successful, i.e.,

$$f(x^k) \le f(x^{k-1}) - \eta_{k-1},$$

and $\eta_k = \eta_{k-1} = \eta > 0$ for all $k \ge \bar{k}$. Since $\eta > 0$, this implies that $\{f(x^k)\}$ diverges to $-\infty$, in contradiction with (i).

(iv) Since, for all k, $\mu_{k+1} = \delta\mu_k$, where $\delta \in (0, 1)$, the claim holds.

(v) If $k \in K_u$, then $\eta_k = \theta\eta_{k-1}$, where $\theta \in (0, 1)$. Since K_u is infinite and $\eta_k = \eta_{k-1}$ if $k \notin K_u$, the claim holds.

(vi) By Proposition 6.3, we have that

$$f(\tilde{x}^k) - f(x^k) \le -\sigma\left(\left\| x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\|\right).$$

By the instructions of the algorithm, $f(x^{k+1}) \le f(\tilde{x}^k)$, and so we can write

$$f(x^{k+1}) - f(x^k) \le -\sigma\left(\left\| x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\|\right),$$

i.e.,
$$\left|f(x^{k+1}) - f(x^k)\right| \geq \sigma\left(\left\|x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\|\right).$$
Since $\{f(x^k)\}$ converges, we get that $\sigma\left(\left\|x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\|\right) \to 0$. By the properties of $\sigma(\cdot)$, we get that $\left\|x^k - \Pi_{\mathcal{X}(y^k)}\left[x^k - \nabla f(x^k)\right]\right\| \to 0$.

□

Before stating the main theorem of this section, it is useful to summarize some theoretical properties of the subsequence $\{(x^k, y^k)\}_{K_u}$ of the unsuccessful iterates. As the proof shows, the next proposition follows easily from the theoretical results we have shown above.

Proposition 6.6. *Let $\{(x^k, y^k)\}$ be the sequence of iterates generated by Algorithm 7, and let $K_u = \{k \mid \eta_k < \eta_{k-1}\}$. Then:*

(i) *$\{(x^k, y^k)\}_{K_u}$ admits accumulation points;*

(ii) *for any accumulation point $(x^\star, y^\star)$ of the sequence of unsuccessful iterates $\{(x^k, y^k)\}_{K_u}$, every point $(\hat{x}, \hat{y}) \in \mathcal{N}(x^\star, y^\star)$ is an accumulation point of a sequence $\{(\hat{x}^k, \hat{y}^k)\}_{K_u}$ where $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}(x^k, y^k)$.*

Proof. (i) By Proposition 6.5, item (ii), $\{(x^k, y^k)\}$ is bounded. Therefore, $\{(x^k, y^k)\}_{K_u}$ is also bounded, and so it admits accumulation points.

(ii) Assumption 6.3 implies that every $(\hat{x}, \hat{y}) \in \mathcal{N}(x^\star, y^\star)$ is an accumulation point of a sequence $\{(\hat{x}^k, \hat{y}^k)\}_{K_u}$, where $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}(x^k, y^k)$.

□

We can now prove the main theoretical result of this section.

Theorem 6.1. *Let $\{(x^k, y^k)\}$ be the sequence generated by Algorithm 7. Every accumulation point $(x^\star, y^\star)$ of $\{(x^k, y^k)\}_{K_u}$ is such that $x^\star$ is an $\mathcal{N}$-stationary point of problem* (6.1).

Proof. Let $(x^\star, y^\star)$ be an accumulation point of $\{(x^k, y^k)\}_{K_u}$. We must show that conditions (i)-(iii) of Definition 2.16 are satisfied.

(i) From the instructions of Algorithm 7 the iterates (x^k, y^k) belong to the set $\mathcal{L}(x^0, y^0)$, which is closed from Assumption 6.2. Any limit point $(x^\star, y^\star)$ belongs to $\mathcal{L}(x^0, y^0)$ and is thus feasible for problem (6.2).

(ii) The result follows from Proposition 6.5, item (vi).

(iii) Considering the way the set K_u is defined, we can observe that for all $k \in K_u$

$$x^k = \tilde{x}^{k-1}, \quad y^k = y^{k-1}.$$

We can thus denote

$$\mathcal{N}^k = \mathcal{N}(x^k, y^k) = \mathcal{N}(\tilde{x}^{k-1}, y^{k-1}).$$

Since $k \in K_u$, for all $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}^k$ either the test at step 11 failed or the point was not included in W_{k-1} and hence

$$f(\hat{x}^k) > f(\tilde{x}^{k-1}) - \eta_{k-1} = f(x^k) - \eta_{k-1}.$$

Since the sequence $\{f(x^k)\}$ is nonincreasing (Proposition 6.5, item (i)), we can write

$$f(x^*) \leq f(x^k) < f(\hat{x}^k) + \eta_{k-1}$$

for all $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}^k$. Taking limits, we get from Proposition 6.5, item (v), Assumption 6.3, and by the continuity of f that $f(x^*) \leq f(\hat{x})$ for all $(\hat{x}, \hat{y}) \in \mathcal{N}(x^*, y^*)$.

Now, note that item (i) of Proposition 6.5 ensures the existence of $f^* \in \mathbb{R}$ satisfying

$$\lim_{k \to \infty} f(x^k) = f(x^*) = f^*. \tag{6.13}$$

Consider any $(\hat{x}, \hat{y}) \in \mathcal{N}(x^*, y^*)$ such that

$$f(\hat{x}) = f^*. \tag{6.14}$$

Proposition 6.6 implies that the point $(\hat{x}, \hat{y})$ is an accumulation point of a sequence $\{(\hat{x}^k, \hat{y}^k)\}_{K_u}$, where $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}^k$. Therefore, by (6.13) and (6.14) we get, for k sufficiently large,

$$f(\hat{x}^k) < f(x^k) + \xi = f(\tilde{x}^{k-1}) + \xi.$$

Thus, for such values of k, we have

$$(\hat{x}^k, \hat{y}^k) \in W_{k-1} = \{(x, y) \in \mathcal{N}^k \mid f(x) \leq f(\tilde{x}^{k-1}) + \xi\}.$$

Steps 9-18 produce the points $z_{k-1}^2, \ldots, z_{k-1}^{j_{k-1}^*}$ (where j_{k-1}^* is the finite number of iterations of steps 9-18 until the test at step 16 is passed), which, by the instructions at Step 10 and by Proposition 6.3, satisfy

$$f(\hat{x}^k) \geq f(z_{k-1}^2) \geq \ldots \geq f(z_{k-1}^{j_{k-1}^*}). \tag{6.15}$$

Again, since $k \in K_u$, the test at step 11 is not passed at iteration $k-1$, and we can write

$$f(z_{k-1}^{j_{k-1}^*}) > f(\tilde{x}^{k-1}) - \eta_{k-1} = f(x^k) - \eta_{k-1}. \tag{6.16}$$

Moreover, as the sequence $\{(\hat{x}^k, \hat{y}^k)\}_{K_u}$ converges to the point $(\hat{x}, \hat{y})$, by (6.13), (6.14), (6.15), (6.16), and by item (v) of Proposition 6.5, we obtain

$$f^* = \lim_{k\to\infty, k\in K_u} f(\hat{x}^k) = \lim_{k\to\infty, k\in K_u} f(z_{k-1}^2) = \lim_{k\to\infty, k\in K_u} f(x^k) = f^*.$$

By Proposition 6.3, we have that

$$f(z_{k-1}^2) \le f(\hat{x}^k) - \sigma\left(\left\|\hat{x}^k - \Pi_{\mathcal{X}(\hat{y}^k)}\left[\hat{x}^k - \nabla f(\hat{x}^k)\right]\right\|\right),$$

which can be rewritten as

$$\left|f(z_{k-1}^2) - f(\hat{x}^k)\right| \ge \sigma\left(\left\|\hat{x}^k - \Pi_{\mathcal{X}(\hat{y}^k)}\left[\hat{x}^k - \nabla f(\hat{x}^k)\right]\right\|\right).$$

Taking limits for $k \to \infty, k \in K_u$, we finally get

$$\left\|\hat{x} - \Pi_{\mathcal{X}(\hat{y})}\left[\hat{x} - \nabla f(\hat{x})\right]\right\| = 0,$$

and the claim holds.

□

The above theorem states that, if any neighborhood $\mathcal{N}$ satisfying the continuity Assumption 6.3 is employed, then all limit points of the sequence produced by the SNS algorithm are $\mathcal{N}$-stationary.

Now, we show that a suitable choice of the neighborhood to be used within Algorithm 7 allows to obtain quite strong convergence properties.

For example, we show that, provided that $\mathcal{N}_\rho$ is employed as neighborhood in 7, with a sufficiently large value of ρ, the SNS procedure converges to basic feasible solutions.

Theorem 6.2. *Let $\{(x^k, y^k)\}$ be the sequence of iterates generated by Algorithm 7 equipped with $\mathcal{N}_\rho$ as neighborhood and $\mathcal{A}^\star$ the set of the accumulation points of the sequence of unsuccessful iterates $\{(x^k, y^k)\}_{K_u}$. If $\rho \ge 2(s - \delta^\star)$, in the definition of the set $\mathcal{N}_\rho(x, y)$, and $\delta^\star = \min\{\|x^\star\|_0 \mid (x^\star, y^\star) \in \mathcal{A}^\star\}$, then given a point $(x^\star, y^\star) \in \mathcal{A}^\star$, $x^\star$ is basic feasible for problem* (6.1).

Proof. Let $J \in \mathcal{J}(x^\star)$ and consider the vector $\hat{y}$ such that $\hat{y}_j = 1\ \forall j \notin J$ and zero otherwise. As $|J| = s$, we have $e^\top \hat{y} = n - s$. Moreover, $I_1(x^\star) \subseteq I_0(\hat{y})$, thus, using Lemma 6.1, we have $(x^\star, \hat{y}) \in \bar{\mathcal{N}}(x^\star) \subseteq \mathcal{N}_\rho(x^\star, y^\star)$. By taking into account Theorem

6.1, we finally get that $x^\star$ is an $\mathcal{N}_\rho$-stationary point of problem Problem (6.1) and that it is also a stationary point of

$$\begin{aligned} &\min f(x) \\ &\text{ s.t. } x \in \mathcal{X}(\hat{y}), \end{aligned}$$

that is

$$x^\star = \Pi_{\mathcal{X}(\hat{y})}(x^\star - \nabla f(x^\star)).$$

Then, by Lemma 2.1, recalling that $\hat{y}_i = 0$ if and only if $i \in J$, we obtain that $x^\star$ is basic feasible. □

Remark 6.1. At a first glance, the result in Theorem 6.2 may appear an ex post result. In fact, the value of $\delta^\star$ cannot be known in advance. However, $\delta^\star \geq 0$, hence we know a priori that the BF property will hold at limit points if we set $\rho = 2s$.

We shall also note that in most cases $\delta^\star$ will be not so far from s, hence small values of ρ should typically be enough to enforce the basic feasibility of solutions.

6.4 Convergence Guarantees under Constraint Qualifications

Continuing the discussion started at the end of the previous section, here we show that, under constraint qualifications and by choosing suitable neighborhoods, it is possible to state convergence results similar and even stronger than those obtained by other well-known algorithms, namely, the PD and the regularization approaches.

First we state the following assumption which implicitly involves constraint qualifications.

Assumption 6.4. Given $\bar{y} \in \mathcal{Y}$ and $\bar{x} \in \mathcal{X}(\bar{y})$, we have that $\bar{x}$ is a stationary point of problem (2.11) if and only if there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that

$$\begin{aligned} \nabla f(\bar{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\bar{x}) + \sum_{i=1}^{p} \mu_i \nabla h_i(\bar{x}) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(\bar{x}) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0,\ \forall\ i \text{ such that } \bar{y}_i = 0. \end{aligned}$$

The above assumption states that $\bar{x}$ is a stationary point of problem (2.11) if and only if it is a KKT point of the following problem

$$\begin{aligned} \min_x\ & f(x) \\ \text{s.t. } & h_i(x) = 0, \quad \forall i = 1, \dots, p, \\ & g_i(x) \leq 0, \quad \forall i = 1, \dots, m, \\ & x_i \bar{y}_i = 0, \quad\ \forall i = 1, \dots, n, \end{aligned}$$

which can be equivalenty rewritten as follows

$$\begin{aligned}
\min_x \; & f(x) \\
\text{s.t. } & h_i(x) = 0, \quad \forall i = 1, \dots, p, \\
& g_i(x) \leq 0, \quad \forall i = 1, \dots, m, \\
& x_i = 0, \qquad \forall i \in I_1(\bar{y}).
\end{aligned}$$

Remark 6.2. As shown in Appendix A, Assumption 6.4 holds when, e.g., the functions g_i are strongly convex with constant $c_i > 0$, for $i = 1, \dots, m$, the functions h_j, for $j = 1, \dots, p$ are affine, and some Cardinality Constraint-Constraint Qualification (CC-CQ) is satisfied. For instance, a standard CC-CQ is the Cardinality Constraint-Linear Independence Constraint Qualification (CC-LICQ) (Burdakov et al., 2016), requiring that the gradients

$$\begin{aligned}
\nabla g_i(\bar{x}) \qquad & \text{for all } i : g_i(\bar{x}) = 0 \\
\nabla h_i(\bar{x}) \qquad & \text{for all } i = 1, \dots, p \\
e_i \qquad & \text{for all } i \in I_1(\bar{y})
\end{aligned}$$

are linearly independent.

From Theorem 6.1, Proposition 2.9 and Assumption 6.4 we immediately get the following result.

Theorem 6.3. *Let $\{(x^k, y^k)\}$ be the sequence generated by Algorithm 7. Every accumulation point $(x^\star, y^\star)$ of the sequence of unsuccessful iterates $\{(x^k, y^k)\}_{K_u}$ is such that there exist multipliers $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^n$ such that*

$$\begin{aligned}
\nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\
\lambda_i \geq 0, \; \lambda_i g_i(x^\star) = 0, \; \forall i = 1, \dots, m, \\
\gamma_i = 0, \; \forall \; i \in I_0(y^\star).
\end{aligned} \tag{6.17}$$

Basically, the above proposition tells us that, under Assumption 6.4, the SNS algorithm produces points that are S-stationary and hence M-stationary for problem (6.1), as long as a neighborhood satisfying the continuity Assumption 6.3 is employed.

In order to state stronger convergence results, we again need to use suitable neighborhoods (e.g., $\mathcal{N}_\rho$ with a sufficiently large value of ρ) in the algorithm.

Theorem 6.4. *Let $\{(x^k, y^k)\}$ be the sequence generated by Algorithm 7 equipped with $\mathcal{N}_\rho$ as neighborhood and $\mathcal{A}^\star$ the set of the accumulation points of the sequence $\{(x^k, y^k)\}_{K_u}$ of unsuccessful iterates. If $\rho \geq 2(s - \delta^\star)$, in the definition of the set $\mathcal{N}_\rho(x, y)$, and $\delta^\star =$*

$\min\{\|x^\star\|_0 \mid (x^\star, y^\star) \in \mathcal{A}^\star\}$, *then given a point* $(x^\star, y^\star) \in \mathcal{A}^\star$ *and for every super support set* $J \in \mathcal{J}(x^\star)$, *we have that there exist multipliers* $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ *and* $\gamma \in \mathbb{R}^n$ *such that*

$$\begin{aligned} \nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) + \sum_{i=1}^{n} \gamma_i e_i = 0, \\ \lambda_i \geq 0,\ \lambda_i g_i(x^\star) = 0,\ \forall i = 1, \dots, m, \\ \gamma_i = 0,\ \forall\, i \in J, \end{aligned} \tag{6.18}$$

i.e., $x^\star$ *satisfies strong Lu-Zhang conditions for problem* (6.1).

Proof. Let $J \in \mathcal{J}(x^\star)$ and consider the vector $\hat{y}$ such that $\hat{y}_j = 1\ \forall j \notin J$ and zero otherwise. We have $I_1(x^\star) \subseteq I_0(\hat{y})$ and, as $|J| = s$, $e^\top \hat{y} = n - s$. Hence, $(x^\star, \hat{y}) \in \bar{\mathcal{N}}(x^\star) \subseteq \mathcal{N}_\rho(x^\star, y^\star)$, where we used Lemma 6.1. By taking into account Theorem 6.1, we finally get that $x^\star$ is an $\mathcal{N}_\rho$-stationary point of problem (6.1) and that it is also a stationary point of

$$\begin{aligned} &\min f(x) \\ &\text{ s.t. } x \in \mathcal{X}(\hat{y}). \end{aligned}$$

Then, by Assumption 6.4, recalling that $\hat{y}_i = 0$ if and only if $i \in J$, we obtain that (6.18) holds. □

Remark 6.3. Similarly as in Remark 6.1, we shall note that Theorem 6.4 guarantees us that all limit points of the sequence $\{x^k, y^k\}_{K_u}$ are such that strong Lu-Zhang conditions are satisfied if the neighborhood $\mathcal{N}_{2s}$ is employed in SNS.

Remark 6.4. It is interesting to note that, unlike the Penalty Decomposition algorithm, SNS is able by a suitable choice of $\mathcal{N}$ to guarantee the convergence to points satisfying strong LZ conditions. In fact, we know that, in the general case, the PD method only guarantees to generate points satisfying Lu-Zhang conditions.

The SNS algorithm would have the same exact convergence results as the PD method if we used the neighborhood

$$\mathcal{N}(x^k, y^k) = \{(x, y) \mid x = x^k,\ y \in \{0,1\}^n,\ e^\top y = n - s,\ y_i x_i^k = 0\,\forall\, i\}.$$

We have seen in Section 2.4 that the above neighborhood makes $\mathcal{N}$-stationarity equivalent to SLZ conditions. However, it does not satisfy the continuity Assumption 6.3: since some of the components of x^k may go to zero asymptotically, at the limit point a larger number of different y (i.e., super support sets) might be needed to be considered.

Hence, with the above neighborhood we would basically check all the super support sets at the current iterate x^k, but it would fail fail to guarantee condition (6.18) to be satisfied by all super support sets at the limit point.

6.5 Concluding Remarks

The introduction of the concept of discrete neighborhood into the analysis of cardinality-constrained problems had already allowed us to see in a new light most of the existing related literature. The Sparse Neighborhood Search framework, which is also base on this concept, additionally allowed to define tailored algorithmic schemes aimed at retrieving points satisfying various optimality conditions.

In particular, the SNS algorithm:

- theoretically outperforms the regularization approach in the general case, producing points that satisfy a stronger condition than S-stationarity;
- with a tailored neighborhood, provides basic feasible solutions in the general setting;
- with a tailored neighborhood, allows to guarantee the SLZ conditions, which the Penalty Decomposition algorithm falls short; the weakness of the PD approach can be characterized in terms of discontinuity of the underlying, implicitly used discrete neighborhood.

We will see later that scanning through the neighborhoods also provides the SNS method with higher exploration capabilities in practice thus getting to overall better solutions in terms of objective value than other state-of-the-art methods.

Chapter 7

Multi-Objective Sparsity-Constrained Optimization: Optimality Conditions and an Algorithmic Approach

In many real-world problems several different objectives have to be taken into account, most of them being in contrast with each other (Gravel et al., 1992; Carrizosa and Frenk, 1998; Fliege, 2001; Palermo et al., 2003; Liuzzi et al., 2003; Pellegrini et al., 2014; Sun et al., 2016). Arguably, the most popular classes of techniques employed to solve multi-objective problems are those of scalarization methods (Pascoletti and Serafini, 1984; Drummond et al., 2008; Eichfelder, 2009) and of heuristic methods based on genetic and evolutionary strategies (Deb et al., 2002; Laumanns et al., 2002; Konak et al., 2006).

However, both these families of approaches present shortcomings. Specifically, scalarization usually requires a deep analysis of the domain and the structure of the problem, in order to identify the weights defining a suitable scalarized objective; moreover, an unfortunate choice of the weights may lead to unbounded scalar problems, even under strong regularity assumptions (Fliege et al., 2009, sec. 7). On the other hand, heuristic methods hardly possess theoretical convergence properties.

To also overcome these limitations, extensions of classical scalar descent methods have been proposed to handle unconstrained and constrained vector optimization problems (see, e.g., Fliege and Svaiter, 2000; Fliege et al., 2009; Drummond and Iusem, 2004).

Few effort, however, has been put by the optimization community in the study of problems where the complexities caused by multiple objectives and sparsity requirements are simultaneously taken into account. The lack of theory and methodologies regarding cardinality-constrained multi-objective optimization has probably reduced the use of such a modeling tool in the practical context. Nonetheless, real-world applications exist that may benefit from the employment of specialized procedures for problems with this kind of formulation.

As an example, let us consider the mean/variance portfolio selection problem, which is one of the most famous ones from the optimization and financial economics literature (Markowitz, 1952, 1994). There exist for this problem both a multi-objective reformulation (Armananzas and Lozano, 2005; Radziukynienė and Žilinskas, 2008; Chen and Wei, 2019) and a sparse variant with cardinality constraints (Bienstock, 1996; Bertsimas and Shioda, 2009; Bertsimas and Cory-Wright, 2018). Indeed, a combination of the two has sometimes been considered in the literature (Chiam et al., 2008; Xidonas et al., 2018; Tian et al., 2019), even though solutions to the problem have then trivially been obtained by evolutionary algorithms or scalarization methods.

In this Chapter we consider multi-objective optimization problems with a cardinality constraint on the vector of decision variables and additional linear constraints. We extend the analysis of necessary and sufficient conditions of (Pareto) optimality from Chapter 2 to this class of problems.

We afterwards propose a Penalty Decomposition type algorithm, exploiting multi-objective descent methods, to tackle the aforementioned family of problems. The algorithm represents a direct extension of the procedure from Chapter 4 to the multi-objective case. We conduct a rigorous convergence analysis for the proposed method, where we prove that the produced sequence of points has limit points, each one being feasible and satisfying first-order optimality conditions.

7.1 Preliminaries

In multi-objective optimization, the aim is to simultaneously minimize a set of functions, i.e., we consider problems of the form

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \; & F(x) = (f_1(x),\dots,f_m(x))^T \\ \text{s.t. } & x \in C, \end{aligned} \tag{7.1}$$

where $C \subseteq \mathbb{R}^n$ is a closed convex set. As the components of F are typically in contrast with each other, there does not exist in the general case a solution minimizing them all together.

A partial ordering relation between vectors in $\mathbb{R}^m$ can be employed. Given two points $u, v \in \mathbb{R}^m$, we denote by $u \leq v$ when $u_i \leq v_i$ for all $i = 1,\dots,m$. We can introduce analogous notation for the other inequality relations $\leq, <, >$. Also, given $F : \mathbb{R}^n \to \mathbb{R}^m$, we say that $x \in \mathbb{R}^n$ dominates $z \in \mathbb{R}^n$ w.r.t. F if $F(x) \leq F(z)$ and $F(x) \neq F(z)$ and we denote this by $F(x) \lneqq F(z)$.

We are now able to recall the classical concept of Pareto optimality for multi-objective optimization.

Definition 7.1. A point $\bar{x} \in C$ is referred to as *Pareto optimal* for problem (7.1) if there does not exist $z \in C$ such that $F(z) \lneqq F(\bar{x})$, i.e., there does not exist $z \in C$ that dominates x.

Pareto optimality is a strong property. A slightly weaker, but more affordable concept is given by weak Pareto optimality.

Definition 7.2. A point $\bar{x} \in C$ is referred to as a *weak Pareto optimum* for problem (7.1) if there does not exist $z \in C$ such that $F(z) < F(\bar{x})$.

It is easy to prove that a Pareto optimal point is also weak-Pareto optimal. Similarly as in the scalar context, local optimality notions can also be introduced.

Definition 7.3. A point $\bar{x} \in C$ is called a *locally Pareto optimal solution* (respectively, *locally weak Pareto optimal*) if there exists a neighborhood $B(\bar{x}, \rho)$ such that $\bar{x}$ is a Pareto optimizer (respectively, a weak Pareto optimizer) for F restricted to $B(\bar{x}, \rho) \cap C$.

Convexity assumptions allow to state a relation of equivalence between local and global optima:

Lemma 7.1. *Consider problem* (7.1). *If F is component-wise convex, then each local Pareto optimal point is globally Pareto optimal.*

Assume now F is continuously differentiable on C, with Jacobian $J_F = (\nabla f_1, \dots, \nabla f_m)^T$. Then, we can introduce a further notion to characterize optimal points.

Definition 7.4. A point $\bar{x} \in C$ is Pareto-critical (or Pareto stationary) for problem (7.1) if

$$\min_{z \in C} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T (z - \bar{x}) = 0. \tag{7.2}$$

Lemma 7.2. *Equation* (7.2) *holds if and only if*

$$\min_{\substack{z \in C \\ \|z - \bar{x}\| \leq 1}} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T (z - \bar{x}) = 0. \tag{7.3}$$

Proof. The implication (7.2) $\Longrightarrow$ (7.3) is trivial. Now, assume (7.3) holds and assume by contradiction that (7.2) is not satisfied. Then, there exists $\bar{z} \in C$ such that, if $\bar{d} = \bar{z} - \bar{x}$, $\|\bar{d}\| > 1$ and $J_F(\bar{x})\bar{d} \ngeq 0$. By the convexity of C, if $\bar{x} + \bar{d} = \bar{z} \in C$, then $\bar{x} + t\bar{d} \in C$ for all $t \in [0, 1]$. Hence, $\bar{x} + \frac{\bar{d}}{\|\bar{d}\|} = \hat{z} \in C$. Also, $\|\hat{z} - \bar{x}\| = \|\bar{x} + \frac{\bar{d}}{\|\bar{d}\|} - \bar{x}\| = 1$. Moreover, $J_F(\bar{x})(\hat{z} - \bar{x}) = J_F(\bar{x})\frac{\bar{d}}{\|\bar{d}\|} = \frac{1}{\|\bar{d}\|} J_F(\bar{x})\bar{d} \ngeq 0$. This contradicts (7.3). □

Since a local/global weak/strong Pareto optimum is such that no feasible direction is a descent direction with respect to all the objectives simultaneously, it is easy to prove that, under differentiability assumptions, a local/global weak/strong Pareto optimizer is also Pareto-critical. The converse is not necessarily true.

Finally, let us state equivalence relationships between Pareto stationary points and Pareto optimizers under convexity assumptions.

Proposition 7.1 (Fliege et al. (2009))**.** *Consider problem* (7.1)*, assuming* $F \in C^1(\mathbb{R}^n)$*. Then, the following implications hold*

- *If* $x \in C$ *is Pareto critical for* (7.1) *and* F *is component-wise convex, then* x *is weakly Pareto optimal.*
- *If* $x \in C$ *is Pareto critical for* (7.1)*,* $F \in C^2(\mathbb{R}^n)$ *and* $\nabla^2 f_j(x)$ *is positive definite* $\forall\, j = 1, \dots, m$*, then* x *is a Pareto optimizer.*

Given a vector-valued function $F : C \to \mathbb{R}^m$, we can define level sets as follows.

Definition 7.5. Let $F : C \to \mathbb{R}^m$ be a vector function. For any $\zeta \in \mathbb{R}^m$, the level set ζ of F, denoted by $\mathcal{L}_F(\zeta)$, is defined as $\mathcal{L}_F(\zeta) = \{x \in C \mid F(x) \leq \zeta\}$.

Remark 7.1. In the remainder of this Chapter we will often consider functions whose level sets are bounded. Note that a continuous vector function with bounded level sets is necessarily bounded below. Also note that the level sets are closed by definition, hence all bounded level sets are compact.

Among many existing approaches for solving multi-objective problems, a particularly relevant class of algorithms is that of multi-objective descent method, whose prototypical incarnation for the convexly constrained case is the Multi-Objective Projected Gradiend Descent (MOPGD) method proposed by Drummond and Iusem (2004). This procedure, together with its theoretical analysis, is described in detail in Appendix B.

7.2 The Problem

In this Chapter, we consider multi-objective cardinality constrained problems, i.e., problems of the form

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \; & F(x) = (f_1(x), \dots, f_m(x))^T \\ \text{s.t. } & \|x\|_0 \leq s, \quad Ax = b, \\ & l \leq x \leq \mu, \end{aligned} \tag{7.4}$$

where $A \in \mathbb{R}^{p \times n}$ is a full rank matrix, $b \in \mathbb{R}^p$, $l, \mu \in \mathbb{R}^n$ with $l \leq 0 \leq \mu$ (l and μ may possibly be infinite), $F : \mathbb{R}^n \to \mathbb{R}^m$ is a continuously differentiable function with Jacobian J_F and $s < n$.

Consistently with the rest of this thesis, we denote by X the convex set $\{x \in \mathbb{R}^n \mid Ax = b,\ l \leq x \leq \mu\}$ and by $\mathcal{X}$ the feasible set $\{x \in \mathbb{R}^n \mid \|x\|_0 \leq s, x \in X\}$. In the following, we will always be assuming that the overall feasible set is nonempty. Linear inequality constraints could explicitly be included in the problem, we chose not to consider them for the sake of simplicity. On the other hand, the extension to general convex constraints $g(x) \leq 0$ would not be straightforward.

Since problem (7.4) is constrained, we should identify the set of feasible directions at a feasible point $\bar{x} \in \mathcal{X}$. We denote this set by $\mathcal{F}(\bar{x})$. By definition, $\mathcal{F}(\bar{x})$ is given by

$$\begin{aligned}\mathcal{F}(\bar{x}) &= \{d \in \mathbb{R}^n \mid \exists \bar{t} > 0 : \|\bar{x} + td\|_0 \leq s, A(\bar{x} + td) = 0, l \leq \bar{x} + td \leq \mu, \forall t \in (0, \bar{t}\,]\} \\ &= \{d \in \mathbb{R}^n \mid \|d_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0, d_i \leq 0 \text{ if } \bar{x}_i = \mu_i, d_i \geq 0 \text{ if } \bar{x}_i = l_i, Ad = 0\}.\end{aligned} \tag{7.5}$$

In order to validate the last equality, we prove the following statement.

Lemma 7.3. *Let $\bar{x} \in \mathbb{R}^n$ be a point such that $\|\bar{x}\|_0 \leq s$. The set of feasible directions at $\bar{x}$ w.r.t. the constraint $\|x\|_0 \leq s$ is given by*

$$\{d \in \mathbb{R}^n \mid \|d_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0\}.$$

Proof. Let d be a direction in $\mathbb{R}^n$. Consider the quantity $\|\bar{x}_{I_0(\bar{x})} + td_{I_0(\bar{x})}\|_0$, for $t > 0$. $\bar{x}_{I_0(\bar{x})} = 0$ by definition, so we can write

$$\|\bar{x}_{I_0(\bar{x})} + td_{I_0(\bar{x})}\|_0 = \|td_{I_0(\bar{x})}\|_0 = \|d_{I_0(\bar{x})}\|_0.$$

We first assume that $\|d_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0$ and show that d is feasible w.r.t. the cardinality constraint. For any $t > 0$ we have

$$\begin{aligned}\|\bar{x} + td\|_0 = \|\bar{x}_{I_1(\bar{x})} + td_{I_1(\bar{x})}\|_0 + \|\bar{x}_{I_0(\bar{x})} + td_{I_0(\bar{x})}\|_0 = \|\bar{x}_{I_1(\bar{x})} + td_{I_1(\bar{x})}\|_0 + \|d_{I_0(\bar{x})}\|_0 \\ \leq |I_1(\bar{x})| + \|d_{I_0(\bar{x})}\|_0 = \|\bar{x}\|_0 + \|d_{I_0(\bar{x})}\|_0 \leq \|\bar{x}\|_0 + s - \|\bar{x}\|_0 = s,\end{aligned}$$

that is, d is feasible.

Now, let $\|d_{I_0(\bar{x})}\|_0 > s - \|\bar{x}\|_0$. For any d, by the continuity axiom, we can always find t sufficiently small such that if $|\bar{x}_i| > 0$ then $|\bar{x}_i + td_i| > 0$. So, for any d and for t sufficiently small, $\|\bar{x}_{I_1(\bar{x})} + td_{I_1(\bar{x})}\|_0 = \|\bar{x}_{I_1(\bar{x})}\|_0$. Therefore for all $\bar{t} > 0$ there exists $t \in (0, \bar{t}\,]$ for which we can write

$$\begin{aligned}\|\bar{x} + td\|_0 &= \|\bar{x}_{I_1(\bar{x})} + td_{I_1(\bar{x})}\|_0 + \|\bar{x}_{I_0(\bar{x})} + td_{I_0(\bar{x})}\|_0 \\ &= \|\bar{x}_{I_1(\bar{x})}\|_0 + \|d_{I_0(\bar{x})}\|_0 = \|\bar{x}\|_0 + \|d_{I_0(\bar{x})}\|_0 > \|\bar{x}\|_0 + s - \|\bar{x}\|_0 = s.\end{aligned}$$

This completes the proof. □

Before turning to the discussion of an algorithmic approach to tackle problem (7.4), we need to characterize its solutions, analyzing necessary and sufficient conditions of Pareto local and global optimality. We will do this in the next section.

7.3 Optimality conditions

Pareto criticality, defined as in Definition 7.4, can be extended to the case of problem (7.4) by simply limiting the search among directions belonging to the feasible directions set $\mathcal{F}(\bar{x})$:

Definition 7.6. A point $\bar{x} \in \mathcal{X}$ is referred to as Pareto critical (or stationary) for problem (7.4) if

$$\min_{\substack{\|d\|\leq 1\\ d\in\mathcal{F}(\bar{x})}} \max_{j=1,\ldots,m} \nabla f_j(\bar{x})^T d = 0. \tag{7.6}$$

Looking carefully, we can realize that Definition 7.6 represents an extension of the BF property to the multi-objective case. Pareto-stationarity is indeed a condition of local (Pareto) optimality, stating that no feasible descent direction exists at a given point.

We now rigorously show that the properties of convexly-constrained Pareto critical points are mirrored in the case of problem (7.4).

Proposition 7.2. *Let $\bar{x} \in \mathcal{X}$ be a local weak Pareto point for problem* (7.4). *Then, $\bar{x}$ is Pareto critical for problem* (7.4).

Proof. Let $\bar{x}$ be a local weak Pareto point for problem (7.4) and assume it is not Pareto critical according to Definition 7.6. Then, there exists $d \in \mathcal{F}(\bar{x})$, $d \neq 0$, such that $J_F(\bar{x})d < 0$, i.e., $\nabla f_j(\bar{x})^T d < 0$ for all $j = 1,\ldots,m$. Therefore, for every $\bar{t} > 0$ there exists $t \in (0, \bar{t}\,]$ such that $f_j(\bar{x} + td) < f_j(\bar{x})$ for all $j = 1,\ldots,m$, that is, $F(\bar{x} + td) < F(\bar{x})$, which is absurd being $\bar{x}$ a locally weak Pareto optimum and d feasible at $\bar{x}$. □

Before turning to the statement about sufficent conditions of local Pareto optimality, we give a useful Lemma.

Lemma 7.4. *Let $\bar{x} \in \mathcal{X}$. Then there exists $\rho > 0$ such that for any $z \in B(\bar{x}, \rho) \cap \mathcal{X}$ there exists $d \in \mathcal{F}(\bar{x})$ such that we can write $z = \bar{x} + d$.*

Proof. By the continuity axiom, we have that there exists $t > 0$ such that if $\|z - \bar{x}\|_2 \leq t$ then $\|z_{I_1(\bar{x})}\|_0 = \|\bar{x}_{I_1(\bar{x})}\|_0$.

Let us assume that for all $\rho > 0$ there exists $z \in B(\bar{x}, \rho) \cap \mathcal{X}$ such that $\|z_{I_0(\bar{x})}\|_0 > s - \|\bar{x}\|_0$. Then this also holds for $\rho < t$. By definition $\|z\|_0 \leq s$, while $\|z_{I_0(\bar{x})}\|_0 = \|\bar{x}_{I_0(\bar{x})}\|_0$ since $\|z - \bar{x}\|_2 \leq \rho < t$. Thus

$$\begin{aligned} s \geq \|z\|_0 = \|z_{I_1(\bar{x})}\|_0 + \|z_{I_0(\bar{x})}\|_0 = \|\bar{x}_{I_1(\bar{x})}\|_0 + \|y_{I_0(\bar{x})}\|_0 \\ = \|\bar{x}\|_0 + \|y_{I_0(\bar{x})}\|_0 > \|\bar{x}\|_0 + s - \|\bar{x}\|_0 = s, \end{aligned}$$

which is absurd.

Hence, we have for ρ sufficiently small that

$$B(\bar{x},\rho) \cap \mathcal{X} \subseteq \{x \in B(\bar{x},\rho) \mid \|x_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0,\ x \in X\}.$$

Let z belong to the set on the left side of the above equation, for a suitable value of ρ. We know from (7.5) that $\mathcal{F}(\bar{x}) = \{d \in \mathbb{R}^n \mid \|d_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0,\ d_i \leq 0 \text{ if } \bar{x}_i = \mu_i,\ d_i \geq 0 \text{ if } \bar{x}_i = l_i,\ Ad = 0\}$. Now, let $d = z - \bar{x}$. We can write

$$\|d_{I_0(\bar{x})}\|_0 = \|z_{I_0(\bar{x})} - \bar{x}_{I_0(\bar{x})}\|_0 = \|z_{I_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0,$$
$$Ad = A(z - \bar{x}) = 0, \qquad d_i = z_i - \bar{x}_i \begin{cases} \leq \mu_i - \bar{x}_i = 0 \text{ if } \bar{x}_i = \mu_i, \\ \geq l_i - \bar{x}_i = 0 \text{ if } \bar{x}_i = l_i, \end{cases}$$

i.e., d is feasible at $\bar{x}$. Since z is arbitrary, we get the thesis. □

Proposition 7.3. *Assume F is a component-wise convex function and let $\bar{x} \in \mathcal{X}$ be a Pareto critical solution of problem* (7.4)*. Then, $\bar{x}$ is a local weak Pareto point for problem* (7.4)*.*

Proof. Let $\bar{x}$ be Pareto critical for problem (7.4) and F be component-wise convex. From the Pareto stationarity of $\bar{x}$, we have

$$\max_{j=1,\dots,m} (J_F(\bar{x})d)_j \geq 0$$

for all $d \in \mathcal{F}(\bar{x})$. That is, for any feasible directions d at $\bar{x}$ there exists $j \in \{1,\dots,m\}$ such that $\nabla f_j(\bar{x})^T d \geq 0$. Let hence d be feasible and j be such that $\nabla f_j(\bar{x})^T d \geq 0$. Recalling the convexity of f_j, we have that

$$f_j(\bar{x} + td) \geq f_j(\bar{x}) + \nabla f_j(\bar{x})^T(td) \geq f_j(\bar{x})$$

for all $t \geq 0$.

Therefore, for any $d \in \mathcal{F}(\bar{x})$ there exists j such that for all $t \geq 0$ it holds $f_j(\bar{x} + td) \geq f_j(\bar{x})$, i.e.,

$$F(\bar{x} + td) \not< F(\bar{x}) \qquad \forall\, d \in \mathcal{F}(\bar{x}) \text{ and } t \geq 0. \tag{7.7}$$

Assume by contradiction that for all $\bar{t} > 0$ there exists $z \in B(\bar{x},\bar{t}\,) \cap \mathcal{X}$ such that $F(z) < F(\bar{x})$. For $\bar{t}$ sufficiently small we know from Lemma 7.4 that $z = \bar{x} + d$ for some $d \in \mathcal{F}(\bar{x})$. But $F(\bar{x}) > F(z) = F(\bar{x} + d)$, which contradicts (7.7). We can finally say that there exists $\bar{t} > 0$ such that $F(\bar{x}) \not> F(y)$ for all $z \in B(\bar{x},\bar{t}) \cap \mathcal{X}$, i.e., $\bar{x}$ is a local weak Pareto optimizer for the problem. □

We now define an extension of the Lu-Zhang conditions for multi-objective problems. LZ conditions represent a more affordable condition to obtain in practice.

Definition 7.7. A point $\bar{x} \in \mathcal{X}$ satisfies the *Multi-objective Lu-Zhang first order optimality conditions* (MOLZ conditions) if there exists a super support set $J \in \mathcal{J}(\bar{x})$ such that

$$\min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{D}_J(\bar{x}) \\ d \in \mathbb{R}^s}} \max_{j=1,\dots,m} \nabla_J f_j(\bar{x})^T d = 0, \tag{7.8}$$

being

$$\mathcal{D}_J(\bar{x}) = \{d_J \in \mathbb{R}^s \mid d_i \leq 0 \text{ if } \bar{x}_i = \mu_i,\ d_i \geq 0 \text{ if } \bar{x}_i = l_i,\ A_J d_J = 0\},$$

where A_J denotes the submatrix of A made of the columns of indexes in J.

Similarly as in the scalar case, MOLZ conditions are necessary conditions of Pareto stationarity and, consequently, of local Pareto optimality. We formalize this fact in the next Propositions.

Proposition 7.4. *Let $\bar{x} \in \mathcal{X}$ be a Pareto critical point for problem* (7.4)*. Then, $\bar{x}$ satisfies MOLZ conditions.*

Proof. Let $\bar{x}$ be a local Pareto critical point for problem (7.4) and assume by contradiction that it does not satisfy MOLZ condition. Let $J \in \mathcal{J}(\bar{x})$ be any super support set.

From the absurd hypothesis, we have that

$$\min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{D}_J(\bar{x})}} \max_{j=1,\dots,m} \nabla_J f_j(\bar{x})^T d < 0.$$

Let

$$\mathcal{D}(\bar{x}) = \{d \in \mathbb{R}^n \mid d_i \leq 0 \text{ if } \bar{x}_i = \mu_i,\ d_i \geq 0 \text{ if } \bar{x}_i = l_i,\ Ad = 0\}.$$

Recalling (7.5), we see that $\mathcal{D}(\bar{x}) \cap \{d \in \mathbb{R}^n \mid \|d_{\bar{J}}\|_0 = 0\} \subseteq \mathcal{F}(\bar{x})$. We can therefore write

$$\begin{aligned} 0 &= \min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{F}(\bar{x})}} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T d \\ &\leq \min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{D}(\bar{x}) \\ \|d_{\bar{J}}\|_0 = 0}} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T d = \min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{D}_J(\bar{x})}} \max_{j=1,\dots,m} \nabla_J f_j(\bar{x})^T d < 0, \end{aligned}$$

which is absurd. The proof is hence complete. □

Corollary 7.1. *Let $\bar{x} \in \mathcal{X}$ be a local weak Pareto point for problem* (7.4)*. Then, $\bar{x}$ satisfies MOLZ conditions.*

Proof. Since $\bar{x}$ is a local weak Pareto point, it is Pareto critical from Proposition 7.2. But then, it satisfies MOLZ conditions from Proposition 7.4. □

Remark 7.2. The converse of Proposition 7.4, i.e., MOLZ conditions imply Pareto stationarity, is not necessarily true. We point this fact out by means of the example below.

Example 7.1. Consider Example 2.6, which can be seen as aninstance of problem (7.4), with $m = 1$. Let $\bar{x} = (1,\ 0,\ 0)^T$. We have

$$\nabla f(\bar{x}) = (2\bar{x}_1 - 2, \quad 2\bar{x}_2, \quad 2\bar{x}_3 - 2)^T = (0,\ 0,\ -2)^T.$$

Let $\bar{d} = -\nabla f(\bar{x})/\|\nabla f(\bar{x})\| = (0,\ 0,\ 1)$. $\bar{d} \in \mathcal{F}(\bar{x})$, as $\|\bar{d}_{I_0(\bar{x})}\|_0 = 1$ and $s - \|\bar{x}\|_0 = 1$. Then

$$\min_{\substack{\|d\| \leq 1 \\ d \in \mathcal{F}(\bar{x})}} \nabla f(\bar{x})^T d \leq \nabla f(\bar{x})^T \bar{d} = -2 < 0,$$

i.e., $\bar{x}$ is not Pareto critical for the problem.

On the other hand, let $J = \{1,2\} \in \mathcal{J}(\bar{x})$. $\nabla_J f(\bar{x}) = 0$, so $\nabla_J f(\bar{x})^T d = 0$ for any $d \in \mathbb{R}^s$. Hence

$$\min_{\substack{\|d\| \leq 1 \\ d \in \mathbb{R}^s}} \nabla_J f(\bar{x})^T d = 0,$$

i.e., $\bar{x}$ satisfies MOLZ conditions for problem (7.4).

In order to obtain an equivalence result w.r.t. Pareto stationarity, MOLZ conditions have to be strengthened. Of course, in order for a point to be Pareto critical, MOLZ conditions have to be satisfied by all possible super support sets. In the following, we finally prove that Pareto stationarity is in fact equivalent to *strong MOLZ conditions*.

Proposition 7.5. *A point $\bar{x} \in \mathcal{X}$ is a Pareto critical point for problem* (7.4) *if and only if it satisfies MOLZ conditions for all $J \in \mathcal{J}(\bar{x})$.*

Proof. In order to prove the thesis, we just need to show that

$$\bigcup_{J \in \mathcal{J}(\bar{x})} \{d \in \mathbb{R}^n \mid \|d_{\bar{J}}\|_0 = 0,\ d_i \leq 0 \text{ if } \bar{x}_i = \mu_i,\ d_i \geq 0 \text{ if } \bar{x}_i = l_i,\ Ad = 0\} = \mathcal{F}(\bar{x}).$$

Let us denote by $\mathcal{U}(\bar{x})$ the set on the left side of the above equation.

First we prove that if $d \in \mathcal{U}(\bar{x})$, then $d \in \mathcal{F}(\bar{x})$. Let $J \in \mathcal{J}(\bar{x})$ be a super support set such that $\|d_{\bar{J}}\|_0 = 0$, which exists since $d \in \mathcal{U}(\bar{x})$. Then

$$\|d_{I_0(\bar{x})}\|_0 = \|d_{\bar{J}}\|_0 + \|d_{J \cap I_0(\bar{x})}\|_0 = \|d_{J \cap I_0(\bar{x})}\|_0 \leq |J \cap I_0(\bar{x})| = s - \|\bar{x}\|_0,$$

that is, $d \in \mathcal{F}(\bar{x})$.

Now, let $d \in \mathcal{F}(\bar{x})$ and let $R \subseteq I_0(\bar{x})$ such that $d_{I_0(\bar{x})\setminus R} = 0$ and $|R| = s - \|\bar{x}\|_0$. Let $J = I_1(\bar{x}) \cup R$. We have $\|d_{\bar{J}}\|_0 = 0$ by the definition of R and J ($\bar{J} = I_0(\bar{x}) \setminus R$). Also $|J| = |I_1(\bar{x}) \cup R| = \|\bar{x}\|_0 + s - \|\bar{x}\|_0 = s$, since $R \subseteq I_0(\bar{x})$ and $I_0(\bar{x}) \cap I_1(\bar{x}) = \emptyset$. As $\bar{x}_{\bar{J}} = 0$, being $\bar{J} \subseteq I_0(\bar{x})$, $d \in \mathcal{U}(\bar{x})$ and the proof is complete. □

7.4 A Penalty Decomposition Scheme

Analogously as in the scalar case, problem (7.4) can be equivalently expressed as

$$\begin{aligned} \min_{x,z \in \mathbb{R}^n} \; & F(x) = (f_1(x), \dots, f_m(x))^T \\ \text{s.t.} \; & \|z\|_0 \leq s, \\ & Ax = b, \\ & l \leq x, z \leq \mu, \\ & x - z = 0. \end{aligned} \tag{7.9}$$

In the above formulation we associated the bound constraints to both blocks of variables. We will motivate this choice shortly. We will refer to the feasible set of the z variable, i.e., $\{z \in \mathbb{R}^n \mid \|z\|_0 \leq s,\ l \leq z \leq \mu\}$, by Z. A quadratic penalty function can be associated with problem (7.9).

Definition 7.8. We define the *multi-objective penalty function* of penalty parameter τ associated to problem (7.9) as

$$Q_\tau(x,z) = F(x) + \frac{\tau}{2}(\|x - z\|^2 + \|Ax - b\|^2)e.$$

Note that, in a multi-objective setting, a objective penalty function shall be obtained adding the penalty term to all the components of the objective function (Cocchi and Lapucci, 2020). Keeping both blocks of variables x and z inside the box allows not to add penalty terms associated with the bound constraints. We denote the components of $Q_\tau(x,z)$ by $q_j(x,z;\tau)$, $j = 1, \dots, m$, that is,

$$Q_\tau(x,z) = (q_1(x,z;\tau), \dots, q_m(x,z;\tau))^T.$$

Let us also define the quantity

$$\theta_{Q_\tau}(x;z) = \min_{\substack{\|d\| \leq 1 \\ l \leq x+d \leq \mu}} \max_{j=1,\dots,m} \nabla_x q_j(x,z;\tau)^T d.$$

Problem (7.9), and consequently the original problem (7.4), can be solved by an alternate minimization scheme, similarly as what is done in the scalar Penalty Decomposition approach. The Multi-Objective Sparse Penalty Decomposition (MO-SPD) procedure proposed in this work is described in Algorithm 8.

The MultiObjectiveDescent(ϕ, p^0, ϵ) procedure invoked in Algorithm 8 is intended to run one of the existing multi-objective descent algorithms on the objective function ϕ, starting at point p^0, until the current solution p^κ is ϵ-stationary, i.e., $\theta_\phi(p^\kappa) \geq -\epsilon$. From here on, we will assume that Algorithm 9 is employed.

The algorithm starts at a point (x^0, y^0) which is feasible for problem (7.9). At every iteration Algorithm 8 repeats a run of (inexact) steepest descent on Q_{τ_k} w.r.t. x and a projection operation onto the feasible set Z, which can be equivalently seen as the exact minimization of each component of the penalty function w.r.t. variable z. Note that formula (2.2) continues to hold even in the case of bound-constrained problems. As soon as the solution at the end of an iteration is approximately critical w.r.t. the x block for the function Q_{τ_k}, the penalty parameter τ_k is increased for the successive iteration, while the stationarity approximation degree ε_k is decreased.

At the beginning of each outer iteration, before starting the "alternate minimization" loop, a test is performed to ensure that the procedure will keep the iterates inside an appropriate level set. If the test is passed, the inner loop will start from the point produced at the previous iteration, otherwise from the starting point, which is guaranteed to satisfy the desired property. In the following Section we will address the asymptotic convergence properties of the proposed algorithm.

Algorithm 8: MultiObjectiveSparsePenaltyDecomposition

```
Input: $\tau_0 > 0, \sigma > 1, x^0 = z^0 \in \mathbb{R}^n$ s.t. $\|x^0\|_0 \leq s$, a sequence $\{\varepsilon_k\}$ s.t. $\varepsilon_k \to 0$.
for $k = 0, 1, \dots$ do
    $\ell = 0$
    $x_{\text{trial}} = \texttt{MultiObjectiveDescent}(Q_{\tau_k}(\cdot, z^k), x^k, \varepsilon_k)$
    if $Q_{\tau_k}(x_{\text{trial}}, z^k) \leq F(x^0)$ then
        $u^0, v^0 = x^k, z^k$
    else
        $u^0, v^0 = x^0, z^0$
    while $\theta_{Q_{\tau_k}}(u^\ell; v^\ell) < -\varepsilon_k$ do
        $u^{\ell+1} = \texttt{MultiObjectiveDescent}(Q_{\tau_k}(\cdot, v^\ell), u^\ell, \varepsilon_k)$
        $v^{\ell+1} = \arg\min_{v \in Z} \frac{\tau_k}{2} \|u^{\ell+1} - v\|^2$
        $\ell = \ell + 1$
    $\tau^{k+1} = \sigma \tau_k$
    $x^{k+1} = u^\ell$
    $z^{k+1} = v^\ell$
Output: The sequence $\{x^k, z^k\}$.
```

7.5 Convergence analysis

We start the theoretical analysis of the MOSPD algorithm proving a technical result which will be needed in the subsequent proofs.

Lemma 7.5. *If $F : \mathbb{R}^n \to \mathbb{R}^m$ has bounded level sets in the multi-objective sense, then, for any $\tau \geq 0$, the penalty function $(x, z) \in \mathbb{R}^n \times \mathbb{R}^n \mapsto Q_\tau(x, z) \in \mathbb{R}^m$ associated with problem* (7.9) *has bounded level sets.*

Proof. Consider an arbitrary $\zeta \in \mathbb{R}^m$. From the hypotheses, $\mathcal{L}_F(\zeta)$ is bounded. Now, let us consider the level set $\mathcal{L}_{Q_\tau}(\zeta)$, for any $\tau > 0$.

Assume by contradiction that $\mathcal{L}_{Q_\tau}(\zeta)$ is not bounded, that is, there exists a sequence $\{x^k, z^k\}$ such that $(x^k, z^k) \in \mathcal{L}_{Q_\tau}(\zeta)$ for all k and $\|(x^k, z^k)\| \to \infty$. Then, either $\|x^k\| \to \infty$ or $\|z^k\| \to \infty$.

If $\|x^k\| \to \infty$, we have $F(x^k) \not\leq \zeta$ for k sufficiently large, as the level set $\mathcal{L}_F(\zeta)$ is bounded. But then, recalling the definition of Q_τ, we have for k sufficiently large $Q_\tau(x^k, z^k) \geq F(x^k) \not\leq \zeta$, i.e., there exists j such that

$$q_j(x^k, z^k; \tau) \geq f_j(x^k) > \zeta_j,$$

contradicting $Q_\tau(x^k, z^k) \leq z$.

Hence $\|z^k\| \to \infty$ while $\{x^k\}$ stays bounded. But $Q_\tau(x^k, z^k) = F(x^k) + \frac{\tau}{2}(\|x^k - z^k\|^2 + \|Ax^k - b\|^2)e \not\leq \zeta$ for k sufficiently large, as $\|x^k - z^k\|^2 \to \infty$, $\|Ax^k - b\|^2 \geq 0$ and F is bounded below, having bounded level sets. This is again a contradiction, which completes the proof. □

Assumption 7.1. From now on, we will make the assumption that the objective function F of problem (7.4) has bounded level sets.

Next, in order to state that the whole algorithm is well posed, we show that the `MultiObjectiveProjectedGradientDescent` method produces an approximate Pareto critical point for the penalty function w.r.t. variable x in finite time.

Lemma 7.6. *The MOPGD procedure employed at line 10 of Algorithm 8 produces a point $u^{\ell+1}$ such that $\theta_{Q_{\tau_k}}(u^{\ell+1}; v^\ell) \geq -\varepsilon_k$ in a finite number of iterations.*

Proof. Assume by contradiction that the assertion is false, i.e., the MOPGD procedure produces an infinite sequence $\{u^t\}$ such that $\theta_{Q_{\tau_k}}(u^t; v^\ell) < -\varepsilon_k$ for all $t = 0, 1, \ldots$.

From Lemma 7.5 and Proposition B.2 we know that there exists $T \subseteq \{0, 1, \ldots\}$ such that $u^t \to \bar{u}$ for $t \to \infty$, $t \in T$, with $\bar{u}$ Pareto critical for $Q_{\tau_k}(\cdot, v^\ell)$, i.e., recalling Lemma 7.2, $\theta_{Q_{\tau_k}}(\bar{u}; v^\ell) = 0$. From Proposition B.2 we know that $\theta_{Q_{\tau_k}}$ is a continuous function, hence for all $t \in T$ sufficiently large it has to hold $\theta_{Q_{\tau_k}}(u^t; v^\ell) \geq -\varepsilon_k$, a contradiction. We hence get the thesis. □

In order to assess algorithm completeness, we also have to show the finiteness of the inner loop.

Lemma 7.7. *Algorithm 8, equipped with MOPGD as descent procedure, does not loop infinitely between steps 9-12.*

Proof. Assume by contradiction that the proposition is false and that, at a certain iteration k, the sequence $\{u^\ell, v^\ell\}$ is infinite, which means that

$$\theta_{Q_{\tau_k}}(u^\ell; v^\ell) < -\varepsilon_k \tag{7.10}$$

for all ℓ. From the instructions of the MOPGD algorithm and Proposition B.2 we have that

$$Q_{\tau_k}(u^{\ell+1}, v^\ell) < Q_{\tau_k}(u^\ell, v^\ell).$$

Moreover, recalling that $v^{\ell+1} \in \arg\min_{v \in Z} \frac{\tau_k}{2}\|u^{\ell+1} - v\|^2$ and $v^\ell \in Z$, we have

$$Q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) - Q_{\tau_k}(u^{\ell+1}, v^\ell) = \frac{\tau_k}{2}\|u^{\ell+1} - v^{\ell+1}\|^2 e - \frac{\tau_k}{2}\|u^{\ell+1} - v^\ell\|^2 e \leq 0.$$

Hence we can write

$$Q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq Q_{\tau_k}(u^{\ell+1}, v^\ell) < Q_{\tau_k}(u^\ell, v^\ell), \tag{7.11}$$

from which we can deduce that $\{u^\ell, v^\ell\}$ belongs to the compact set $\mathcal{L}_{Q_{\tau_k}}(Q_{\tau_k}(u^0, v^0))$. Therefore, there exists $K \subseteq \{0, 1, \ldots\}$ such that $(u^\ell, v^\ell) \to (\bar{u}, \bar{v})$ for $k \in K, k \to \infty$.

F is continuously differentiable, and so is Q_{τ_k}. Thus, $J_{Q_{\tau_k}}(u^\ell, v^\ell) \to J_{Q_{\tau_k}}(\bar{u}, \bar{v})$ for $k \in K, k \to \infty$. Also, from (7.11), we know that the whole sequence $\{Q_{\tau_k}(u^\ell, v^\ell)\}$ is monotonically decreasing and thus convergent to some value $\bar{Q}$, which is finite as Q_{τ_k} has bounded level sets and is thus bounded below. So

$$Q_{\tau_k}(u^\ell, v^\ell) \to \bar{Q} > -\infty. \tag{7.12}$$

We can rewrite (7.10) as

$$\min_{\substack{\|d\| \leq 1 \\ l \leq x+d \leq u}} \max_{j=1,\ldots,m} (J_{Q_{\tau_k}}(u^\ell, v^\ell) d)_j < -\varepsilon_k. \tag{7.13}$$

Let

$$d^\ell \in \arg\min_{\substack{\|d\| \leq 1 \\ l \leq x+d \leq \mu}} \max_{j=1,\ldots,m} \nabla_x q_j(u^\ell, v^\ell; \tau_k)^T d.$$

From (7.13) we have

$$J_{Q_{\tau_k}}(u^\ell; v^\ell) d^\ell < -\varepsilon_k e < 0. \tag{7.14}$$

Now, the sequence $\{d^\ell\}$ is bounded ($\|d^\ell\| \leq 1$), so taking the limits (along a suitable subsequence, if necessary), we have $d^\ell \to \bar{d}$ as $\ell \to \infty$, $\ell \in K$. Taking the limits in (7.14), we get

$$\lim_{\substack{\ell\to\infty \\ \ell\in K}} J_{Q_{\tau_k}}(u^\ell; v^\ell)d^\ell = J_{Q_{\tau_k}}(\bar{u};\bar{v})\bar{d} \leq -\varepsilon_k e < 0. \tag{7.15}$$

From the instructions of the MOPGD Algorithm equipped with the Armijo-type line search (Algorithm 10), we know that

$$\begin{aligned} Q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) &\leq Q_{\tau_k}(u^{\ell+1}, v^\ell) \leq Q_{\tau_k}(u^\ell + \alpha_\ell d^\ell, v^\ell) \\ &\leq Q_{\tau_k}(u^\ell, v^\ell) + \beta\alpha_\ell \left[\max_{j=1,\ldots,m} \nabla_x q_j(u^\ell, v^\ell; \tau_k)^T d^\ell\right] e. \end{aligned}$$

Taking the limits for $\ell \in K$, $\ell \to \infty$, recalling (7.12) and (7.15), we get

$$0 \leq \lim_{\substack{\ell\to\infty \\ \ell\in K}} \beta\alpha^\ell \max_{j=1,\ldots,m} \nabla_x q_j(u^\ell, v^\ell; \tau_k)^T d \leq \lim_{\substack{\ell\to\infty \\ \ell\in K}} -\beta\alpha_\ell\varepsilon_k.$$

Now, assume that $\alpha_\ell \to 0$. From the instructions of Algorithm 10, we have that for all $q \in \mathbb{N}$ there exists $\bar{\ell} \in K$ such that for all $\ell \in K$, $\ell \geq \bar{\ell}$, we have

$$Q_{\tau_k}\left(u^\ell + \frac{1}{2^q}d^\ell; v^\ell\right) \nleq Q_{\tau_k}(u^\ell) + \frac{\beta}{2^q} J_{Q_{\tau_k}}(u^\ell)d^\ell.$$

Taking the limits for $\ell \to \infty$ as $\ell \in K$, $\ell \to \infty$, along a suitable subsequence if needed, we have that for some j it holds

$$q_j\left(\bar{u} + \frac{1}{2^q}\bar{d}, \bar{v}; \tau_k\right) \geq q_j(\bar{u}, \bar{v}; \tau_k) + \frac{\beta}{2^q}\nabla_x q_j(\bar{u}, \bar{v}; \tau_k)^T\bar{d}.$$

Being q arbitrary, we have from Proposition B.1 that

$$\max_{j=1,\ldots,m} \nabla_x q_j(\bar{u}, \bar{v}; \tau_k)^T \bar{d} \geq 0,$$

which contradicts (7.15).

Hence, there exists $\nu > 0$ such that $\alpha_\ell \geq \nu$ for all $\ell \in K$ sufficiently large. Thus, we get

$$0 \leq \lim_{\substack{\ell\to\infty \\ \ell\in K}} -\beta\alpha_\ell\epsilon_k < 0,$$

which is again a contradiction. □

We are finally able to address the asymptotic convergence properties of Algorithm 8. We begin by stating the existence and the feasibility of limit points of the generated sequence.

Proposition 7.6. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 8. Then $\{x^k, z^k\}$ admits cluster points. All limit points $(\bar{x}, \bar{z})$ are feasible for problem* (7.9)*, and $\bar{x}$ is feasible for problem* (7.4)*.*

Proof. Consider a generic iteration k. Since instructions 10-11 of the algorithm both do not increase the value of any of the components of Q_{τ_k}, we have that

$$Q_{\tau_k}(x^{k+1}, z^{k+1}) \leq ... \leq Q_{\tau_k}(u^1, v^0) \leq Q_{\tau_k}(u^0, v^0). \tag{7.16}$$

From the definition of (u^0, v^0), we either have $(u^0, v^0) = (x^k, z^k)$ or $(u^0, v^0) = (x^0, z^0)$. In the first case, we have, by the definition of x_{trial}, that

$$Q_{\tau_k}(u^1, v^0) = Q_{\tau_k}(x_{\text{trial}}, z^k) \leq F(x^0),$$

where the last inequality holds, as in this case the condition at line 5 is satisfied. In the second case we have

$$\begin{aligned} Q_{\tau_k}(u^1, v^0) \leq Q_{\tau_k}(u^0, v^0) &= Q_{\tau_k}(x^0, z^0) \\ &= F(x^0) + \frac{\tau_k}{2}\|x^0 - z^0\|^2 e = F(x_0). \end{aligned}$$

So, putting everything together back in (7.16) we get

$$Q_{\tau_k}(x^{k+1}, y^{k+1}) \leq F(x^0). \tag{7.17}$$

But, by the definition of Q_{τ_k} it also holds $F(x^{k+1}) \leq Q_{\tau_k}(x^{k+1}, z^{k+1})$, so $F(x^{k+1}) \leq F(x^0)$. As k is arbitrary, it follows that $\{x^{k+1}\} \subset \mathcal{L}_F(F(x^0))$, which is compact by the assumptions; hence $\{x^k\}$ is bounded.

From equation (7.17), we also have

$$Q_{\tau_k}(x^{k+1}, z^{k+1}) = F(x^{k+1}) + \frac{\tau_k}{2}\|x^{k+1} - z^{k+1}\|^2 e \leq F(x^0).$$

Hence, for any $j = 1, \ldots, m$, we have

$$q_j(x^{k+1}, z^{k+1}; \tau_k) = f_j(x^{k+1}) + \frac{\tau_k}{2}(\|x^{k+1} - z^{k+1}\|^2 + \|Ax^{k+1} - b\|^2) \leq f_j(x^0).$$

Dividing by τ_k we get

$$\|x^{k+1} - y^{k+1}\|^2 + \|Ax^{k+1} - b\|^2 \leq 2\frac{f_j(x^0) - f_j(x^{k+1})}{\tau_k}. \tag{7.18}$$

Taking the limits for $k \to \infty$, recalling the boundedness of $\{x^{k+1}\}$, that F is bounded below and that $\tau_k \to \infty$, we have that $Ax^{k+1} - b \to 0$ and $x^{k+1} - z^{k+1} \to 0$; the latter implies that z^k is also a bounded sequence. Hence, the sequence $\{x^k, z^k\}$ is bounded and therefore admits limit points. Let $(\bar{x}, \bar{z})$ be one of such limit points, i.e., there exists $K \subseteq \{0, 1, \ldots\}$ such that $\{x^{k+1}, z^{k+1}\} \to (\bar{x}, \bar{z})$ for $k \to \infty$, $k \in K$; then, taking the limits in (7.18) for $k \to \infty$, $k \in K$, we get

$$\|\bar{x} - \bar{z}\|^2 + \|A\bar{x} - b\|^2 \leq 0,$$

which completes the proof. □

Now, we can finally turn to the proposition assessing optimality results for all limit points of $\{x^k\}$, after a technical Lemma is given which is necessary for the proof.

Lemma 7.8. *Let $U \subset \{1, \ldots, n\}$, $k \leq |U|$, $A \in \mathbb{R}^{p \times n}$, $l \leq x \leq \mu$ and $F : \mathbb{R}^n \to \mathbb{R}^m$ continuously differentiable. Let also $B = \{d \mid l \leq x + d \leq \mu,\ Ad = 0\}$ and $\bar{B} = \{d \mid d_i \geq 0 \text{ if } x_i = l_i,\ d_i \leq 0 \text{ if } x_i = \mu_i,\ Ad = 0\}$. If*

$$\min_{\substack{\|d\| \leq 1 \\ d \in B \\ \|d_U\|_0 \leq k}} \max_{j=1,\ldots,m} \nabla f_j(x)^T d = 0 \tag{7.19}$$

then

$$\min_{\substack{\|d\| \leq 1 \\ d \in \bar{B} \\ \|d_U\|_0 \leq k}} \max_{j=1,\ldots,m} \nabla f_j(x)^T d = 0. \tag{7.20}$$

Proof. Assume that (7.19) holds and, by contradiction, that (7.20) is not satisfied. Let

$$\bar{d} \in \arg\min_{\substack{\|d\| \leq 1 \\ d \in \bar{B} \\ \|d_U\|_0 \leq k}} \max_{j=1,\ldots,m} \nabla f_j(x)^T d.$$

It's easy to see that $B \subseteq \bar{B}$, that $0 \in B$ and that both B and $\bar{B}$ are convex sets. Hence, there exists $0 < t < 1$ sufficiently small such that $t\bar{d} \in B$. Also, we have $\|t\bar{d}\| \leq 1$ and $\|t\bar{d}_U\| = \|\bar{d}_U\|_0 \leq k$. Hence $t\bar{d}$ is feasible for the problem in (7.19). But $\max_{j=1,\ldots,m} \nabla f_j(x)^T (t d) = t \max_{j=1,\ldots,m} \nabla f_j(x)^T d < 0$. The last two statements combined contradict (7.19). □

Proposition 7.7. *Let $\{x^k, z^k\}$ be the sequence generated by Algorithm 8 applied to problem* (7.9). *Suppose that $(\bar{x}, \bar{z})$ is a limit point of $\{x^{k+1}, z^{k+1}\}$, i.e., there exists $K \subseteq \{0, 1, \ldots\}$ such that $(x^k, z^k) \to (\bar{x}, \bar{z})$ for $k \to \infty$, $k \in K$. Then $\bar{x}$ satisfies MOLZ conditions for problem* (7.4).

Proof. We know from Proposition 7.6 that $\bar{x} = \bar{z}$ and $A\bar{x} = b$. Moreover, from the instructions of the algorithm, at each iteration k we have that

$$\min_{\substack{\|d\| \leq 1 \\ l \leq x^{k+1} + d \leq \mu}} \max_{j=1,\ldots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T (A x^{k+1} - b) + \tau_k (x^{k+1} - z^{k+1}) \right)^T d \geq -\varepsilon_k$$

and, recalling that $\varepsilon_k \to 0$,

$$\lim_{\substack{k\to\infty\\ k\in K}} \min_{\substack{\|d\|\le 1\\ l\le x^{k+1}+d\le \mu}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d = 0. \tag{7.21}$$

In addition, we have

$$z^{k+1} = \arg\min_{z\in Z} \tau_k \|x^{k+1}-z\|^2, \tag{7.22}$$

which, recalling (2.2), gives

$$\begin{cases} z_i^{k+1} = x_i^{k+1} & \text{if } i \in \mathcal{G}(x^{k+1}), \\ z_i^{k+1} = 0 & \text{if } i \notin \mathcal{G}(x^{k+1}), \end{cases} \tag{7.23}$$

where we recall that the index set $\mathcal{G}(x^{k+1})$ contains at most s elements, corresponding to the not null components of x^{k+1} with the largest absolute value.

Note that $|\mathcal{G}(x^{k+1})| < s$ implies $\|x^{k+1}\|_0 < s$ and hence $z^{k+1} = x^{k+1}$, therefore we can write

$$-\tau_k(x_i^{k+1} - z_i^{k+1}) = 0 \begin{cases} \forall\, i \in \mathcal{G}(x^{k+1}) & \text{if } |\mathcal{G}(x^{k+1})| = s, \\ \forall\, i \in \{1,\dots,n\} & \text{if } |\mathcal{G}(x^{k+1})| < s. \end{cases} \tag{7.24}$$

There are finitely many possible sets $\mathcal{G}(x^{k+1})$, therefore at least one of them is repeated infinitely on K. Thus, let us assume that $K_1 \subseteq K$ is such that $\mathcal{G}(x^{k+1}) = I$ for all $k \in K_1$.

Now, let $\mathcal{G}^\star = \mathcal{G}(\bar{x})$. We can prove by similar reasonings as in the proof of Theorem 4.1 that $\mathcal{G}^\star \subseteq \mathcal{G}$.

We thus have the following three possible cases:

(i) $|\mathcal{G}| = s, \mathcal{G} = \mathcal{G}^\star$;

(ii) $|\mathcal{G}| < s$;

(iii) $|\mathcal{G}| = s, \mathcal{G} \supset \mathcal{G}^\star$.

We will address these three cases one at a time:

(i) We are in the case $\mathcal{G} = \mathcal{G}^\star = I_1(\bar{x})$. We know, recalling (7.21), that

$$\begin{aligned} 0 &\ge \min_{\substack{\|d\|\le 1\\ l\le x^{k+1}+d\le \mu\\ \|d_{\bar{\mathcal{G}}}\|_0=0\\ Ad=0}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d \\ &\ge \min_{\substack{\|d\|\le 1\\ l\le x^{k+1}+d\le \mu}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d \xrightarrow[k\in K_1]{k\to\infty} 0. \end{aligned} \tag{7.25}$$

In addition, recalling (7.24), we can write

$$\begin{aligned}
&\min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{\bar{\mathcal{G}}}\|_0=0\\ Ad=0}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d\\
&\quad= \min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{\bar{\mathcal{G}}}\|_0=0\\ Ad=0}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k(x^{k+1}-z^{k+1})\right)^T d + \tau_k(Ax^{k+1}-b)^T Ad\\
&\quad= \min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{\bar{\mathcal{G}}}\|_0=0\\ Ad=0}} \max_{j=1,\dots,m} \sum_{i=1}^{n} \nabla_i f_j(x^{k+1})d_i + \tau_k d_i(x_i^{k+1}-z_i^{k+1})\\
&\quad= \min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ Ad=0}} \max_{j=1,\dots,m} \sum_{i\in\mathcal{G}} \nabla_i f_j(x^{k+1})d_i = \min_{\substack{\|d\|\leq 1\\ l\leq x+d\leq \mu\\ \|d_{\bar{\mathcal{G}}}\|_0=0\\ Ad=0}} \max_{j=1,\dots,m} \nabla f_j(x^{k+1})^T d.
\end{aligned} \tag{7.26}$$

Substituting (7.26) in (7.25), recalling that $I_0(\bar{x}) = \bar{\mathcal{G}}$, Lemma 7.8 and that $\mathcal{F}(\bar{x})$ is given by (7.5) with $s - \|\bar{x}\|_0 = 0$, taking the limits we get that $\bar{x}$ is Pareto critical for problem and hence MOLZ conditions hold.

(ii) We are in the case $|\mathcal{G}| < s$, so $\mathcal{G} = I_1(x^{k+1})$ for all $k \in K_1$ and $\mathcal{G}^\star = I_1(\bar{x})$. We know from (7.24) that $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $i = 1, \dots, n$, for any $k \in K_1$. Similarly as above, we can write

$$\begin{aligned}
0 \geq &\min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{I_0(\bar{x})}\|_0\leq n-\|\bar{x}\|_0}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d\\
\geq &\min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d \xrightarrow[k\in K_1]{k\to\infty} 0
\end{aligned} \tag{7.27}$$

and using (7.24), after some algebraic manipulation, we have

$$\begin{aligned}
&\min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{I_0(\bar{x})}\|_0\leq n-\|\bar{x}\|_0\\ Ad=0}} \max_{j=1,\dots,m} \left(\nabla f_j(x^{k+1}) + \tau_k A^T(Ax^{k+1}-b) + \tau_k(x^{k+1}-z^{k+1})\right)^T d\\
&\qquad= \min_{\substack{\|d\|\leq 1\\ l\leq x^{k+1}+d\leq \mu\\ \|d_{I_0(\bar{x})}\|_0\leq n-\|\bar{x}\|_0\\ Ad=0}} \max_{j=1,\dots,m} \nabla f_j(x^{k+1})^T d.
\end{aligned}$$

Again, substituting in (7.27), taking the limits and recalling Lemma 7.8 we get that $\bar{x}$ is Pareto critical for problem (7.4) and therefore satisfies MOLZ conditions.

(iii) We are in the case $|\mathcal{G}| = s$, $\mathcal{G} \supset \mathcal{G}^\star$, $\mathcal{G} = \mathcal{G}(x^{k+1})$ for all $k \in K_1$, $\mathcal{G}^\star = I_1(\bar{x})$. We know from (7.24) that $\tau_k(x_i^{k+1} - z_i^{k+1}) = 0$ for all $i \in \mathcal{G}$. We can write again equations (7.25) and (7.26) and substitute the latter in the first. Taking the limits for $k \to \infty$, $k \in K_1$ and recalling Lemma 7.8 we get that $\bar{x}$ satisfies MOLZ conditions by selecting the super support set $J = \mathcal{G}$.

Putting everything together, we have from (i), (ii) and (iii) that MOLZ conditions are always satisfied at the limit point $\bar{x}$. □

Remark 7.3. Similarly as in Remark 4.3, it is easy to observe if there exists a subsequence $\hat{K} \subset K$ s.t. $\|x^k\|_0 = \|\bar{x}\|_0$ for all $k \in \hat{K}$ or $\|x^k\|_0 < s$ for all $k \in \hat{K}$, $\bar{x}$ is a Pareto-critical. This better characterization of the algorithm tells us that the limit points are Pareto critical in most cases.

7.6 Concluding Remarks

The primary aim of this Chapter was to lay the theoretical foundation for the analysis of sparse multi-objective optimization tasks and to propose an algorithmic approach to deal with this class of problems.

We have established first order necessary and sufficient optimality conditions for problems with sparsity and linear constraints. We have then defined a convergent Penalty Decomposition type method designed to properly tackle the considered class of problems. In particular, a multi-objective penalty function has been defined and sequentially optimized, for increasing values of the penalty parameter, employing an alternating minimization scheme where minimization w.r.t. the original variables is carried out by multi-objective descent.

Further work shall be focused on strategies to make the algorithm deal with sets of points and produce an approximation of the whole Pareto set, rather than a single solution, similarly to what has been done for other multi-objective descent type algorithms (Cocchi et al., 2020b, 2021; Custódio et al., 2011; Fliege and Vaz, 2016).

Chapter 8

Computational Experiments

In this Chapter, we present the results of a broad set of computational experiments designed to evaluate the efficiency and the effectiveness of the algorithms introduced in the previous chapters.

We put a special focus in emphasizing how the theoretical properties proved for Algorithms 2, 5, 7, 8 translate into actual experimental benefits when these methods are employed in practice.

8.1 Benchmark

Before turning to the experimental setting and the results, we shall introduce the test problems used for the comparisons.

Sparse Logistic Regression Problem

The problem of *sparse logistic regression* (Hastie et al., 2009; Civitelli et al., 2021) has important applications, for instance, in machine learning (Weston et al., 2003; Bach et al., 2012). Given a dataset having N samples $\{r^1, \dots, r^N\}$, with n features and N corresponding labels $\{t_1, \dots, t_N\}$ belonging to $\{-1, 1\}$, the sparse logistic regression problem can be formulated as follows

$$\min_{w} \; L(w) = \sum_{i=1}^{N} \log\left(1 + \exp\left(-t_i(w^T r^i)\right)\right) \qquad \text{s.t.} \qquad \|w\|_0 \leq s. \tag{8.1}$$

The benchmark we consider is made up of 18 problems of the form (8.1), obtained as described hereafter. We employed 6 binary classification datasets, listed in Table 8.1. All the datasets are from the UCI Machine Learning Repository (Dua and Graff, 2017).

For each dataset, we removed data points with missing variables; moreover, we one-hot encoded the categorical variables and standardized the other ones to zero mean and unit standard deviation. For every dataset, we chose 3 different values of s, specified later in this Chapter, in order to define 3 different problems of the form (8.1).

Table 8.1: List of datasets used for experiments on sparse logistic regression.

Dataset	N	n	Abbreviation
Heart (Statlog)	270	25	`heart`
Breast Cancer Wisconsin (Prognostic)	194	33	`breast`
QSAR Biodegradation	1055	41	`biodeg`
SPECTF Heart	267	44	`spectf`
Spambase	4601	57	`spam`
Adult a2a	2265	123	`a2a`

Neural Networks Compression Problem

Artificial neural networks are typically highly overparameterized models; in applications, however, it is useful to employ smaller networks, for the sake of both prediction speed and memory usage. An important task that has had a renewed interest in machine learning is thus that of pruning an already trained neural network (Reed, 1993).

In recent years, this task has been tackled by reformulating the neural network compression problem as a cardinality constrained problem (Carreira-Perpinán and Idelbayev, 2018):

$$\begin{aligned} \min_w \; & L(w) \\ \text{s.t. } & \|w\|_0 \leq s, \end{aligned} \tag{8.2}$$

where L is the loss function employed to train the net. The pruning operation is obtained by warm-starting problem (8.2) with the trained parameters of the network.

In our experiments, we designed a simple feed-forward neural network to perform classification on the MNIST dataset (LeCun and Cortes, 2010). Input images have been fed to the network as one-dimensional vectors of length $28 \times 28 = 784$; the network architecture consists of a single hidden layer of 32 neurons and the 10 output neurons, for a total of 25450 trainable parameters. We used the sigmoid function $\sigma(t) = 1/(1 + \exp(-t))$ as the activation function for the hidden units, while the output layer performs the softmax operation. We initially trained the network with the classical ADAM optimizer (Kingma and Ba, 2014), run for 200 epochs, obtaining a network having a test accuracy of 92%. The loss function is the softmax cross-entropy function (Goodfellow et al., 2016).

Sparse Portfolio Selection Problem

We consider the mean/variance portfolio selection problem (Markowitz, 1952, 1994), which has been so relevant to the financial community that it has driven enormous attention from researchers in both fields of operations research and economics. The problem, in the original Markovitz formulation, is given by

$$\begin{aligned} \min_{x\in\mathbb{R}^n}\ & \frac{\psi}{2}x^TQx - c^Tx \\ \text{s.t. } & x \geq 0, \quad e^Tx = 1, \end{aligned} \tag{8.3}$$

where $x \in \mathbb{R}^n$ is the vector of decision variables, being x_i the fraction of the available capital to be invested into asset i, $c \in \mathbb{R}^n$ is the vector of expected returns and $Q \in \mathbb{R}^{n\times n}$ is the positive semi-definite variance-covariance matrix.

To improve the realism of the model, several constraints have been proposed to be added to problem (8.3). Among them, one of the most relevant is certainly the cardinality constraint $\|x\|_0 \leq s$ (Bienstock, 1996; Bertsimas and Shioda, 2009; Bertsimas and Cory-Wright, 2018). The sparsity requirement is particularly relevant not only because managers pay monitoring costs for non-zero positions, but also because investors hardly trust portfolio managers who do not control the number of positions held. Problem (8.3) with the cardinality constraint can be reformulated as a MIQP optimization problem by means of the introduction of binary indicator variables z_i, $i = 1, \dots, n$, that model whether $x_i = 0$ or not and the replacement of $\|x\|_0 \leq s$ by $\sum_{i=1}^n z_i \leq s$.

The objective function of (8.3) is the sum of two terms that represent in fact two distinct, contrasting goals. Indeed, the underlying financial problem is inherently a bi-objective optimization problem, which is scalarized for a simpler optimization process. The pure multi-objective formulation has occasionally been considered in the literature (Armananzas and Lozano, 2005; Radziukynienė and Žilinskas, 2008; Chen and Wei, 2019), even with the additional cardinality constraint (Chiam et al., 2008; Xidonas et al., 2018; Tian et al., 2019), leading to the problem

$$\begin{aligned} \min_{x\in\mathbb{R}^n}\ & \left(\frac{\psi}{2}x^TQx,\ c^Tx\right)^T \\ \text{s.t. } & x \geq 0, \quad e^Tx = 1, \quad \|x\|_0 \leq s. \end{aligned} \tag{8.4}$$

We chose problem (8.4) as a benchmark for the multi-objective setting because we can easily obtain a basis for comparison; indeed, the global solution of problem (8.3) with the cardinality constraint constitutes, for every value of ψ, a Pareto optimal point for problem (8.4). Hence, we can obtain a set of reference Pareto points by solving with a MIQP solver the scalarized problem for several values of the trade-off parameter ψ.

The data used in the experiments consists of daily data for securities from the FTSE 100 index, from 01/2003 to 12/2007. The three datasets are referred to as DTS1, DTS2, and DTS3, and are formed by 12, 24, and 48 securities, respectively. We also included three datasets from the Fama/French benchmark collection (FF10, FF17, and FF48, with cardinalities 10, 17, and 48), using the monthly returns from 07/1971 to 06/2011. The datasets are generated as by Brito and Vicente (2013) and Cocchi et al. (2020a). For each dataset, we built two instances of problem (8.4), with with two different values of s.

8.2 Comparison of PD Approaches: Convex Case

The purpose of this first block of experiments is to evaluate the proposed inexact minimization strategy for the PD approach (both in its gradient-based and derivative-free versions), compared with the exact minimization approach of the original algorithm. To this aim, we consider the problem of sparse logistic regression, where the objective function is convex, but the solution of the subproblems in the x variables cannot be obtained in closed form, i.e., it requires the adoption of an iterative method.

Implementation Details

Algorithms 1, 2 and 5 have been implemented in Python 3.6. We used as test benchmark the set of 18 sparse logistic regression problems described in Section 8.1 with s corresponding to the 25%, 50% and 75% of the number n of features of each dataset.

The algorithms start from the feasible initial point $x^0 = z^0 = 0 \in \mathbb{R}^n$. Their common parameters have been set as follows: $\tau_0 = 1$ and $\theta = 1.1$. The three algorithms differ only in the x-minimization step. Concerning the line search parameters of Algorithm 2, we set $\gamma = 10^{-5}$ and $\beta = 0.5$. As for the derivative-free Algorithm 5, we set $\delta = 0.5, \gamma = 10^{-5}, \sigma = 2$.

The x-minimization step for Algorithm 1 has been performed by the BFGS (Bertsekas, 1997) solver included in the `scipy` library (Virtanen et al., 2020). In particular, the inner iterates of the BFGS solver have been stopped whenever the current point $u^{\ell+1}$ is such that $\|\nabla_x q_{\tau_k}(u^{\ell+1}, v^\ell)\| \leq 10^{-5}$, i.e., when the current point is a good approximation of a stationary point and hence, being the penalty function q_{τ_k} strictly convex with respect to u, of the global minimizer.

For a fair comparison, we employ for the three PD procedures the same stopping criteria for the outer and the inner loop. Specifically, we used the practical stopping criteria proposed by Lu and Zhang (2013): the inner loop stops when the decrease of the value of the function q_{τ_k} is sufficiently small, i.e., when

$$q_{\tau_k}(u^\ell, v^\ell) - q_{\tau_k}(u^{\ell+1}, v^{\ell+1}) \leq \epsilon_{\text{in}}, \tag{8.5}$$

where $\epsilon_{\text{in}} = 10^{-4}$; the outer loop is stopped when x and z are sufficiently close, i.e., as soon as

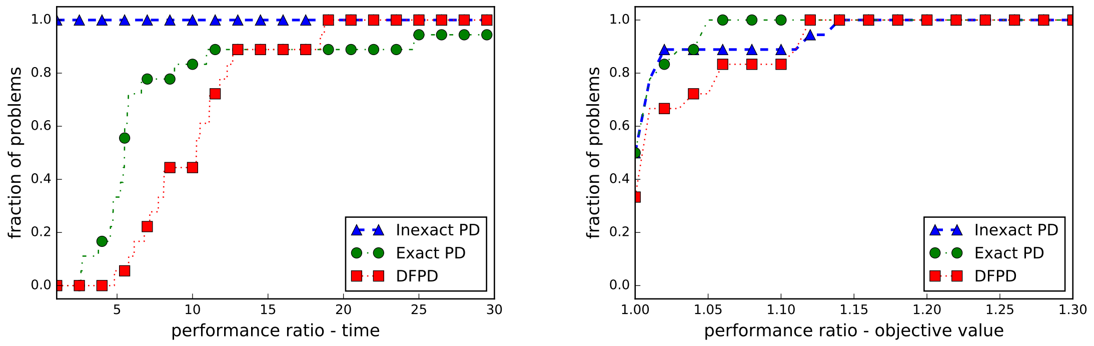
$$\|x^{k+1} - z^{k+1}\| \leq \epsilon_{\text{out}}, \tag{8.6}$$

where $\epsilon_{\text{out}} = 10^{-4}$.

All the experiments have been carried out on an Intel(R) Core(TM) i7- 6700 CPU @ 3.40GHz machine with 4 physical cores (8 threads) and 16 GB RAM.

Numerical Results

The three algorithms, Algorithm 1 called Exact PD, Algorithm 2 called Inexact PD, and Algorithm 5 called DFPD, have been compared using the performance profiles (Dolan and Moré, 2002). We recall that, in performance profiles, each curve represents, given a performance metric, the cumulative distribution of the ratio between the result obtained by a solver on an instance of a problem and the best result obtained by any considered solver on that instance. The results of the comparison are shown in Figure 8.1.



(a) Runtime

(b) Objective value

Figure 8.1: Performance profiles of runtime (a) and attained objective value (b) for the Exact, Inexact and Derivative-Free Penalty Decomposition algorithms, on 18 sparse logistic regression problems.

From the results in Figure 8.1b, we can observe that the performances of the three algorithms, in terms of attained objective function values, are quite close, with rather slight fluctuations. It's worth remarking that different local minima can be attained by different algorithms, even for equal starting points, because of the nonconvex nature of problem (8.1).

On the other hand, as shown in Figure 8.1a, the inexact version of the PD algorithm clearly outperforms the other two algorithms in terms of efficiency. This aspect can be valuable in connection with a global optimization strategy, where many local minimizations have to be performed and the availability of an efficient local solver may be useful.

The derivative-free algorithm is about an order of magnitude slower than its direct gradient-based counterpart, which is reasonable, considering that the size of the considered problems is quite large in the perspective of derivative-free optimization. In fact, the difference between the speed of gradient-based and derivative-free methods on problems with relatively large size is usually even larger; here, this gap is mitigated, since there is a large set of instructions shared by all the versions of the algorithm.

On the whole, this computational experience confirms the validity of the proposed approach. We remark that we tested the simplest implementation of the proposed algorithm, that is, performing, in the x-minimization step, a single line search along the steepest direction. Benefits, in terms of attained function values, could be obtained by performing more iterations of a descent method and by introducing a suitable inner stopping criterion. As already observed, this can be done to improve the effectiveness of the algorithm preserving its global convergence properties.

8.3 Comparison of PD Approaches: Nonconvex Case

In this Section, we turn to the evaluation of the proposed variant of the PD approach in the nonconvex setting, which was one of the main motivations for the introduction of the inexact approach. We do so with a test problem representing an actual application of the Penalty Decomposition method in the literature.

Specifically, we performed the pruning of the neural network described in Section 8.1 with the Penalty Decomposition strategy, as suggested by Carreira-Perpinán and Idelbayev (2018). Here, we do not consider the derivative-free version of the algorithm, since the dimensionality of the considered problem is too high.

We would like to stress that here we are interested in evaluating the performance of the algorithm from an optimization perspective, i.e., in terms of training speed and value of the training loss, not in terms of the prediction performance of the obtained models.

Implementation Details

We repeat the experiment for 11 different values of s. Clearly, due to the nonlinearity of the objective function, the exact minimization at the x-update step is in fact not viable; in the work of Carreira-Perpinán and Idelbayev (2018), this issue is overlooked and a local optimizer is employed, therefore the theoretical guarantees of the approach come from the novel analysis from Chapter 4.

Our interest lies in comparing the performance of Algorithm 2 when the x update is performed by means of a single descent step or by running the gradient descent algorithm until a stationary point is reached. We compare the performance of the two variants of the PD procedure in terms of runtime and attained objective value.

The network has been implemented and trained using the `Tensorflow 1.14` library. Note that for both PD algorithms we perform gradient descent steps using the full batch gradient, i.e., no SGD strategy is employed.

As for the stopping criterion, we use (8.5) and (8.6) for both algorithms, with $\epsilon_{\text{in}} = 10^{-4}$ and $\epsilon_{\text{out}} = 10^{-5}$. For the complete gradient descent version of the algorithm, we stop the x-update process if the gradient is sufficiently small, i.e., $\|\nabla_x q_{\tau_k}(u^{\ell+1}, v^\ell)\| \leq 10^{-1}$ or if the decrease in the value of q_{τ_k} is smaller than 10^{-3}.

Concerning the other parameters, we set $\tau_0 = 1$, $\theta = 1.1$, $\gamma = 10^{-5}$ and $\beta = 0.5$. In order to avoid numerical issues, we normalize the descent direction if the norm of the gradient is larger than 1.

The experiments have been carried out on an Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz machine with 4 physical cores (8 threads) and 16 GB RAM.

Numerical Results

We show the results of the experiment, in terms of runtime, in Figure 8.2. We can observe that performing a single descent step allows us to save a significant amount of time, for all the tested values of s.

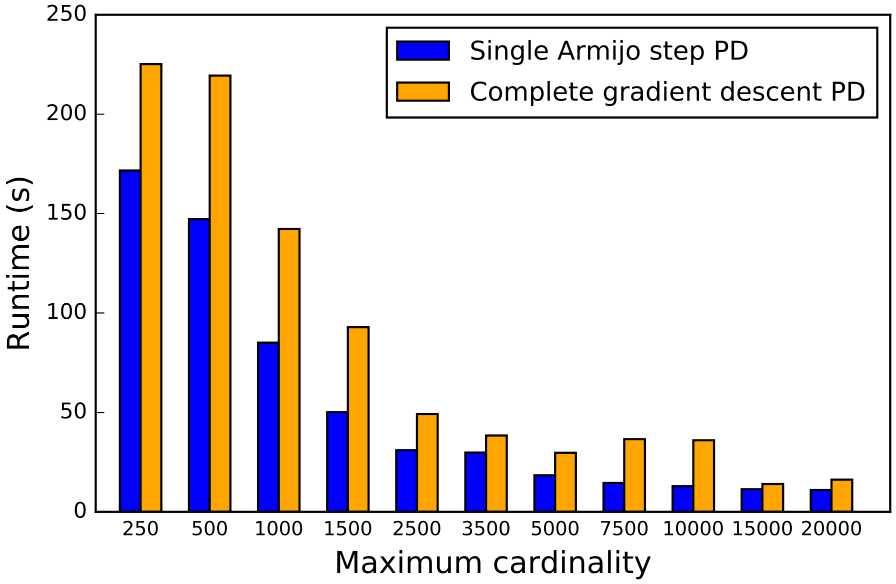


Figure 8.2: Comparison of the performance, in terms of runtime, of two variants of the inexact PD algorithm on neural network compression problems.

We do not show the results concerning the obtained objective values, as in every instance of the experiment the two methods attained the same value, up to negligible differences in the order of 10^{-2}.

8.4 Sparse Neighborhood Search Performance

Concerning the SNS Algorithm 7, we are particularly interested from a computational point of view in studying two relevant aspects. Specifically, here we want to:

- analyze the benefits and the costs of increasing the size of the neighborhood;
- assess the performance of the proposed approach, compared to the the Greedy Sparse-Simplex (GSS) method and the Penalty Decomposition (PD) approach (in the original, exact version).

To these aims, we considered again the problem of sparse logistic regression, where the objective function is continuously differentiable and convex, but the solution of the problem for a fixed support set requires the adoption of an iterative method. Note that we preferred to consider a problem without other constraints in addition to the sparsity one, in order to simplify the analysis of the behavior of the proposed algorithm.

Implementation details

Algorithms SNS, PD and GSS have been implemented in Python 3.7, mainly exploiting libraries `numpy` and `scipy`. The convex subproblems of both PD and GSS have been solved up to global optimality by using the L-BFGS algorithm (in the implementation from Liu and Nocedal 1989, provided by `scipy`). We also employed L-BFGS for the local optimization steps in SNS.

All algorithms start from the feasible initial point $x^0 = 0 \in \mathbb{R}^n$. For the PD algorithm, we set the starting penalty parameter to 1 and its growth rate to 1.05. The algorithm stops when $\|x^k - z^k\| < 0.0001$, as suggested by Lu and Zhang (2013). As for the GSS, we stop the algorithm as soon as $\|x^{k+1} - x^k\| \leq 0.0001$.

Concerning our proposed Algorithm 7, we take into account four versions, employing the neighborhood $\mathcal{N}_\rho$ defined in Definition 2.14 with radius $\rho \in \{1, 2, 3, 4\}$. The parameters have been set as follows:

- $\xi = 10^3$,
- $\theta = 0.5$,
- $\eta_0 = 10^{-5}$.

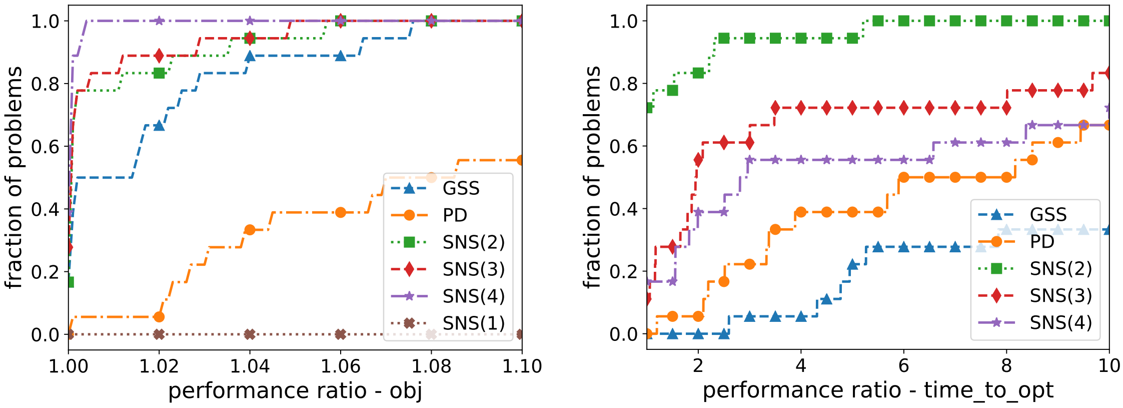
For what concerns μ_0 and δ, we actually keep the value of μ fixed to 10^{-6}. We again employ the stopping criterion $\|x^{k+1} - x^k\| \leq 0.0001$.

For all the algorithms, we have also set a time limit of 10^4 seconds. All the experiments have been carried out on an Intel(R) Xeon E5-2430 v2 @2.50GHz CPU machine with 6 physical cores (12 threads) and 16 GB RAM.

Numerical Results

As benchmark for our experiments, we considered the 18 sparse logistic regression problems from Section 8.1 with s set equal to to 3, 5 and 8 in (8.1). For SNS and GSS we consider the computational time employed to find the best solution.

In Figure 8.3 the performance profiles (Dolan and Moré, 2002) w.r.t. the objective function values and the runtimes (intended as the time to find the best solution) attained by the different algorithms are shown. We do not report the runtime profile of SNS(1) since it is much faster than all the other methods and thus would dominate the plot, making it poorly informative. We can however note that unfortunately its speed is outweighed by the very poor quality of the solutions.



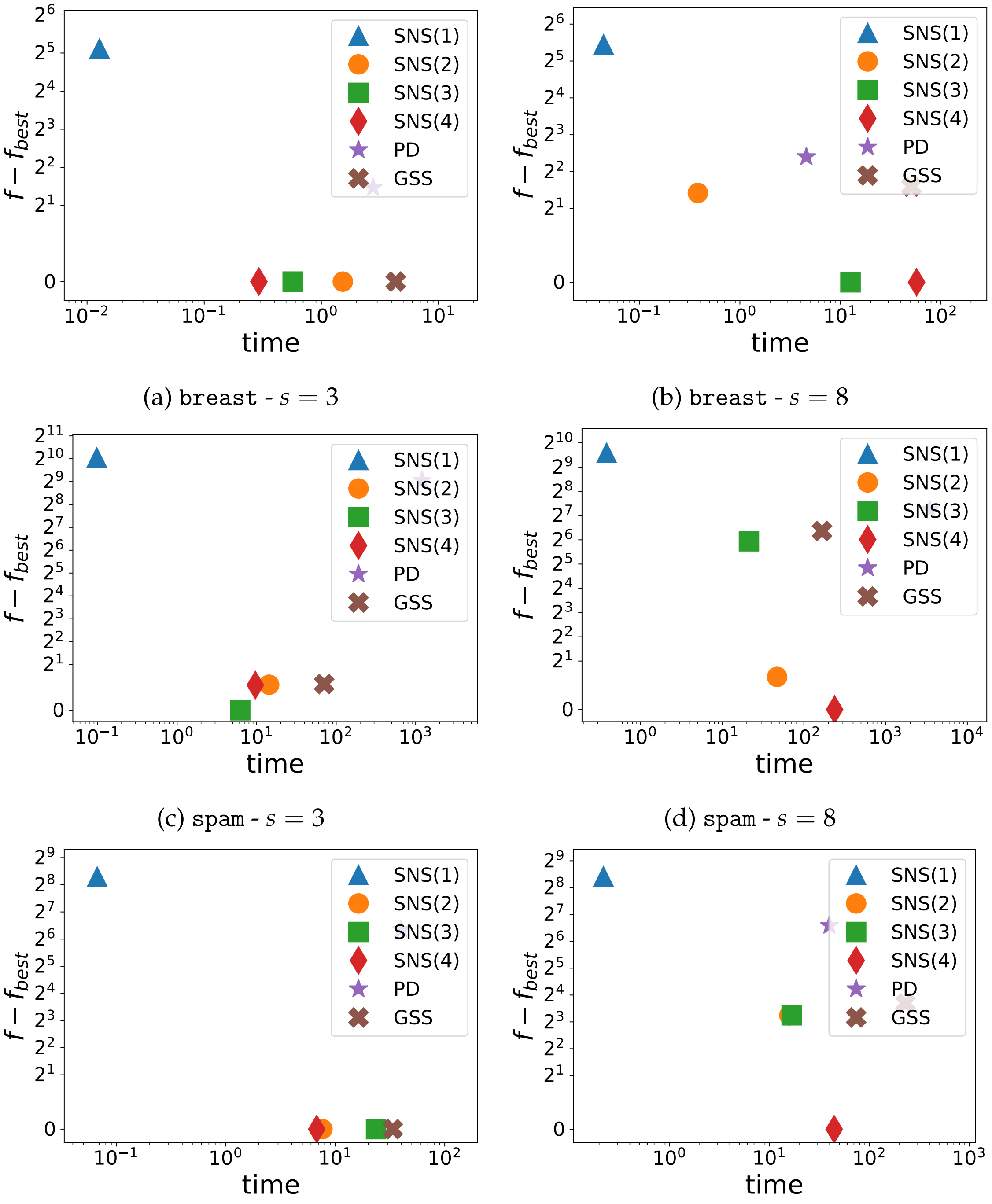
(a) objective value

(b) Objective value

Figure 8.3: Performance profiles for the considered algorithms on 18 sparse logistic regression problems.

We can observe that increasing the size of the neighborhood consistently leads to higher quality solutions, even though the computational cost grows. We can see that SNS (with a sufficiently large neighborhood) has better performances than the other algorithms known from the literature; in particular, while the neighborhood radius $\rho = 1$ only allows to perform forward selection, with poor outcomes, $\rho \geq 2$ makes swap operations possible, with a significant impact on the exploration capabilities.

The GSS has worse quality performance than SNS(2), which is reasonable, since its move set is actually smaller and optimization is always carried out w.r.t. a single



(a) breast - $s = 3$

(b) breast - $s = 8$

(c) spam - $s = 3$

(d) spam - $s = 8$

(e) a2a - $s = 3$

(f) a2a - $s = 8$

Figure 8.4: Quality/cost trade-off for the algorithms on sparse logistic regression problems from datasets breast, spam and a2a.

variable and not the entire active set. However, it proved to also be slower than the SNS, mostly because of two reasons: it always tries all feasible moves, not necessarily accepting the first one that provides an objective decrease, and it requires many more iterations to converge, since it considers one variable at a time.

Finally, the PD method appears not to be competitive from both points of view: it is slow at converging to a feasible point and it has substantially no global optimization features that could guide to globally good solutions.

It is interesting to remark how considering larger neighborhoods appears to be particularly useful in problems where the sparsity constraint is less strict and thus combinatorially more challenging. As an example, we show the runtime-objective tradeoff for the `breast`, `spam` and `a2a` problems for $s = 3$ and $s = 8$ in Figure 8.4. We can observe that for $s = 3$, SNS finds good, similar solutions for either $\rho = 2, 3$ or 4, with a similar computational cost. On the other hand, as s grows to 8, using $\rho = 4$ allows to significantly improve the quality of the solution without a significant increase in terms of runtime.

8.5 Effectiveness of the Multi-objective PD Scheme

As we have anticipated, we chose problem (8.4) to test the proposed Algorithm 8 since we can easily obtain a basis for comparison. It is worth emphasizing that MOSPD could however be employed with much more general problems, whereas MIP modeling to solve the scalarized problem becomes impractical as soon as the objective function gets nonconvex or more than quadratic.

We are interested in the quality of the solutions retrieved by MOSPD on sparse portfolio selection problems, figuring that the performance on this class of problems is an indicator of its behavior in more general cases.

Implementation Details

The scripts for the experiments have been written in Python3. MOSPD solver makes use of the `numpy` library (Oliphant, 2006); the LP problem for the computation of descent directions is solved with `Gurobi 9.0.0` (Gurobi Optimization, 2020).

For the parameters of the algorithm, we set $\tau_0 = 10$, $\sigma = 1.1$, $\varepsilon_0 = 10^{-5}$ for DTS problems and $\varepsilon_0 = 10^{-3}$ for FF problems, $\varepsilon_{k+1} = \max\{0.8\varepsilon_k, \varepsilon_0/100\}$, $\beta = 10^{-5}$; the algorithm is stopped when $\|x^{k+1} - z^{k+1}\| \leq 10^{-3}$ and $|1 - e^T z| < 10^{-4}$; as final solution we retain vector $\bar{z}$ which strictly satisfies, from a numerical point of view, the cardinality constraint; we employ the MOPGD algorithm to refine the solution $\bar{z}$ returned by MOSPD; the MOPGD procedure is run starting at $\bar{z}$ and keeping fixed the zero variables, so that the cardinality constraint is implicitly handled.

We remark that, since the problem has a bounded feasible set, the restart strategy at lines 6-9 of Algorithm 8 is not necessary. The MIQP scalarized problem is solved with `Gurobi 9.0.0`.

The numerical comparison is carried out by running multiple times MOSPD and the scalar MIQP solver. MOSPD is started from $n+1$ different initial solutions: all points with one component equal to 1 and all others set to zero and the vector with all components equal to $1/n$; the scalarization approach is again performed $n+1$ times, for values of λ in $\{2^j \mid j = i - \lfloor \frac{n}{2} \rfloor,\ i = 0, \dots, n\}$.

Numerical Results

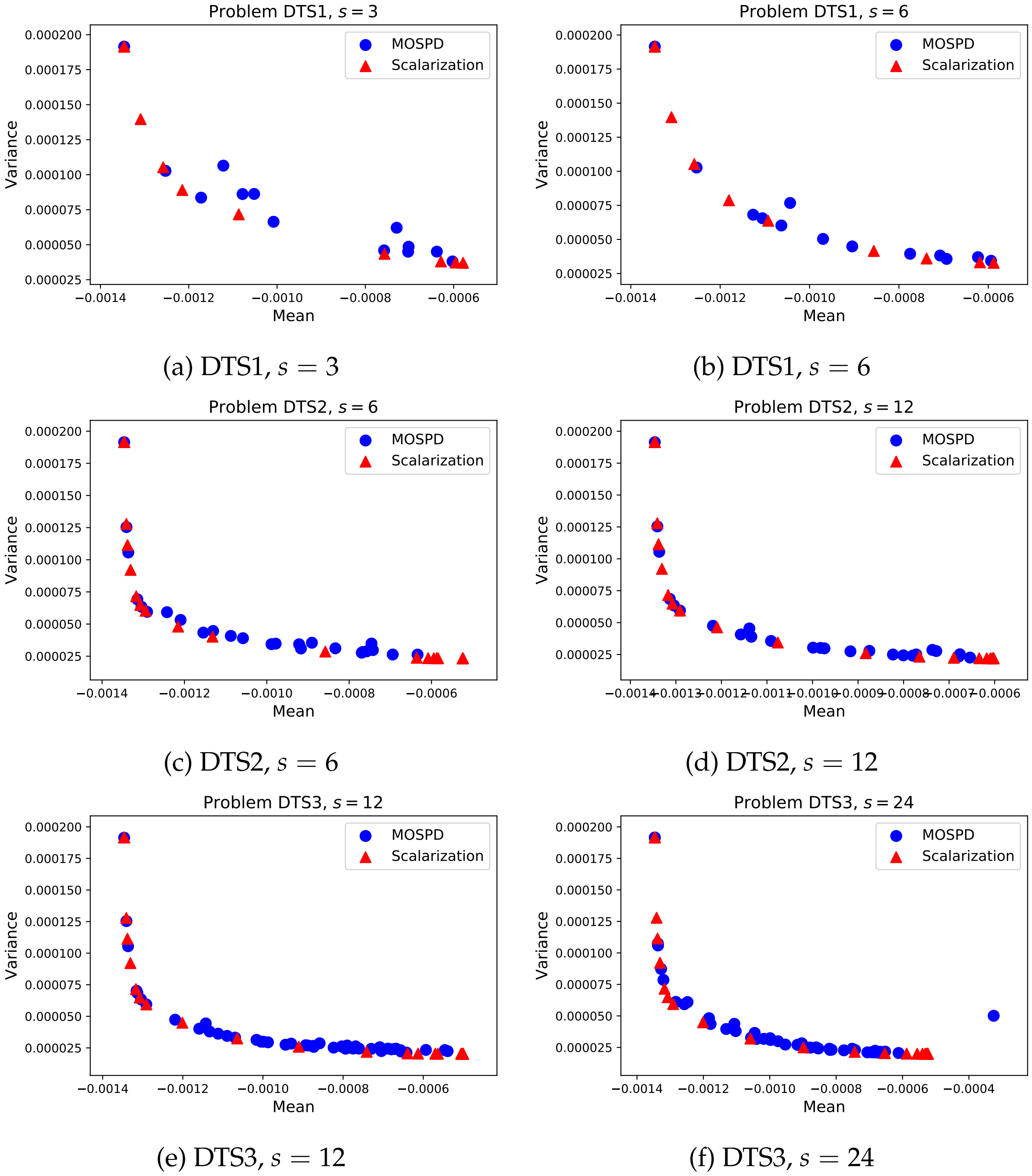
The results of the experiment are shown in Figures 8.5 and 8.6. We can observe that MOSPD is able to retrieve, in most cases, a good quality approximation of the Pareto front, using the one produced by the scalarization method as reference. We can also see that some runs of MOSPD stop at dominated, and hence not Pareto optimal, solutions, which is not surprising: as previously remarked, while scalarization, if solved to optimality, is guaranteed to produce a Pareto optimzer, MOSPD is only proved to generate a point satisfying MOLZ conditions. Besides, it is interesting to note that the set of optimal solutions produced by MOSPD is generally more diversified than that obtained by scalarization.

In order to quantitatively characterize the above qualitative considerations, we employ the popular metrics defined by Custódio et al. (2011): purity, Δ-spread and Γ-spread. We recall that the purity metric measures the quality of the generated front, i.e., how good the non-dominated points computed by a solver are with respect to those computed by any other solver. Here, a higher value is a better value. On the other hand, the spread metrics are essential to measure the uniformity of the generated front in the objectives space. Particularly, the Γ-spread metric is defined as the maximum ℓ_∞ distance between adjacent points in the retrieved Pareto front. The Δ-spread metric is quite similar to the standard deviation of the ℓ_∞ distances between adjacent points in the retrieved Pareto front.

We report the performance obtained by the two considered methods on the 12 test problems in Table 8.2. We can see that the values of purity, Γ-spread and Δ-spread support the visual impression that MOSPD produces approximate Pareto fronts whose points are better distributed (better spread values), even if some solutions are dominated (lower purity).

In conclusion, the set of solutions retrieved by MOSPD is, once the dominated solutions are filtered out, comparable to the one obtained by the scalarization method, while it also allows the user to choose, a posteriori, among a more diverse spectrum of optimal solutions, being the points well distributed along the front.

We would like to stress again that the discussed experiment is aimed at assessing the performance of MOSPD in a case where a valid alternative is available; with more complex problems, the scalarization approach is no more viable, while we expect MOSPD to reproduce the good performance achieved on the portfolio selection problem.



(a) DTS1, $s = 3$

(b) DTS1, $s = 6$

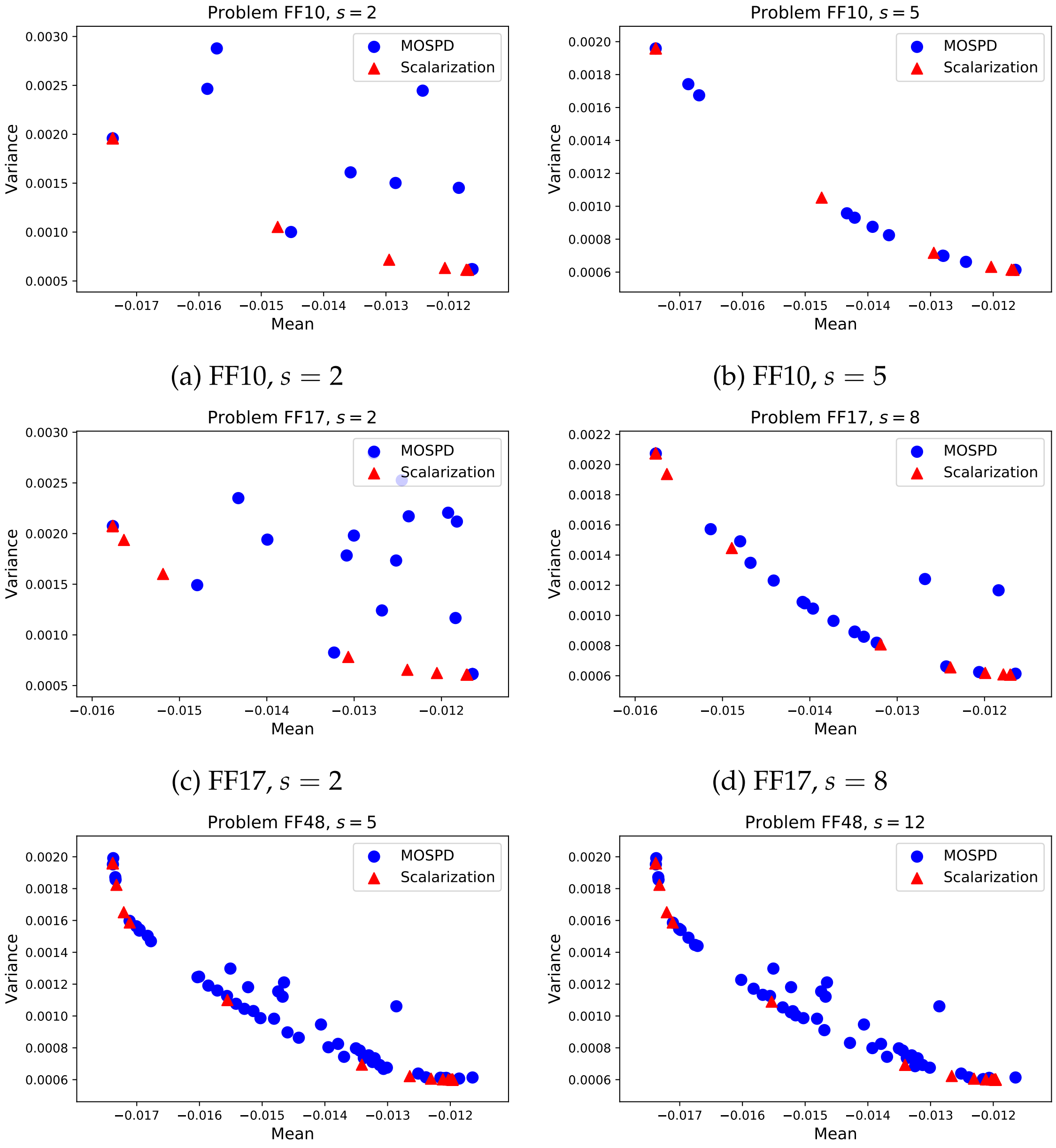
(c) DTS2, $s = 6$

(d) DTS2, $s = 12$

(e) DTS3, $s = 12$

(f) DTS3, $s = 24$

Figure 8.5: Results of the tests on DTS problems.



(a) FF10, $s = 2$

(b) FF10, $s = 5$

(c) FF17, $s = 2$

(d) FF17, $s = 8$

(e) FF48, $s = 5$

(f) FF48, $s = 12$

Figure 8.6: Results of the tests on FF problems.

Table 8.2: Performance metrics (Purity, Γ-spread and Δ-spread) obtained by Pareto front approximations produced by multi-start versions of MOSPD and of the scalarization approach.

Problem	s	**Algorithm**	**Purity**	**Γ-spread**	**Δ-spread**
DTS1	3	MOSPD	0.714	**0.00025**	0.824
		Scalarization	**0.888**	0.00033	**0.776**
	6	MOSPD	0.818	**0.00012**	1.168
		Scalarization	**0.888**	0.00023	**0.701**
DTS2	6	MOSPD	0.823	**0.00014**	**1.102**
		Scalarization	**0.944**	0.00027	1.157
	12	MOSPD	0.894	**0.00010**	1.151
		Scalarization	**0.944**	0.00019	**1.060**
DTS3	12	MOSPD	0.785	**0.00014**	**1.279**
		Scalarization	**0.966**	0.00016	1.318
	24	MOSPD	0.875	**0.00008**	**1.311**
		Scalarization	**0.966**	0.00015	1.324
FF10	2	MOSPD	**1.0**	0.0028	**0.424**
		Scalarization	**1.0**	**0.0026**	1.046
	5	MOSPD	**1.0**	**0.00235**	**0.988**
		Scalarization	0.833	0.00264	1.046
FF17	2	MOSPD	0.5	**0.00158**	**0.910**
		Scalarization	**1.0**	0.00212	1.173
	8	MOSPD	0.866	**0.00079**	**0.740**
		Scalarization	**1.0**	0.00170	1.146
FF48	5	MOSPD	0.8	**0.00074**	**0.742**
		Scalarization	**1.0**	0.00214	1.523
	12	MOSPD	0.903	**0.00069**	**0.812**
		Scalarization	**1.0**	0.00212	1.522

Chapter 9

Conclusions

In this thesis work, we have dealt with mathematical optimization problems with sparsity constraints. Specifically, emphasis was put on the cases where the objective function is nonconvex and/or the number of variables is rather high, so that the problem has to be tackled in terms of a continuous optimization problem.

In this context, we put order, from a theoretical point of view, among various well-known, as well as novel, necessary conditions of optimality for this class of problems. Then, we addressed a variety of algorithms designed to produce, in practice, solutions satisfying these conditions. In particular, we proposed tailored algorithms for complex settings such as the nonconvex, the derivative-free and the multi-objective one. Moreover, we introduced a completely new algorithmic scheme that, taking into account the combinatorial, discrete nature of the problem, is able to obtain the highest possible theoretical guarantees and also has remarkable exploration capabilities in a global optimization perspective.

We finally showed by a diverse set of computational experiments that the proposed approaches actually exhibit good performance in practice.

On the basis of the results of the present dissertation, new avenues of research open up; in particular, topics for future work include:

- the conception and the analysis of an inexact version of the GSS algorithm for the nonconvex case;
- the extension of the inexact PD scheme to the case with additional constraints;
- the study of optimality conditions for sparsity constrained problems in the nonsmooth setting;
- the design of a tailored algorithm (reasonably PD-type) for the nonsmooth setting;
- the extension of the SNS algorithm to the derivative-free and the multi-objective settings;

- the extension of the theoretical analysis in the multi-objective setting to the case with general additional constraints.

Appendix A

On the Relationship Between Stationarity Conditions and KKT Conditions

Consider the continuous optimization problem

$$\begin{aligned} \min_x \ & f(x) \\ \text{s.t. } & x \in X, \end{aligned} \tag{A.1}$$

where $X = \{x \in \mathbb{R}^n \mid h(x) = 0,\ g(x) \leq 0\}$ is a convex set (h_i, $i = 1, \dots, p$ are affine functions, g_i, $i = 1, \dots, m$, are convex functions). We assume f and g to be continuously differentiable; h is differentiable, being affine.

Definition A.1. A point $x^\star \in X$ is a *stationary point* for problem (A.1) if, for any direction d feasible at $x^\star$, we have

$$\nabla f(x^\star)^\top d \geq 0.$$

It can be shown that a point $x^\star$ is stationary for problem (A.1) if and only if

$$x^\star = \Pi_X[x^\star - \nabla f(x^\star)], \tag{A.2}$$

where Π_X denotes as usual the orthogonal projection operator. Stationarity is a necessary condition of optimality for problem (A.1). It is possible to show that a point satisfying the KKT conditions is always a stationary point. Vice versa is true by stronger assumptions on the set of feasible directions.

Proposition A.1. *Let $x^\star \in X$ satisfy KKT conditions for problem* (A.1)*. Then, $x^\star$ is stationary for problem* (A.1).

Proof. Assume $x^\star$ satisfies KKT conditions with multipliers λ and μ. Let d be any feasible direction at $x^\star$. Since X is convex, we know that:

$$\nabla h_i(x^\star)^\top d = 0 \quad \forall i = 1, \dots, p, \tag{A.3}$$

$$\nabla g_i(x^\star)^\top d \leq 0 \quad \forall i : g_i(x^\star) = 0. \tag{A.4}$$

Moreover, from KKT conditions we know that

$$\lambda_i = 0 \quad \forall\, i : g_i(x^\star) < 0. \tag{A.5}$$

We know that

$$\nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{m} \mu_i \nabla h_i(x^\star) = 0,$$

hence

$$\left(\nabla f(x^\star) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star) + \sum_{i=1}^{p} \mu_i \nabla h_i(x^\star) = 0 \right)^\top d = 0,$$

and then

$$\nabla f(x^\star)^\top d + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^\star)^\top d + \sum_{i=1}^{m} \mu_i \nabla h_i(x^\star)^\top d = 0.$$

From equations (A.3) and (A.5), we get

$$\nabla f(x^\star)^\top d + \sum_{i: g_i(x^\star)=0} \lambda_i \nabla g_i(x^\star)^\top d = 0,$$

thus, recalling (A.4) and $\lambda \geq 0$,

$$\nabla f(x^\star)^\top d = - \sum_{i: g_i(x^\star)=0} \lambda_i \nabla g_i(x^\star)^\top d \geq 0.$$

Since d is an arbitrary feasible direction, we get the thesis. □

Proposition A.2. *Let $x^\star \in X$ be a stationary point for problem* (A.1)*. Assume that one of the following conditions holds:*

(i) the set of feasible directions $D(x^\star)$ is such that

$$D(x^\star) = \{ d \in \mathbb{R}^n \mid \nabla g_i(x^\star)^\top d \leq 0 \, \forall i : g_i(x^\star) = 0, \nabla h_i(x^\star)^\top d = 0 \; \forall i = 1, \dots, p \}$$

(ii) the set of feasible directions $D(x^\star)$ is such that

$$D(x^\star) = \{ d \in \mathbb{R}^n \mid \nabla g_i(x^\star)^\top d < 0 \, \forall i : g_i(x^\star) = 0, \nabla h_i(x^\star)^\top d = 0 \; \forall i = 1, \dots, p \}$$

and a constraint qualification holds.

Then, $x^\star$ is a KKT point.

Proof. We prove the two cases separately:

(i) Let $x^\star$ be a stationary point. Then, there does not exist a direction $d \in D(x^\star)$ such that
$$\nabla f(x^\star)^\top d < 0.$$
This implies that the system
$$\begin{array}{lll} \nabla f(x^\star)^\top d & < 0 & \\ \nabla g_i(x^\star)^\top d & \leq 0 & i : g_i(x^\star) = 0 \\ \nabla h_i(x^\star)^\top d & \leq 0 & i = 1, \ldots, p \\ -\nabla h_i(x^\star)^\top d & \leq 0 & i = 1, \ldots, p \end{array}$$
does not admit solution. By Farkas' Lemma we get the thesis.

(ii) Let $x^\star$ be a stationary point. Then, there does not exist a direction $d \in D(x^\star)$ such that
$$\nabla f(x^\star)^\top d < 0.$$
This implies that the system
$$\begin{array}{lll} \nabla f(x^\star)^\top d & < 0 & \\ \nabla g_i(x^\star)^\top d & < 0 & i : g_i(x^\star) = 0 \\ \nabla h_i(x^\star)^\top d & = 0 & i = 1, \ldots, p \end{array}$$
does not admit solution. By Motzkin's theorem we get that $x^\star$ satisfies the Fritz-John conditions and hence, by assuming a constraint qualification, the thesis is proved.

□

Condition (i) of Proposition A.2 holds if the functions g_i, $i = 1, \ldots, m$ and h_j, $j = 1, \ldots, p$ are affine.

Condition (ii) of Proposition A.2 holds by assuming that the convex functions g_i, for $i = 1, \ldots, m$ are such that
$$g_i(x + td) \geq g_i(x) + t\nabla g_i(x)^\top d + \frac{1}{2}\gamma t^2 \|d\|^2 \tag{A.6}$$
with $\gamma > 0$. Indeed, in this case it is easy to see that a direction d is a feasible direction at $x^\star$ if and only if
$$\nabla g_i(x^\star)^\top d < 0 \quad i : g_i(x^\star) = 0 \qquad \nabla h_j(x^\star)^\top d = 0 \quad i = 1, \ldots, p$$
Condition (A.6) is satisfied by assuming that the functions g_i are twice continuosly differentiable and the Hessian matrix is positive definite.

Condition (A.6) holds also for continuously differentiable functions g_i assuming that they are strongly convex with constant $c_i > 0$, i.e., that for $i = 1, \ldots, m$ it holds

$$g_i(y) \geq g_i(x) + \nabla g_i(x)^\top (y - x) + \frac{c_i}{2} \|y - x\|^2, \quad \forall\, x, y.$$

Appendix B

Multi-Objective Projected Gradient Descent Method

The Multi-Objective Projected Gradient Descent (MOPGD) method to solve problems of the form (7.1) has been proposed by Drummond and Iusem (2004) and then further developed and analyzed by Fukuda and Drummond (2011) and Fukuda and Drummond (2013); the main results related to MOPGD can be found summarized in the survey from Fukuda and Drummond (2014).

In this work we employed a simple variant (formally defined by the pseudocode in Algorithm 9) of the standard MOPGD proposed by Drummond and Iusem (2004). Specifically, we used a different definition for the constrained steepest descent direction, similarly to what is done for the Multi-objective Steepest Descent algorithm by Fliege and Svaiter (2000, Section 3.1), i.e., we replaced

$$\arg\min_{z\in C}\max_{j=1,\dots,m}\nabla f_j(x)^T(z-x)+\frac{1}{2}\|z-x\|^2$$

with

$$\arg\min_{\substack{z\in C\\ \|z-x\|\leq 1}}\max_{j=1,\dots,m}\nabla f_j(x)^T(z-x). \tag{B.1}$$

This choice is motivated by the fact that, as long as C is defined by linear constraints and the ℓ_∞ norm is used, problem (B.1) is an LP problem that can be solved easily. We refer to the optimal value of problem (B.1) by $\theta(x)$.

The idea of the method is that of taking, at each iteration, a step along the steepest common descent feasible direction. In order to guarantee convergence to Pareto critical points, Algorithm 9 makes use of a backtracking Armijo-like line search procedure, which is described in Algorithm 10. The idea of the line search procedure is that of halving the step size as long as a sufficient decrease hasn't been reached for all objective functions.

Algorithm 9: `MultiObjectiveProjectedGradientDescent`

```
Input: $\beta \in (0,1)$, $x^0 \in \mathbb{R}^n$.
$k = 0$
while $x^k$ is not Pareto critical do
    Compute
        $$z^k \in \underset{\substack{z \in C \\ \|z - x^k\| \leq 1}}{\arg\min} \max_{j=1,\dots,m} \nabla f_j(x^k)^T (z - x^k)$$
    set $d^k = z^k - x^k$
    $\alpha_k = \texttt{ConstrainedLineSearch}(x^k, d^k, \beta)$
    $x^{k+1} = x^k + \alpha_k d^k$
    $k = k + 1$
return $x^k$
```

In this Appendix, we state the theoretical properties of the employed variant of the MOPGD procedure that are needed in the convergence analysis in Chapter 7. The following Proposition guarantees that, given a feasible common descent direction at x, Algorithm 10 always returns a strictly positive step length in a finite number of iterations.

Proposition B.1 (Drummond and Iusem (2004); Proposition 1)**.** *Consider problem* (7.1)*. If F is continuously differentiable, $\beta \in (0,1)$, $x, z \in C$, $d = z - x$ and $J_F(x)d < 0$, then there exists $\epsilon \in (0,1)$ such that , for all $t \in (0, \epsilon)$,*

$$F(x + td) < F(x) + \beta t J_F(x) d.$$

We next state and prove the properties of Algorithm 9. We provide for the sake of completeness the proofs for these properties, which, although similar to those of the standard MOPGD, cannot be found in the literature.

Algorithm 10: `ConstrainedLineSearch`

```
Input: $\beta \in (0,1)$, $x \in \mathbb{R}^n$, $d \in \mathbb{R}^n$.
$j = 0$
while $F(x + \frac{1}{2^j} d) \not\leq F(x) + \beta \frac{1}{2^j} J_F(x) d$ do
    set $j = j + 1$
return $\frac{1}{2^j}$
```

Proposition B.2. *Let $\{x^k\}$ be the sequence generated by Algorithm 9 on problem* (7.1)*. Then:*

(a) the mapping $x^k \mapsto \theta(x^k)$ is continuous;

(b) $\{x^k\} \subset C$;

(c) the sequence $\{F(x^k)\}$ is monotonically strictly decreasing;

(d) every accumulation point, if any, of $\{x^k\}$ is a feasible stationary point;

(e) if C is bounded, or if F has bounded level sets in the multi-objective sense, $\{x^k\}$ admits at least one accumulation point.

Proof. We prove the properties one at a time:

(a) This property comes straightforwardly from the definition of θ, i.e., form the definition problem (B.1).

(b) The update rule of Algorithm 9 is given by $x^{k+1} = x^k + \alpha_k d^k$. Now, d^k is feasible at x^k by the definition of problem (B.1) and the convexity of C. Also, $\alpha_k \leq 1$ by the instructions of Algorithm 10. Hence, from the convexity of C, $x^k + \alpha_k d \in C$.

(c) From the instructions of Algorithm 10, we have

$$F(x^{k+1}) = F(x^k + \alpha_k d^k) \leq F(x^k) + \beta\alpha_k J_F(x^k) d^k \leq F(x^k) + \beta\alpha_k \theta(x^k) e < F(x^k),$$

where the last step comes from the fact that if it was $\theta(x^k) = 0$ then Algorithm 9 would stop and that $\alpha_k > 0$.

(d) Let $\bar{x}$ be an accumulation point of the sequence $\{x^k\}$, i.e., there exists $K \subseteq \{0, 1, \ldots\}$ such that $x^k \to \bar{x}$ for $k \to \infty$, $k \in K$. From (b) and recalling the closedness of C, we have that $\bar{x} \in C$. From (c), we have that $F(x^{k+1}) < F(x^k)$, hence the sequence $\{F(x^k)\}$ has limit $\bar{F}$; F is continuous, therefore $F(x^k) \to F(\bar{x})$ for $k \in K$, $k \to \infty$; hence, the limit of the whole sequence $\bar{F}$ is equal to $F(\bar{x})$ and is therefore finite.

From the instructions of Algorithm 10, we also have that

$$F(x^k + \alpha_k d^k) \leq F(x^k) + \beta\alpha_k J_F(x^k) d^k \leq F(x^k) + \left(\beta\alpha_k \max_{j=1,\ldots,m} \nabla f_j(x^k)^T d^k\right) e.$$

$\|d^k\| \leq 1$ by the definition of problem (B.1), hence the sequence d^k is bounded. Thus, there exists $K_1 \subseteq K$ such that $d^k \to \bar{d}$ (and similarly $z^k \to \bar{z}$) when $k \to \infty$, $k \in K_1$.

Now, for all $k \in K_1$, we have

$$F(x^{k+1}) - F(x^k) \leq \left(\beta\alpha_k \max_{j=1,\ldots,m} \nabla f_j(x^k)^T d^k\right) e.$$

Taking the limits we get that

$$0 = \bar{F} - \bar{F} \leq \beta \left(\lim_{\substack{k\to\infty \\ k\in K_1}} \alpha_k \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T d^k \right) e.$$

Recalling that $\beta > 0$, $\alpha_k > 0$ for all k by the properties of Algorithm 10 and $\max_{j=1,\dots,m} \nabla f_j(x^k)^T d^k \leq 0$ by the definition of d^k, we get that

$$\lim_{\substack{k\to\infty \\ k\in K_1}} \alpha_k \max_{j=1,\dots,m} \nabla f_j(x^k)^T d^k = 0.$$

We have now two possible cases:

(i) $\max_{j=1,\dots,m} \nabla f_j(x^k)^T d^k \to 0$ as $k \to \infty$, $k \in K$. This implies that

$$\begin{aligned} 0 &= \lim_{\substack{k\to\infty \\ k\in K_1}} \max_{j=1,\dots,m} \nabla f_j(x^k)^T d^k = \lim_{\substack{k\to\infty \\ k\in K_1}} \max_{j=1,\dots,m} \nabla f_j(x^k)^T (z^k - x^k) \\ &= \lim_{\substack{k\to\infty \\ k\in K_1}} \min_{\substack{z\in C \\ \|z-x^k\|\leq 1}} \max_{j=1,\dots,m} \nabla f_j(x^k)^T (z - x^k) \\ &= \min_{\substack{z\in C \\ \|z-\bar{x}\|\leq 1}} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T (\bar{z} - \bar{x}) \end{aligned}$$

which, from Lemma 7.2, implies that $\bar{x}$ is Pareto-critical.

(i) $\alpha_k \to 0$ as $k \to \infty$, $k \in K_1$. From the instructions of Algorithm 10, we have that for all $q \in \mathbb{N}$ there exists $\bar{k} \in K_1$ such that for all $k \in K_1$, $k \geq \bar{k}$, we have

$$F\left(x^k + \frac{1}{2^q} d^k\right) \not\leq F(x^k) + \frac{\beta}{2^q} J_F(x^k) d^k.$$

Taking the limits (along a suitable subsequence if needed), we have that for some j it holds

$$f_j\left(\bar{x} + \frac{1}{2^q}\bar{d}\right) \geq f_j(\bar{x}) + \frac{\beta}{2^q} \nabla f_j(\bar{x})^T \bar{d}.$$

Being q arbitrary, we have from Proposition B.1 that

$$\max_{j=1,\dots,m} \nabla f_j(\bar{x})^T \bar{d} \geq 0,$$

from which we can conclude that

$$0 \leq \max_{j=1,\dots,m} \nabla f_j(\bar{x})^T \bar{d} = \min_{\substack{z\in C \\ \|z-\bar{x}\|\leq 1}} \max_{j=1,\dots,m} f_j(\bar{x})^T (z - \bar{x}) \leq 0,$$

which, recalling Lemma 7.2, completes the proof.

(e) If C is bounded, we have from (b) that $\{x^k\}$ is contained in a bounded set, hence the sequence has at least one accumulation point. On the other hand, if F has bounded level sets in the multi-objective sense, we have that $\mathcal{L}_F(F(x^0)) = \{x \in C \mid F(x) \leq F(x^0)\}$ is bounded; form (c) we have that $\{x^k\} \subset \mathcal{L}_F(F(x^0))$, hence the sequence has at least one accumulation point.

□

Appendix C

Publications

Journal papers

1. G. Galvan, **M. Lapucci**, T. Levato, M. Sciandrone, "An Alternating Augmented Lagrangian method for constrained nonconvex optimization", *Optimization Methods and Software*, 35.3 (2020): 502-520. **Candidate's contributions**: participated in the theoretical analysis, in the implementation of the algorithm and in carrying out numerical experiments.

2. G. Galvan, **M. Lapucci**. "On the convergence of inexact augmented Lagrangian methods for problems with convex constraints." *Operations Research Letters* 47.3 (2019): 185-189. **Candidate's contributions**: participated in the literature review and in the theoretical analysis.

3. L. Di Gangi, **M. Lapucci**, F. Schoen, A. Sortino, "An efficient optimization approach for best subset selection in linear regression, with application to model selection and fitting in autoregressive time-series." *Computational Optimization and Applications* 74.3 (2019): 919-948. **Candidate's contributions**: participated in the literature review, in the design of the algorithm and in the design and implementation of the experiments; carried out the theoretical analysis.

4. G. Cocchi, **M. Lapucci**. "An augmented Lagrangian algorithm for multi-objective optimization." *Computational Optimization and Applications* 77.1 (2020): 29-56. **Candidate's contributions**: carried out the literature review; designed the algorithms; carried out the theoretical analysis; participated in the design of the experiments.

5. G. Galvan, **M. Lapucci**, C.-J. Lin, M. Sciandrone, "A Two-Level Decomposition Framework Exploiting First and Second Order Information for SVM Training Problems." *Journal of Machine Learning Research* 22 (2021): 23-1. **Candidate's contributions**: participated in the literature review; designed and im-

plemented the algorithm; participated in the design of the experiments; carried out the experiments.

6. **M. Lapucci**, T. Levato, M. Sciandrone "Convergent Inexact Penalty Decomposition Methods for Cardinality-Constrained Problems." *Journal of Optimization Theory and Applications* 188.2 (2021): 473-496. **Candidate's contributions**: contributed to algorithm design, carried out the theoretical analysis; designed and carried out the experiments.

7. Fulvia Ceccarelli, Giulio Olivieri, Alessio Sortino, Lorenzo Dominici, Filmon Arefayne, Alessandra Ida Celia, Enrica Cipriano, Cristina Garufi, **Matteo Lapucci**, Silvia Mancuso, Francesco Natalucci, Valeria Orefice, Carlo Perricone, Carmelo Pirone, Viviana Antonella Pacucci, Francesca Romana Spinelli, Simona Truglia, Cristiano Alessandri, Marco Sciandrone, Fabrizio Conti, "Comprehensive disease control in systemic lupus erythematosus", *Seminars in Arthritis and Rheumatism*, Volume 51, Issue 2, 2021, Pages 404-408, **Candidate's contributions**: contributed to experiments design.

8. E. Civitelli, **M. Lapucci**, F. Schoen, A. Sortino, "An effective procedure for feature subset selection in logistic regression based on information criteria", *Computational Optimization and Applications*, 80, 1–32 (2021). **Candidate's contributions**: participated in the design of the algorithm and in the design of the experiments; carried out the literature review and the theoretical analysis.

9. G. Cocchi, **M. Lapucci**, P. Mansueto. "Pareto Front Approximation through a Multi-Objective Augmented Lagrangian Method." *EURO Journal on Computational Optimization* 9 (2021): 100008. **Candidate's contributions**: carried out the literature review; designed the algorithm; carried out the theoretical analysis; participated in the design of the experiments.

10. F. Ceccarelli, **M. Lapucci**, G. Olivieri, A. Sortino, F. Natalucci, F.R. Spinelli, C. Alessandri, M. Sciandrone, F. Conti, "Can Machine Learning models support physicians in Systemic Lupus Erythematosus diagnosis? Results from a monocentric cohort", *Joint Bone Spine*, 2021. **Candidate's contributions**: contributed to experiments design; contributed to carrying out the computational experiments.

11. R. Bisori, **M. Lapucci**, M. Sciandrone, "A Study on Sequential Minimal Optimization Methods for Standard Quadratic Problems", *4OR*, 2021. **Candidate's contributions**: Carried out the literature review, the theoretical analysis and the design of the experiments; contributed to carrying out the experiments.

12. L. Di Gangi, **M. Lapucci**, F. Schoen, A. Sortino, "Improved Maximum Likelihood Estimation of ARMA Models", *Lobachevsky Journal of Mathematics*, 2022.

Candidate's contributions: contributed to literature review, theoretical analysis and experiments design.

Papers under review

- **M. Lapucci**, "A Penalty Decomposition Approach for Multi-objective Cardinality-Constrained Optimization Problems", submitted to *Optimization Methods and Software*. **Candidate's contributions**: Single author, did everything.

- **M. Lapucci**, D. Pucci, "Mixed-Integer Quadratic Programming Reformulations of Multi-TaskLearning Models", submitted to *Mathematics in Engineering, special Issue on Mathematics of Machine Learning*. **Candidate's contributions**: carried out model design and experiments design; contributed to experiments implementation.

- **M. Lapucci**, F. Schoen, A. Sortino, "Regression and Black-Box Global Optimization Through Sparse RBF Models", submitted to *Journal of Global Optimization*. **Candidate's contributions**: contributed to algorithm design and experiments design; carried out the theoretical analysis.

- E. Civitelli, A. Sortino, **M. Lapucci**, F. Bagattini, G. Galvan, "A Robust Initialization of Residual Blocks for Effective ResNet Training without Batch Normalization", submitted to *IEEE Transactions on Neural Networks and Learning Systems*. **Candidate's contributions**: contributed to methodology design, theoretical derivations and experiments design.

- **M. Lapucci**, T. Levato, F. Rinaldi, M. Sciandrone, "A Unifying Framework for Sparsity Constrained Optimization", submitted to *INFORMS Mathematics of Operations Research*. **Candidate's contributions**: contributed to literature review, algorithm design, theoretical analysis; designed and carried out the computational experiments.

- **M. Lapucci**, P. Mansueto, F. Schoen, "A Memetic Procedure for Global Multi-Objective Optimization", submitted to *Mathematical Programming Computation*. **Candidate's contributions**: devised paper's initial concept; contributed to algorithm design, theoretical analysis and experiments design.

Bibliography

Anagnostopoulos, K. P. and Mamanis, G. (2010). A portfolio optimization model with three objectives and discrete variables. *Computers & Operations Research*, 37(7):1285–1297.

Armananzas, R. and Lozano, J. A. (2005). A multiobjective approach to the portfolio optimization problem. In *2005 IEEE Congress on Evolutionary Computation*, volume 2, pages 1388–1395. IEEE.

Bach, F., Jenatton, R., Mairal, J., Obozinski, G., et al. (2012). Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106.

Beck, A. and Eldar, Y. C. (2013). Sparsity constrained nonlinear optimization: Optimality conditions and algorithms. *SIAM Journal on Optimization*, 23(3):1480–1509.

Beck, A. and Hallak, N. (2016). On the minimization over sparse symmetric sets: projections, optimality conditions, and algorithms. *Mathematics of Operations Research*, 41(1):196–223.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

Beck, A. and Tetruashvili, L. (2013). On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization*, 23(4):2037–2060.

Belotti, P., Bonami, P., Fischetti, M., Lodi, A., Monaci, M., Nogales-Gómez, A., and Salvagnin, D. (2016). On handling indicator constraints in mixed integer programming. *Computational Optimization and Applications*, 65(3):545–566.

Ben Mhenni, R., Bourguignon, S., and Ninin, J. (2021). Global optimization for sparse solution of least squares problems. *Optimization Methods and Software*, pages 1–30.

Berge, C. (1963). *Topological Spaces: Including a Treatment of Multi-valued Functions, Vector Spaces and Convexity*. Macmillan.

Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334.

Bertsekas, D. P. and Tsitsiklis, J. N. (1989). *Parallel and distributed computation: numerical methods*, volume 23. Prentice hall Englewood Cliffs, NJ.

Bertsimas, D. and Cory-Wright, R. (2018). A scalable algorithm for sparse portfolio selection. *arXiv preprint arXiv:1811.00138*.

Bertsimas, D., Cory-Wright, R., and Pauphilet, J. (2019). A unified approach to mixed-integer optimization: Nonlinear formulations and scalable algorithms. *arXiv preprint arXiv:1907.02109*.

Bertsimas, D. and King, A. (2017). Logistic regression: From art to science. *Statistical Science*, pages 367–384.

Bertsimas, D., King, A., Mazumder, R., et al. (2016). Best subset selection via a modern optimization lens. *Annals of Statistics*, 44(2):813–852.

Bertsimas, D., Pauphilet, J., and Van Parys, B. (2017). Sparse classification: A scalable discrete optimization perspective. *arXiv preprint arXiv:1710.01352*.

Bertsimas, D. and Shioda, R. (2009). Algorithm for cardinality-constrained quadratic optimization. *Computational Optimization and Applications*, 43(1):1–22.

Bienstock, D. (1996). Computational study of a family of mixed-integer quadratic programming problems. *Mathematical Programming*, 74(2):121–140.

Birgin, E. G. and Martínez, J. M. (2014). *Practical augmented Lagrangian methods for constrained optimization*. SIAM.

Blumensath, T. and Davies, M. E. (2009). Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274.

Boudt, K. and Wan, C. (2020). The effect of velocity sparsity on the performance of cardinality constrained particle swarm optimization. *Optimization Letters*, 14(3):747–758.

Brito, R. P. and Vicente, L. N. (2013). Efficient cardinality/mean-variance portfolios. In *IFIP Conference on System Modeling and Optimization*, pages 52–73. Springer.

Burdakov, O. P., Kanzow, C., and Schwartz, A. (2016). Mathematical programs with cardinality constraints: reformulation by complementarity-type conditions and a regularization method. *SIAM Journal on Optimization*, 26(1):397–425.

Candès, E. J. and Wakin, M. B. (2008). An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30.

Carlini, N. and Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE.

Carreira-Perpinán, M. A. and Idelbayev, Y. (2018). "Learning-compression" algorithms for neural net pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8532–8541.

Carrizosa, E. and Frenk, J. B. G. (1998). Dominating sets for convex functions with some applications. *Journal of Optimization Theory and Applications*, 96(2):281–295.

Cesarone, F., Scozzari, A., and Tardella, F. (2013). A new method for mean-variance portfolio optimization with cardinality constraints. *Annals of Operations Research*, 205(1):213–234.

Chang, T.-J., Meade, N., Beasley, J. E., and Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research*, 27(13):1271–1302.

Chartrand, R. (2007). Exact reconstruction of sparse signals via nonconvex minimization. *IEEE Signal Processing Letters*, 14(10):707–710.

Chen, C. and Wei, Y. (2019). Robust multiobjective portfolio optimization: a set order relations approach. *Journal of Combinatorial Optimization*, 38(1):21–49.

Chen, S. S., Donoho, D. L., and Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159.

Chen, X., Xu, F., and Ye, Y. (2010). Lower bound theory of nonzero entries in solutions of $\ell_2 - \ell_p$ minimization. *SIAM Journal on Scientific Computing*, 32(5):2832–2852.

Chiam, S., Tan, K., and Al Mamum, A. (2008). Evolutionary multi-objective portfolio optimization in practical context. *International Journal of Automation and Computing*, 5(1):67–80.

Civitelli, E., Lapucci, M., Schoen, F., and Sortino, A. (2021). An effective procedure for feature subset selection in logistic regression based on information criteria. *Computational Optimization and Applications*, pages 1–32.

Cocchi, G. and Lapucci, M. (2020). An augmented lagrangian algorithm for multiobjective optimization. *Computational Optimization and Applications*, 77(1):29–56.

Cocchi, G., Lapucci, M., and Mansueto, P. (2021). Pareto front approximation through a multi-objective augmented Lagrangian method. *EURO Journal on Computational Optimization*, 9:100008.

Cocchi, G., Levato, T., Liuzzi, G., and Sciandrone, M. (2020a). A concave optimization-based approach for sparse multiobjective programming. *Optimization Letters*, 14(3):535–556.

Cocchi, G., Liuzzi, G., Lucidi, S., and Sciandrone, M. (2020b). On the convergence of steepest descent methods for multiobjective optimization. *Computational Optimization and Applications*, 77(1):1–27.

Custódio, A. L., Madeira, J. A., Vaz, A. I. F., and Vicente, L. N. (2011). Direct multisearch for multiobjective optimization. *SIAM Journal on Optimization*, 21(3):1109–1140.

d'Aspremont, A., Bach, F., and El Ghaoui, L. (2008). Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9(7).

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197.

Deng, G.-F., Lin, W.-T., and Lo, C.-C. (2012). Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization. *Expert Systems with Applications*, 39(4):4558–4566.

Di Gangi, L., Lapucci, M., Schoen, F., and Sortino, A. (2019). An efficient optimization approach for best subset selection in linear regression, with application to model selection and fitting in autoregressive time-series. *Computational Optimization and Applications*, 74(3):919–948.

Di Lorenzo, D., Liuzzi, G., Rinaldi, F., Schoen, F., and Sciandrone, M. (2012). A concave optimization-based approach for sparse portfolio selection. *Optimization Methods and Software*, 27(6):983–1000.

Dolan, E. D. and Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306.

Donoho, D. L. and Tsaig, Y. (2008). Fast solution of ℓ_1-norm minimization problems when the solution may be sparse. *IEEE Transactions on Information Theory*, 54(11):4789–4812.

Drummond, L. G. and Iusem, A. N. (2004). A projected gradient method for vector optimization problems. *Computational Optimization and applications*, 28(1):5–29.

Drummond, L. G., Maculan, N., and Svaiter, B. F. (2008). On the choice of parameters for the weighting method in vector optimization. *Mathematical Programming*, 111(1-2):201–216.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Eichfelder, G. (2009). An adaptive scalarization method in multiobjective optimization. *SIAM Journal on Optimization*, 19(4):1694–1718.

Feng, M., Mitchell, J. E., Pang, J.-S., Shen, X., and Wächter, A. (2013). Complementarity formulations of l0-norm optimization problems. *Industrial Engineering and Management Sciences. Technical Report. Northwestern University, Evanston, IL, USA*.

Fliege, J. (2001). OLAF – a general modeling system to evaluate and optimize the location of an air polluting facility. *OR-Spektrum*, 23(1):117–136.

Fliege, J., Drummond, L. G., and Svaiter, B. F. (2009). Newton's method for multiobjective optimization. *SIAM Journal on Optimization*, 20(2):602–626.

Fliege, J. and Svaiter, B. F. (2000). Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, 51(3):479–494.

Fliege, J. and Vaz, A. I. F. (2016). A method for constrained multiobjective optimization based on SQP techniques. *SIAM Journal on Optimization*, 26(4):2091–2119.

Foucart, S. and Rauhut, H. (2013). An invitation to compressive sensing. In *A mathematical introduction to compressive sensing*, pages 1–39. Springer.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Fukuda, E. H. and Drummond, L. G. (2011). On the convergence of the projected gradient method for vector optimization. *Optimization*, 60(8-9):1009–1021.

Fukuda, E. H. and Drummond, L. G. (2013). Inexact projected gradient method for vector optimization. *Computational Optimization and Applications*, 54(3):473–493.

Fukuda, E. H. and Drummond, L. M. G. (2014). A survey on multiobjective descent methods. *Pesquisa Operacional*, 34(3):585–620.

Galvan, G. and Lapucci, M. (2019). On the convergence of inexact augmented Lagrangian methods for problems with convex constraints. *Operations Research Letters*, 47(3):185–189.

Galvan, G., Lapucci, M., Levato, T., and Sciandrone, M. (2020). An alternating augmented Lagrangian method for constrained nonconvex optimization. *Optimization Methods and Software*, 35(3):502–520.

Gao, J. and Li, D. (2013). Optimal cardinality constrained portfolio selection. *Operations Research*, 61(3):745–761.

Ge, D., Jiang, X., and Ye, Y. (2011). A note on the complexity of L_p minimization. *Mathematical Programming*, 129(2):285–299.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Gotoh, J.-y., Takeda, A., and Tono, K. (2018). DC formulations and algorithms for sparse optimization problems. *Mathematical Programming*, 169(1):141–176.

Gravel, M., Martel, J. M., Nadeau, R., Price, W., and Tremblay, R. (1992). A multicriterion view of optimal resource allocation in job-shop production. *European Journal of Operational Research*, 61(1-2):230–244.

Grippo, L. and Sciandrone, M. (1999). Globally convergent block-coordinate techniques for unconstrained optimization. *Optimization Methods and Software*, 10(4):587–637.

Grippo, L. and Sciandrone, M. (2000). On the convergence of the block nonlinear Gauss–Seidel method under convex constraints. *Operations Research Letters*, 26(3):127–136.

Guillot, D., Rajaratnam, B., Rolfs, B. T., Maleki, A., and Wong, I. (2012). Iterative thresholding algorithm for sparse inverse covariance estimation. *arXiv preprint arXiv:1211.2532*.

Gurobi Optimization, L. (2020). Gurobi optimizer reference manual.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Jörnsten, K., Näsberg, M., and Smeds, P. (1985). *Variable Splitting: A New Lagrangean Relaxation Approach to Some Mathematical Programming Models*. LiTH MAT R.: Matematiska Institutionen. University of Linköping, Department of Mathematics.

Kanzow, C., Raharja, A. B., and Schwartz, A. (2021). An augmented Lagrangian method for cardinality-constrained optimization problems. *Journal of Optimization Theory and Applications*, pages 1–21.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Konak, A., Coit, D. W., and Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*, 91(9):992–1007.

Laumanns, M., Thiele, L., Deb, K., and Zitzler, E. (2002). Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary computation*, 10(3):263–282.

Le Thi, H. A., Dinh, T. P., Le, H. M., and Vo, X. T. (2015). DC approximation approaches for sparse optimization. *European Journal of Operational Research*, 244(1):26–46.

LeCun, Y. and Cortes, C. (2010). MNIST handwritten digit database.

Li, D. and Sun, X. (2006). *Nonlinear integer programming*, volume 84. Springer Science & Business Media.

Liu, D. C. and Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528.

Liuzzi, G., Lucidi, S., Parasiliti, F., and Villani, M. (2003). Multiobjective optimization techniques for the design of induction motors. *IEEE Transactions on Magnetics*, 39(3):1261–1264.

Lu, Z. and Zhang, Y. (2013). Sparse approximation via penalty decomposition methods. *SIAM Journal on Optimization*, 23(4):2448–2478.

Lucidi, S., Piccialli, V., and Sciandrone, M. (2005). An algorithm model for mixed variable programming. *SIAM Journal on Optimization*, 15(4):1057–1084.

Mairal, J., Bach, F., and Ponce, J. (2014). Sparse modeling for image and vision processing. *arXiv preprint arXiv:1411.3230*.

Malioutov, D. M., Cetin, M., and Willsky, A. S. (2005). Homotopy continuation for sparse signal representation. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 5, pages v–733. IEEE.

Mangasarian, O. (1999). Minimum-support solutions of polyhedral concave programs. *Optimization*, 45(1-4):149–162.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, 7(1):77–91.

Markowitz, H. M. (1994). The general mean-variance portfolio selection problem. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 347(1684):543–549.

Miller, A. (2002). *Subset selection in regression*. CRC Press.

Miyashiro, R. and Takano, Y. (2015). Mixed integer second-order cone programming formulations for variable selection in linear regression. *European Journal of Operational Research*, 247(3):721–731.

Mourad, N. and Reilly, J. P. (2010). Minimizing nonconvex functions for sparse vector reconstruction. *IEEE Transactions on Signal Processing*, 58(7):3485–3496.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234.

Nguyen, T. T., Soussen, C., Idier, J., and Djermoune, E.-H. (2019). NP-hardness of ℓ_0 minimization problems: revision and extension to the non-negative setting. In *13th International Conference on Sampling Theory and Applications, SampTA 2019*.

Oliphant, T. E. (2006). *A guide to NumPy*, volume 1. Trelgol Publishing USA.

Palermo, G., Silvano, C., Valsecchi, S., and Zaccaria, V. (2003). A system-level methodology for fast multi-objective design space exploration. In *Proceedings of the 13th ACM Great Lakes symposium on VLSI*, pages 92–95. ACM.

Pascoletti, A. and Serafini, P. (1984). Scalarizing vector optimization problems. *Journal of Optimization Theory and Applications*, 42(4):499–524.

Pellegrini, R., Campana, E., Diez, M., Serani, A., Rinaldi, F., Fasano, G., Iemma, U., Liuzzi, G., Lucidi, S., and Stern, F. (2014). Application of derivative-free multi-objective algorithms to reliability-based robust design optimization of a high-speed catamaran in real ocean environment1. *Engineering Optimization IV-Rodrigues et al.(Eds.)*, page 15.

Radziukynienė, I. and Žilinskas, A. (2008). Evolutionary methods for multi-objective portfolio optimization. In *Proceedings of the World Congress on Engineering*, volume 2, pages 1155–1159.

Reed, R. (1993). Pruning algorithms-a survey. *IEEE Transactions on Neural Networks*, 4(5):740–747.

Rinaldi, F., Schoen, F., and Sciandrone, M. (2010). Concave programming for minimizing the zero-norm over polyhedral sets. *Computational Optimization and Applications*, 46(3):467–486.

Ruszczynski, A. (2011). *Nonlinear optimization*. Princeton university press.

Sun, Y., Ng, D. W. K., Zhu, J., and Schober, R. (2016). Multi-objective optimization for robust power efficient and secure full-duplex wireless communication systems. *IEEE Transactions on Wireless Communications*, 15(8):5511–5526.

Teng, Y., Yang, L., Yu, B., and Song, X. (2017). A penalty PALM method for sparse portfolio selection problems. *Optimization Methods and Software*, 32(1):126–147.

Tian, Y., Zhang, X., Wang, C., and Jin, Y. (2019). An evolutionary algorithm for large-scale sparse multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 24(2):380–393.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tillmann, A. M., Bienstock, D., Lodi, A., and Schwartz, A. (2021). Cardinality Minimization, Constraints, and Regularization: A Survey. *arXiv preprint arXiv:2106.09606*.

Virtanen, P., Gommers, R., Oliphant, T. E., and SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Weston, J., Elisseeff, A., Schölkopf, B., and Tipping, M. (2003). Use of the zero norm with linear models and kernel methods. *The Journal of Machine Learning Research*, 3:1439–1461.

Xidonas, P., Hassapis, C., Mavrotas, G., Staikouras, C., and Zopounidis, C. (2018). Multiobjective portfolio optimization: bridging mathematical theory with asset management practice. *Annals of Operations Research*, 267(1-2):585–606.

Yin, W., Osher, S., Goldfarb, D., and Darbon, J. (2008). Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging sciences*, 1(1):143–168.

Yu, B., Mitchell, J. E., and Pang, J.-S. (2019). Solving linear programs with complementarity constraints using branch-and-cut. *Mathematical Programming Computation*, 11(2):267–310.

Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286.