BookID 188985_ChapID FM_Proof# 1 - 06/05/11

Heinz H. Bauschke • Regina S. Burachik
Patrick L. Combettes • Veit Elser
D. Russell Luke • Henry Wolkowicz
Editors

Fixed-Point Algorithms for Inverse Problems in Science and Engineering

Springer

Editors
Heinz H. Bauschke
Department of Mathematics and Statistics
University of British Columbia
Okanagan Campus
Kelowna, British Columbia
Canada
heinz.bauschke@ubc.ca

Regina S. Burachik
School of Mathematics & Statistics
Division of Information Technology
Engineering & the Environment
University of South Australia
Mawson Lakes Campus
Mawson Lakes Blvd.
5095 Mawson Lakes
South Australia
regina.burachik@unisa.edu.au

Patrick L. Combettes
Université Pierre et Marie Curie
Laboratoire Jacques-Louis Lions
Place Jussieu 4
75005 Paris
France
plc@math.jussieu.fr

Veit Elser
Department of Physics, Lab. Atomic
& Solid State
Cornell University
Clark Hall
14853–2501 Ithaca, New York
USA
ve10@cornell.edu

D. Russell Luke
Department of Mathematical Sciences
University of Delaware
Ewing Hall
19716 Newark, Delaware
USA
rluke@math.udel.edu

Henry Wolkowicz
Department of Combinatorics
& Optimization
Faculty of Mathematics
University of Waterloo
Waterloo, Ontario
Canada
hwolkowicz@uwaterloo.ca

ISSN 1931-6828
ISBN 978-1-4419-9568-1 e-ISBN 978-1-4419-9569-8
DOI 10.1007/978-1-4419-9569-8
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2011928237

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Contents

Chapter 1
Entropic regularization of the ℓ_0 function

Jonathan M. Borwein and D. Russell Luke

Abstract Many problems of interest where more than one solution is possible seek, among these, the one that is sparsest. The objective that most directly accounts for sparsity, the ℓ_0 metric, is usually avoided since this leads to a combinatorial optimization problem. The function $\|x\|_0$ is often viewed as the limit of the ℓ_p metrics. Naturally, there have been some attempts to use this as an objective for p small, though this is a nonconvex function for $p < 1$. We propose instead a scaled and shifted Fermi-Dirac entropy with two parameters, one controlling the smoothness of the approximation and the other the *steepness* of the metric. Our proposed metric is a convex relaxation for which a strong duality theory holds, yielding dual methods for metrics approaching the desired $\|\cdot\|_0$ function. Without smoothing, we propose a dynamically reweighted subdifferential descent method with "exact" line search that is finitely terminating for constraints that are well-separated. This algorithm is shown to recapture in a special case certain well-known "greedy" algorithms. Consequently we are able to provide an explicit algorithm whose fixed point, under the appropriate assumptions, is the sparsest possible solution. The variational perspective yields general strategies to make the algorithm more robust.

Key words: convex optimization, Fenchel duality, entropy, regularization, sparsity, signal processing
AMS 2010 Subject Classification: 49M20, 65K10, 90C30

Jonathan M. Borwein
CARMA, School of Mathematical and Physical Sciences, University of Newcastle, NSW 2308, Australia. Research supported by the Australian Research Council. e-mail: jonathan.borwein@newcastle.edu.au

D. Russell Luke
Institute for Numerical and Applied Mathematics, Universität Göttingen, Germany. Research supported by the US National Science Foundation grant DMS-0712796. e-mail: r.luke@math.uni-goettingen.de

1.1 Introduction

Let $\mathbb{E}$ and $\mathbb{Y}$ be Euclidean spaces, and let $A : \mathbb{E} \to \mathbb{Y}$ be linear. We consider the problem

$$\begin{array}{ll} \underset{x\in\mathbb{E}}{\text{minimize}} & \varphi(x) \\ \text{subject to} & A(x) = b \end{array} \tag{1.1}$$

where $\varphi(x) : \mathbb{E} \to \mathbb{R}$ is a lower semi-continuous (lsc), symmetric subadditive function that, in one way or another, counts the nonzero elements of x. This model has received a great deal of attention recently in applications where the number of constraints is much smaller than the dimension of the domain. Examples include the well-known compressed sensing [4], where $\mathbb{E} = \mathbb{R}^n$, $\mathbb{Y} = \mathbb{R}^m$ ($m \ll n$) and $\varphi(x) \equiv \sum_j |\text{sign}(x_j)|$.

Another instance of importance is low-rank matrix reconstruction [13, 5]. Here $\mathbb{E} = \mathbb{R}^{m\times n}$, $\mathbb{Y} = \mathbb{R}^{m\times n}$ and $\varphi(x) \equiv \text{rank}(x)$. The goal in both of these applications is to find a "sparsest" solution x^* to $A(x) = b$. Both of the optimization problems associated with these examples are combinatorial and, in general, NP-hard [12]. At the expense of some generality we will narrow our discussion to the case where $\mathbb{E} = \mathbb{R}^n$ and $\mathbb{Y} = \mathbb{R}^m$.

Before addressing the counting objective directly, we review some elementary observations about the most common relaxation of this problem, ℓ_1 optimization.

1.1.1 Elementary ℓ_1 minimization

A natural first step toward solving such problems has been to solve convex relaxations instead, $\varphi(x) = \|x\|_1 \equiv \varphi_1(x)$. It has been known for some time that ℓ_1 optimization promotes sparsity in underdetermined systems [15, 7]. Later works established criteria under which the solution to (1.1) is unique and exactly matches the true signal x^* [8, 9, 6]. Sparsity of the true signal x^* and the structure of the matrix A are key requirements.

A qualitative geometric interpretation of these facts is obtained by considering the Fenchel dual [1] to problem (1.1) when $\varphi = \varphi_1$:

$$\begin{array}{ll} \underset{y\in\mathbb{R}^m}{\text{maximize}} & b^T y \\ \text{subject to} & \left(A^T y\right)_j \in [-1,1] \quad j = 1,2,\ldots,n. \end{array} \tag{1.2}$$

By *strong Fenchel duality* the optimal values of the primal and dual problems are equivalent, and a solution of the dual problem yields a solution to the primal. The dual problem yields valuable geometric insight. Elementary facts from linear programming guarantee that the solution includes a vertex of the polyhedron described by the constraints. The number of active constraints in the dual problem provides a crude upper bound on the number of nonzero elements of the sparsest solution to the primal problem. Unless the number of active constraints in the dual problem is

less than or equal to the number of measurements m, there is no hope of uniquely recovering x^*. Supposing that the solution to (1.2) is indeed unique, a more vexing question is whether or not the corresponding primal solution is the sparsest solution to $Ax = b$. Here, it appears, convex analysis is at a loss to provide an answer.

1.1.2 ℓ_0 *minimization*

We gain some insight into this breakdown by considering the dual of the original sparse optimization problem. For $\varphi(x) = \sum_j |\text{sign}(x_j)| \equiv \varphi_0(x)$ in (1.1) the equivalence of the primal and dual problems is lost due to the nonconvexity of the objective. The theory of Fenchel duality still yields *weak duality*, but this is of limited use in this instance. The Fenchel dual to (1.1) when $\varphi = \varphi_0$ is

$$\begin{array}{ll} \underset{y\in\mathbb{R}^m}{\text{maximize}} & b^T y \\ \text{subject to} & \left(A^T y\right)_j = 0 \quad j = 1,2,\ldots,n. \end{array} \tag{1.3}$$

If we denote the *values* of the primal (1.1) and dual problems (1.3) by p and d respectively, then these values satisfy the *weak duality inequality* $p \geq d$. The primal problem is a combinatorial optimization problem, and hence NP-hard; the dual problem, however, is a linear program, which is finitely terminating. Relatively elementary variational analysis provides a lower bound on the sparsity of signals x that satisfy the measurements. In this instance, however, the lower bound only reconfirms what we already know. Indeed, if A is full rank, then the only solution to the dual problem is $y = 0$. In other words, the minimal sparsity of the solution to the primal problem is greater than or equal to zero, which is obvious. The loss of information in passing from primal to dual formulations of nonconvex problems is a common phenomenon and at the heart of the difficulties in answering some very basic questions about sparse, and more generally nonconvex, optimization.

Our goal in this paper is two-fold: first to dig deeper into the convex analysis to see what can indeed be learned about the nonconvex problem from various convex relaxations, and second, to take what has been learned by other means and incorporate these advances into convex analysis and algorithms. As we showed with example (1.3), the dual of the ℓ_0 problem is uninformative but trivial to solve. The conventional approach is to view ℓ_0 as a limit of the nonconvex p-metrics. However, the ℓ_p problems for $0 < p < 1$ are also NP hard and the duals to these optimization problems suffer the same loss of information that the dual to the ℓ_0 function suffers. The question that motivates our work is whether one can use convex relaxations approaching something related to the ℓ_0 function – something in the dual space – that are still informative with respect to the original ℓ_0 problem, but yield optimization problems that are solvable in polynomial time. The connection between the nonconvex and the convex that we explore is the Fenchel conjugate of the ℓ_0 function, which can be written as the limit of convex functions. We then study how well our proposed convex relaxations work for solving the sparse recovery problem.

1.1.3 Notation

Throughout this work we use $\|\cdot\|$ without any subscript to denote the L^2-norm. When a different norm is meant, a subscript is added explicitly to the norm as with $\|\cdot\|_1$. We denote the *projection* of a point z onto the set C with respect to the L^2 norm by $P_C(z)$ where

$$P_C(z) \equiv \{x \in C \mid \|x-z\| = \inf_{y\in C} \|z-y\|\}.$$

We denote the nonnegative orthant in $\mathbb{R}^n$ by $\mathbb{R}^n_+$ and the *extended reals* by $\overline{\mathbb{R}} \equiv \mathbb{R} \cup \{+\infty\}$. It is not uncommon to define the objective φ on the extended reals as a mapping from $\mathbb{R}^n$ to $\overline{\mathbb{R}}$. The *normal cone mapping* of a set $C \subset \mathbb{R}^n$ at a point x is defined by

$$N_C(x) \equiv \begin{cases} \{w \in \mathbb{R}^n \text{ with } (z-x)^T w \leq 0 \text{ for all } z \in C\} & \text{if } x \in C \\ \emptyset & \text{if } x \notin C. \end{cases}$$

We denote by $ri(C)$ the *relative interior* of C, that is the interior of C relative to its affine hull. The *indicator function* of a set C, ι_C is defined by

$$\iota_C(x) \equiv \begin{cases} 0 & \text{for } x \in C \\ +\infty & \text{for } x \notin C. \end{cases}$$

We use the indicator function to treat constraint sets as functions. For a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ and a point $\bar{x}$ in the domain of f, the *subdifferential* of f at $\bar{x}$, denoted $\partial f(\bar{x})$ is defined by

$$\partial f(\bar{x}) \equiv \left\{ w \in \mathbb{R}^n \mid w^T(x-\bar{x}) \leq f(x) - f(\bar{x}), \text{ for all } x \in \mathbb{R}^n \right\}. \tag{1.4}$$

when $\bar{x}$ is not in the domain of f we define $\partial f(\bar{x}) = \emptyset$. The *Fenchel conjugate* of a mapping $f : \mathbb{R}^n \to [-\infty, +\infty]$, denoted $f^* : \mathbb{R}^n \to [-\infty, +\infty]$, is defined by

$$f^*(y) = \sup_{x\in\mathbb{R}^n} \{y^T x - f(x)\}. \tag{1.5}$$

The conjugate is always convex (as a supremum of affine functions) while $f = f^{**}$ exactly if f is convex, proper (not everywhere infinite) and lower semi-continuous (lsc) [1]. Finally, we make frequent reference to boxes in $\mathbb{R}^n$ centered at the origin with sides of length $2I_j$ $(j = 1, 2, \ldots, n)$; these are denoted by $R_I \equiv [-I_1, I_1] \times [-I_2, I_2] \times \ldots [-I_n, I_n]$ for $I = (I_1, I_2, \ldots, I_n)$.

1.2 Entropic regularization of the zero metric

The Fenchel conjugates of the functions $\varphi_1(x) \equiv \|x\|_1$ and $\varphi_0(x) \equiv \sum_j |\text{sign}(x_j)|$ are given respectively by

$$\varphi_1^*(y) \equiv \begin{cases} 0 & y \in [-1,1] \\ +\infty & \text{else} \end{cases} \qquad (\varphi_1(x) \equiv \|x\|_1) \tag{1.6}$$

$$\varphi_0^*(y) \equiv \begin{cases} 0 & y = 0 \\ +\infty & \text{else} \end{cases} \qquad (\varphi_0(x) \equiv \|x\|_0). \tag{1.7}$$

It is not uncommon to consider the function $\|\cdot\|_0$ as the limit of $\left(\sum_j |x_j|^p\right)^{1/p}$ as $p \to 0$. The notation is misleading since $\|\cdot\|_0$ is not a norm; the fact that

$$\|x\|_0 = \lim_{p \to 0^+} \sum_j |x_j|^p$$

shows that $d_0(x,y) := \|x-y\|_0$ still produces a metric since $\sum_j |x_j - y_j|^p$ does for $0 < p < 1$.

We propose an alternative strategy based on regularization of the conjugates. For $L \in \mathbb{R}^n_+$ and $\varepsilon > 0$ define the rectangle $R_L \equiv [-L_1, L_1] \times [-L_2, L_2] \times \cdots \times [-L_n, L_n]$ and

$$\phi_{\varepsilon,L}(y) \equiv \sum_{j=1}^n \psi_{\varepsilon,L_j}(y_j) \qquad (y = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n) \tag{1.8}$$

where

$$\psi_{\varepsilon,L_j}(y_j) \equiv \begin{cases} \varepsilon \left(\frac{(L_j+y_j)\ln(L_j+y_j) + (L_j-y)\ln(L_j-y_j)}{2L_j \ln(2)} - \frac{\ln(L_j)}{\ln(2)} \right) & \text{for } |y_j| < L_j \\ \varepsilon & \text{for } |y_j| = L_j \\ +\infty & \text{for } |y_j| > L_j. \end{cases} \tag{1.9}$$

This is a scaled and shifted *Fermi-Dirac entropy* [3, 1]. The value at the endpoints $y_j = \pm L_j$ follows from defining $0\ln(0) = 0$, which is standard in the literature. The inclusion of the endpoints ($y_j = \pm L_j$) in the domain of definition of $\psi_{\varepsilon,L_j}(y_j)$ provides a type of continuity in the limiting cases, namely as the closed interval $[-L_j, L_j]$ degenerates to the point $[0]$ and the relaxation parameter $\varepsilon \to 0$. This entropy is a smooth convex function on the interior of its domain and so elementary calculus can be used to calculate the Fenchel conjugate,

$$\phi_{\varepsilon,L}^*(x) = \sum_{j=1}^n \left(\frac{\varepsilon}{\ln(2)} \ln\left(4^{x_j L_j/\varepsilon} + 1\right) - x_j L_j - \varepsilon \right). \tag{1.10}$$

(Calculate the gradient of the objective in the Fenchel problem (1.5), satisfy first order conditions for optimality and substitute the optimal solution back into (1.5) to get (1.10) for the optimal value parameterized by the dual variable.)

- For $\varepsilon > 0$ fixed we have

$$\lim_{L\to 0} \phi_{\varepsilon,L}(y) = \begin{cases} 0 & y = 0 \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{L\to 0} \phi^*_{\varepsilon,L}(x) = 0.$$

- For $L > 0$ fixed, in the limit as $\varepsilon \to 0$ we have

$$\lim_{\varepsilon\to 0} \phi_{\varepsilon,L}(y) = \begin{cases} 0 & y \in R_L \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{\varepsilon\to 0} \phi^*_{\varepsilon,L}(x) = L|x|.$$

We write $\phi_{0,L}$, $\phi^*_{0,L}$, $\phi_{\varepsilon,0}$ and $\phi^*_{\varepsilon,0}$ for the limits. In contrast to the limit of $\phi_{\varepsilon,L}(y)$ for $\varepsilon > 0$ fixed, if $y = L$ in the limiting process we have $\lim_{L\to 0} \phi_{\varepsilon,L}(L) = \varepsilon$. By $\phi_{\varepsilon,0}(0)$ we mean the former limit, so that $\phi_{\varepsilon,0}(0) = 0$. Note that $\|\cdot\|_0$ and $\phi^*_{\varepsilon 0}$ have the same conjugate, but unlike $\|\cdot\|_0$ the biconjugate of $\phi^*_{\varepsilon 0}$ is itself, that is $\phi^{***} = \phi^*$. Also note that $\phi_{\varepsilon,L}$ and $\phi^*_{\varepsilon,L}$ are convex and smooth on the interior of their domains for all $\varepsilon, L > 0$. This is in contrast to metrics of the form $\left(\sum_j |x_j - y_j|^p\right)$ which are nonconvex for $p < 1$.

In order to maintain identification with φ in (1.1) we define

$$\varphi_{\varepsilon,L} \equiv \phi^*_{\varepsilon,L} \quad \text{and} \quad \varphi^*_{\varepsilon,L} \equiv \phi^{**}_{\varepsilon,L} = \phi_{\varepsilon,L}$$

where we have used the fact that the biconjugate of $\phi_{\varepsilon,L}$ is itself. We therefore consider the problem

$$\inf\{\varphi_{\varepsilon,L}(x) \mid x \in \mathbb{R}^n \text{ with } Ax = b\} \tag{1.11}$$

as a smooth convex relaxation of the conventional ℓ_p optimization for $0 \le p \le 1$. Our numerical approach to solve this problem will be to solve the dual.

Using Fenchel duality, the dual to this problem is the concave optimization problem

$$\sup\{y^T b - \varphi^*_{\varepsilon,L}(A^T y) \mid y \in \mathbb{R}^m\}. \tag{1.12}$$

where, again, $\varphi^*_{\varepsilon,L}(x) = \phi_{\varepsilon,L}(x)$ is given by (1.8). We reformulate this as a minimization problem

$$\underset{y\in\mathbb{R}^m}{\text{minimize}} \qquad \varphi^*_{\varepsilon,L}(A^T y) - y^T b \tag{1.13}$$

which we will solve with the method described next.

The objective in the dual problem is smooth and convex, so we could in principle apply any number of efficient unconstrained optimization algorithms. Also, for this relaxation, the same numerical techniques can be used for all $L \to 0$.

1.3 Algorithms: subgradient descent

The central algorithm we explore in this note is simple (sub)gradient descent on the dual problem (1.13):

Algorithm 1.3.1 (subgradient descent) Given $y^0 \in \mathbb{R}^m$, for $\nu = 0, 1, 2 \ldots$ generate the sequence $\{y^\nu\}_{\nu=0}^{\infty}$ via

$$y^{\nu+1} = y^\nu + \lambda_\nu d^\nu$$

where $d^\nu \in -\partial\left(\varphi^*_{\varepsilon,L}\left(A^T y^\nu\right) - b^T y^\nu\right)$ and λ_ν is an appropriate step length parameter.

For $\varepsilon > 0$, $\varphi^*_{\varepsilon,L}$ is continuously differentiable on its domain, and the algorithm amounts to the method of steepest descent.

1.3.1 Nonsmooth Case: $\varepsilon = 0$

In this section we present and analyze a subgradient descent method with exact line search and variants thereof suitable for solving the dual problem above for the case $\varepsilon = 0$, that is, we do not smooth the problem.

Using the notation of indicator functions, we have

$$\varphi^*_{0,L}(x) = \iota_{R_L}(x) \equiv \begin{cases} 0 & \text{for } x \in R_L \\ +\infty & \text{otherwise.} \end{cases}$$

The specific instance of (1.13) that we address is

$$\min_{y \in \mathbb{R}^m} \iota_{R_L}(A^T y) - y^T b. \tag{1.14}$$

Since the set R_L is a rectangle, nonsmooth calculus yields the following simple expression for the subdifferential of the dual objective:

$$\partial\left(\iota_{R_L}\left(A^T y^\nu\right) - b^T y^\nu\right) = A N_{R_L}\left(A^T y^\nu\right) - b. \tag{1.15}$$

Here we have used the fact that the subdifferential of the indicator function to the box R_L at the point x, denoted $\partial \iota_{R_L}(x)$ is equivalent to the normal cone mapping of R_L at the point x

$$\partial \iota_{R_L}(x) = N_{R_L}(x) \equiv \begin{cases} \{w \in \mathbb{R}^n \text{ with } (z-x)^T w \le 0 \text{ for all } z \in R_L\} & \text{if } x \in R_L \\ \emptyset & \text{if } x \notin R_L. \end{cases}$$

Remark 1.3.2 It is important to note that we assume that we can perform exact arithmetic. This assumption is necessary due to the composition of the normal cone mapping of R_L with A^T: while we can determine the exact evaluation of the normal cone for a given $A^T y^\nu$, we cannot guarantee exact evaluation of the matrix-vector product and, since the normal cone mapping is not Lipschitz continuous on R_L, this can lead to large computational errors. ■

Problem (1.14) is a linear programming problem. The algorithm we analyze below solves problem a *parametric* version of problem (1.14) where the parameter L changes dynamically at each iteration. To see how the parameter might be changed from one iteration to the next, we look to a trivial extension of the primal problem:

$$\begin{array}{ll} \underset{(x,L)\in\mathbb{R}^n\times\mathbb{R}^n_+}{\text{minimize}} & \sum_{j=1}^n L_j|x_j| \\ \text{subject to} & Ax=b \end{array} \tag{1.16}$$

It is clear that $L=0$ and any feasible x is an optimal solution to problem (1.16), and that the (global) optimal value is 0. However, this is not the only solution. Indeed, the sparsest solution x^* to $Ax=b$ and the weight L^* satisfying $L^*_j=0$ only for those elements j on the support of x^* is also a solution. The algorithm we study below finds a weight *compatible* with the sparsest element x^*. A more satisfying reformulation would yield a weight that is in some sense *optimal* for the sparsest element x^*, but this is beyond the scope of this work.

1.3.2 Dynamically Rescaled Descent with Exact Line Search

There are three unresolved issues in our discussion to this point, namely how to choose the element of the subdifferential, how to choose the step length and how to adjust the weights L_j. Our strategy is given in Algorithm 1.3.4 below. In the description of the algorithm we use some geometric notions that we introduce first. It will be convenient to define the set C by

$$C\equiv\{y\in\mathbb{R}^m \mid A^Ty\in R_L\}.$$

This set is polyhedral as the domain of a linear mapping with box constraints.

Lemma 1.3.3 (normal cone projection) *Let A be full rank and denote the normal cone to $C\equiv\{y\in\mathbb{R}^m \mid A^Ty\in R_L\}$ at $\overline{y}\in C$ by $N_C(\overline{y})$. Then*

$$P_{N_C(\overline{y})}b=A\overline{w} \tag{1.17}$$

for

$$\overline{w}=argmin\{\|Aw-b\|^2 \mid w\in N_{R_L}(A^T\overline{y})\}. \tag{1.18}$$

Proof. If A is full rank, then all points $y\in C$ satisfy the constraint qualification that A is injective on $N_{R_L}(A^Ty)$, that is, the only vector $w\in N_{R_L}(A^Ty)$ for which $Aw=0$ is $w=0$. Then by convex or nonsmooth analysis (see e.g., [14, Theorem 6.14]) the set C is *regular* and

$$N_C(y)=AN_{R_L}(A^Ty)=\left\{u=Aw \mid w\in N_{R_L}(A^Ty)\right\}.$$

By the definition of the projection

$$P_{N_C(\bar{y})}b \equiv \operatorname{argmin}\left\{\|u-b\|^2 \mid u \in N_C(\bar{y})\right\}$$

hence

$$\begin{aligned} P_{N_C(\bar{y})}b &= \operatorname{argmin}\left\{\|u-b\|^2 \mid u \in AN_{R_L}(A^T\bar{y})\right\} \\ &= A\operatorname{argmin}\left\{\|Aw-b\|^2 \mid w \in N_{R_L}(A^T\bar{y})\right\} = A\overline{w}. \end{aligned}$$

■

Algorithm 1.3.4 (Dynamically Rescaled Descent with Exact Line Search)

Initialization: Set $\nu = 0$, $\tau > 0$, $L^0 = (\|a_1\|, \|a_2\|, \ldots, \|a_n\|)$ where a_j is the jth column of A, $y^0 = 0$ and the direction $d^0 = b$.

Main iteration: While $\|d^\nu\| > \tau$ do

- (Exact line search.) Calculate the step length $\lambda_\nu > 0$ according to

$$\lambda_\nu \equiv \operatorname*{argmin}_{\lambda>0}\left\{\iota_{R_{L^\nu}}\left(A^T\left(y^\nu + \lambda d^\nu\right)\right) - b^T\left(y^\nu + \lambda d^\nu\right)\right\}. \tag{1.19}$$

  Set $y' = y^\nu + \lambda_\nu d^\nu$.
- (Subgradient selection and preliminary rescaling.) Define

$$\mathbb{J}^{\nu+1} = \{j \mid |a_j^T y'| = L_j^\nu\}, \tag{1.20}$$

$$\begin{aligned} S(L,\mathbb{J},\gamma) &= (s_1(L,\mathbb{J},\gamma), s_2(L,\mathbb{J},\gamma), \ldots, s_n(L,\mathbb{J},\gamma)) \\ \text{where} \qquad s_j(L,\mathbb{J},\gamma) &= \begin{cases} \gamma L_j & \text{for all } j \in \mathbb{J} \\ L_j & \text{else,} \end{cases} \end{aligned} \tag{1.21}$$

  and

$$\begin{aligned} &C(L,\mathbb{J},\gamma) = \{y \in \mathbb{R}^n \mid A^T y \in R_{S(L,\mathbb{J},\gamma)}\} \quad \text{where} \\ &R_{S(L,\mathbb{J},\gamma)} \equiv [-s_1(L,\mathbb{J},\gamma), s_1(L,\mathbb{J},\gamma)] \times \cdots \times [-s_n(L,\mathbb{J},\gamma), s_n(L,\mathbb{J},\gamma)] \end{aligned} \tag{1.22}$$

  Choose $\gamma' \geq 0$ small enough that $P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \in ri(N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y''))$ for $y'' = \gamma' y'$. Compute the direction

$$d^{\nu+1} \equiv b - P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \tag{1.23}$$

- (Rescaling.) Let

$$\mathbb{J}_+^{\nu+1} \equiv \left\{j \mid a_j^T d^{\nu+1} > 0\right\}, \qquad \mathbb{J}_-^{\nu+1} \equiv \left\{j \mid a_j^T d^{\nu+1} < 0\right\},$$

  and define

$$\mathbb{I}^{\nu+1}(\gamma) \equiv \underset{j\in\mathbb{J}_+^{\nu+1}\cup\mathbb{J}_-^{\nu+1}}{\operatorname{argmin}} \left\{ \left\{ \frac{L_j^\nu - \gamma a_j^T y'}{a_j^T d^{\nu+1}} \;\middle|\; j\in\mathbb{J}_+^{\nu+1} \right\}, \left\{ \frac{-L_j^\nu - \gamma a_j^T y'}{a_j^T d^{\nu+1}} \;\middle|\; j\in\mathbb{J}_-^{\nu+1} \right\} \right\}. \quad (1.24)$$

Choose $\gamma^{\nu+1}\in[0,\gamma']$ to satisfy

$$\mathbb{I}^{\nu+1}(\gamma^{\nu+1}) \subset \mathbb{I}^{\nu+1}(0). \quad (1.25)$$

Set

$$L_j^{\nu+1} = \begin{cases} \gamma^{\nu+1} L_j^\nu & \text{for } j\in\mathbb{J}^{\nu+1} \\ L_j^\nu & \text{else} \end{cases} \quad (1.26)$$

and $y^{\nu+1}=\gamma^{\nu+1}y'$. Increment $\nu=\nu+1$.

End do.

We begin with some observations. The next proposition shows that the directions chosen by (1.23) with $P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \in ri(N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y''))$ for $y''=\gamma' y'$ are subgradient descent directions that are not only feasible, but orthogonal to the active constraints. We use orthogonality of the search directions to the active constraints to guarantee finite termination of the algorithm.

Proposition 1.3.5 (feasible directions) *Let $C\equiv\{y\in\mathbb{R}^m \mid A^Ty\in R_L\}$, $\bar y\in C$ and define the direction $\bar d\equiv b-P_{N_C(\bar y)}b$. Then $-\bar d\in\partial\left(\iota_{R_L}(A^T\bar y)-b^T\bar y\right)$ and there exists a $\overline{\lambda}>0$ such that $\bar y+\lambda\bar d\in C$ for all $\lambda\in[0,\overline{\lambda}]$.*

Moreover, if $P_{N_C(\bar y)}b\in ri(N_C(\bar y))$, then the direction $\bar d$ is orthogonal to the jth column of A for all j such that $a_j^T\bar y=L_j$.

Proof. The inclusion $-\bar d\in\partial\left(\iota_{R_L}(A^T\bar y)-b^T\bar y\right)$ follows immediately from (1.15). The feasibility of this direction follows from Lemma 1.3.3 and the polyhedrality of C since the polar to the normal cone to C at a point $y\in C$ is therefore equivalent to the tangent cone, which consists only of *feasible directions* to C at y, defined as a direction d for which $\bar y+\lambda d\in C$ for all $\lambda>0$ sufficiently small.

Indeed, let a_j denote the jth column of the matrix A and recall the definition of the *contingent cone* to C at $y\in C$:

$$K_C(y)\equiv\{w\in\mathbb{Y}\mid \text{ for all } \nu\ y+\lambda^\nu w^\nu\in C \text{ for some } w^\nu\to w,\ \lambda^\nu\searrow 0\}.$$

Since C is convex the contingent cone and the tangent cone are equivalent [2, Corollary 6.3.7] and since C is polyhedral the tangent cone can be written as

$$T_C(y)\equiv\{w\in\mathbb{Y}\mid \text{ for all } \nu\ y+\lambda^\nu w\in C \text{ for some } \lambda^\nu\searrow 0\},$$

that is, the tangent cone consists entirely of feasible directions. Now the tangent and normal cones to C are convex and polar to each other [14, Corollary 6.30], so, by

Lemma 1.3.3, what remains to be shown is that $b - P_{N_C(\bar{y})}b$ lies in the polar to the normal cone to C. This follows since $N_C(y)$ is nonempty closed and convex. Hence for all $w \in N_C(\bar{y})$ and for any b

$$w^T(b - P_{N_C(\bar{y})}b) \leq 0,$$

that is, $b - P_{N_C(\bar{y})}b$ is in the polar to the normal cone.

To see the final statement of the proposition, denote by $\bar{\mathbb{J}}$ the set

$$\{j = 1,2,\ldots,n \mid a_j^T \bar{y} = L_j\}.$$

If the projection lies on the relative interior to $N_C(\bar{y})$, then the projection onto $N_C(\bar{y})$ is equivalent to the projection onto the subspace containing $N_C(\bar{y})$:

$$P_{N_C(\bar{y})}b = P_{D(\bar{y})}b$$

where

$$D(\bar{y}) \equiv \left\{Aw \mid w \in \mathbb{R}^n \text{ with } w_j = 0 \text{ for } j \notin \bar{\mathbb{J}}\right\}.$$

Thus $a_j^T(b - P_{N_C(\bar{y})}b) = a_j^T(b - P_{N_D(\bar{y})}b) = 0$ for $j \in \bar{\mathbb{J}}$ as claimed. ■

Remark 1.3.6 (detection of orthogonality of feasible directions) The interiority condition $P_{N_C(\bar{y})}b \in ri(N_C(\bar{y}))$ guaranteeing orthogonality of the directions can easily be checked. Let $\bar{w} \equiv \operatorname{argmin}\{\|Aw - b\|^2 \mid w \in N_{R_L}(A^T\bar{y})\}$. By Lemma 1.3.3 $P_{N_C(\bar{y})}b = A\bar{w}$. Then $A\bar{w}$ and hence $P_{N_C(\bar{y})}b$ lies in $ri(N_C(\bar{y}))$ if and only if $\bar{w}_j \neq 0$ for all j such that $a_j^T\bar{y} = L_j$. ■

Calculation of the direction in (1.23) of Algorithm 1.3.4 is suggested by Lemma 1.3.3 where it is shown that the projection is the mapping of the solution to a least squares problem over a cone. Also, by Proposition 1.3.5, the direction is the negative of a subgradient of the objective in (1.14) with the box $S(L^\nu, \mathbb{J}^{\nu+1}, \gamma')$, that is

$$-d^{\nu+1} \equiv -b + P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \in \partial\left(\iota_{R_{S(L^\nu,\mathbb{J}^{\nu+1},\gamma')}}(A^Ty'') - b^Ty''\right)$$

The description as a projection onto the normal cone of a polyhedron is perhaps less helpful than the explicit formulation of Lemma 1.3.3 for suggesting how this can be computed, but it provides greater geometrical insight. Moreover, the projection provides an elegant criterion for maintaining orthogonality of the search directions with the active constraints.

The exact line search step has an explicit formulation given in the next proposition.

Proposition 1.3.7 (exact line search) *Let $\bar{y} \in C$ and $\bar{d} = b - P_{N_C(\bar{y})}b$. Define the index sets*

$$\bar{\mathbb{J}}_+ \equiv \left\{j \mid a_j^T\bar{d} > 0\right\}, \qquad \bar{\mathbb{J}}_- \equiv \left\{j \mid a_j^T\bar{d} < 0\right\}.$$

The exact line search step length $\bar{\lambda}$ given by (1.19) *has the explicit representation*

$$\bar{\lambda} \equiv \min \left\{ \min_{j \in \bar{\mathbb{J}}_+} \left\{ \frac{L_j - a_j^T \bar{y}}{a_j^T \bar{d}} \right\}, \min_{j \in \bar{\mathbb{J}}_-} \left\{ \frac{-L_j - a_j^T \bar{y}}{a_j^T \bar{d}} \right\} \right\} > 0. \tag{1.27}$$

Proof. Application of nonsmooth calculus provides a generalization to the fact from optimization of smooth objectives that the exact line search step extends to the tangent of a level set of the objective, from which we can extract (1.27). However, it is perhaps easiest to see the explicit formulation by direct inspection: the indicator function ι_{R_L} is zero at all points in R_L, so the step length is the largest λ such that $A^T\left(\bar{y} + \lambda \bar{d}\right) \in R_L$, i.e. the largest λ such that

$$a_j^T\left(\bar{y} + \lambda \bar{d}\right) \leq L_j \quad \text{for all } j \in \bar{\mathbb{J}}_+.$$

and

$$a_j^T\left(\bar{y} + \lambda \bar{d}\right) \geq -L_j \quad \text{for all } j \in \bar{\mathbb{J}}_-.$$

Note that by Proposition 1.3.5 it is not possible to have $a_j^T \bar{d} > 0$ and $a_j^T \bar{y} = L_j$ or, similarly $a_j^T \bar{d} < 0$ and $a_j^T \bar{y} = -L_j$, hence the step length is guaranteed to be positive, and we are done. ■

1.4 Convergence to Sparse Solutions

We show in this section that for sufficiently sparse solutions x^* to $Ax = b$, the steepest subgradient descent algorithm with exact line search (Algorithm 1.3.4) recovers x^* exactly. Before we continue, however, we must specify precisely what is meant by "sufficiently sparse".

Definition 1.4.1 (mutual coherence) *Let a_j denote the jth column of A. The* mutual coherence *of A is defined as*

$$\mu(A) \equiv \max_{1 \leq k,j \leq n,\ k \neq j} \frac{|a_k^T a_j|}{\|a_k\| \|a_j\|}$$

where $0/0 \equiv 1$.

The mutual coherence characterizes the dependence between columns of A. The mutual coherence of unitary matrices, for instance, is zero; for matrices with columns of zeros, the mutual coherence is 1.

Lemma 1.4.2 (uniqueness of sparse representations [8]) *Let $A \in \mathbb{R}^{m \times n}$ $(m < n)$ be full rank. If there exists an element x^* such that $Ax^* = b$ and*

$$\|x^*\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(A)}\right), \tag{1.28}$$

then it is unique and sparsest possible (has minimal support).

In the case of matrices that are not full rank – and thus unitarily equivalent to matrices with columns of zeros – only the trivial equation $Ax = 0$ has a unique sparsest possible solution. For unitary matrices $\mu(A) = 0$, we interpret $1/0 = +\infty$.

The sparsity condition of Lemma 1.4.2 yields a more direct representation that will be useful later.

Lemma 1.4.3 (sparsity conditions) *Let $A \in \mathbb{R}^{m\times n}$ ($m < n$) be full rank. For $b \in \mathbb{R}^m \setminus \{0\}$ given and x^* a solution to $Ax = b$, define $\mathbb{J} = \{j \mid x^*_j \neq 0\}$ and denote by $J \in \mathbb{J}$ an element of x^* satisfying*

$$|x^*_J|\|a_J\| \geq |x^*_j|\|a_j\| \quad \textit{for all} \quad j = 1,2,\ldots,n.$$

If the solution x^* satisfies condition (1.28) *then there exists a $\bar{\gamma} > 0$ such that, for all $y \in \mathbb{B} \equiv \{y \in \mathbb{R}^m \mid \|y\| = 1\}$ and all $\gamma \in [0, \bar{\gamma}]$*

$$\max_{k\notin\mathbb{J}} \frac{|a_k^T b|}{\|a_k\| - \gamma|a_k^T y|} < \frac{|a_J^T b|}{\|a_J\| + \gamma|a_J^T y|}. \tag{1.29}$$

Proof. We use continuity of the terms in (1.29) with respect to γ and y to simplify the operative inequality and prove the statement for the case $\gamma = 0$.

Reduction to the case $\gamma = 0$.

For all $\bar{\gamma}$ small enough the function

$$g(y,\gamma) \equiv \max_{k\notin\mathbb{J}} \frac{|a_k^T b|}{\|a_k\| - \gamma|a_k^T y|}$$

is a continuous function on the compact domain $\mathbb{B} \times [0, \bar{\gamma}]$. Likewise, for *any* $\bar{\gamma} > 0$ the function

$$h(y,\gamma) \equiv \frac{|a_J^T b|}{\|a_J\| + \gamma|a_J^T y|}$$

is continuous. By continuity, the existence of $\bar{\gamma} > 0$ such that (1.29) holds for all $\gamma \in [0, \bar{\gamma}]$ and $y \in \mathbb{B}$ is then equivalent to

$$g(y,0) = \max_{k\notin\mathbb{J}} \frac{|a_k^T b|}{\|a_k\|} < \frac{|a_J^T b|}{\|a_J\|} = h(y,0). \tag{1.30}$$

We therefore limit our attention to (1.30).

Reformulation of (1.28)**.**

Starting with (1.28) we have

$$\begin{aligned}
\|x^*\|_0 = |\mathbb{J}| &< \frac{1}{2}\left(\frac{1}{\mu(A)}+1\right) \\
&\iff \\
\mu(A)|\mathbb{J}| &< \frac{1}{2}(1+\mu(A)) \\
&\iff \\
|x_J^*|\|a_J\||\mathbb{J}|\mu(A) &< \frac{1}{2}|x_J^*|\|a_J\|(1+\mu(A)) \\
&\iff \\
|x_J^*|\|a_J\||\mathbb{J}|\mu(A) &< |x_J^*|\|a_J\|(1+\mu(A)(1-|\mathbb{J}|)). \qquad (1.31)
\end{aligned}$$

Here we have denoted the cardinality of $\mathbb{J}$ by $|\mathbb{J}|$.

Upper and lower bounds.

It remains to show that the left hand side of (1.31) is an upper bound for the left hand side of (1.30) and, similarly, that the right hand side of (1.31) is a lower bound for the right hand side of (1.30).

Substituting Ax^* for b in (1.30) yields the equivalent statement

$$\frac{|\sum_{i\in\mathbb{J}} x_i^* a_k^T a_i|}{\|a_k\|} < \frac{|\sum_{i\in\mathbb{J}} x_i^* a_J^T a_i|}{\|a_J\|} \quad \text{for all } k \notin \mathbb{J}. \qquad (1.32)$$

For the lower bound, we have

$$\begin{aligned}
\frac{|\sum_{i\in\mathbb{J}} x_i^* a_J^T a_i|}{\|a_J\|} &\geq |x_J^*|\|a_J\| - \sum_{i\in\mathbb{J}\setminus\{J\}} \frac{|x_i^*||a_J^T a_i|}{\|a_J\|} \\
&\geq |x_J^*|\|a_J\| - \sum_{i\in\mathbb{J}\setminus\{J\}} |x_i^*|\|a_i\|\mu(A) \\
&\geq |x_J^*|\|a_J\|(1-(|\mathbb{J}|-1)\mu(A)).
\end{aligned}$$

In summary

$$|x_J^*|\|a_J\|(1+(1-|\mathbb{J}|)\mu(A)) \leq \frac{|\sum_{i\in\mathbb{J}} x_i^* a_J^T a_i|}{\|a_J\|}. \qquad (1.33)$$

For the upper bound we have

$$\begin{aligned}
\frac{|\sum_{i\in\mathbb{J}} x_i^* a_k^T a_i|}{\|a_k\|} &\leq \sum_{i\in\mathbb{J}} \frac{|x_i^*|\,|a_k^T a_i|}{\|a_k\|} \\
&\leq \sum_{i\in\mathbb{J}} |x_i^*|\,\|a_i\|\mu(A) \\
&\leq |x_J^*|\,\|a_J\||\mathbb{J}|\mu(A)
\end{aligned}$$

or

$$\frac{|\sum_{i\in\mathbb{J}} x_i^* a_k^T a_i|}{\|a_k\|} \leq |x_J^*|\,\|a_J\||\mathbb{J}|\mu(A) \qquad (1.34)$$

Inequality (1.31) together with (1.32), (1.33) and (1.34) yield (1.30). By the continuity argument at the beginning of the proof, we have thus shown that (1.28) implies (1.29) as claimed. ■

The next lemma provides a sufficient condition for monotonicity of the cardinality of the set of active indices from one iteration of Algorithm 1.3.4. This is an important feature for the finite termination of Algorithm 1.3.4 proved in Theorem 1.4.5.

Lemma 1.4.4 (step length) *For a given $L = (L_1, L_2, \ldots, L_n)$ and the corresponding sets R_L and $C \equiv \{y \in \mathbb{R}^m \mid A^T y \in R_L\}$, let the point $\bar{y} \in C$ satisfy $P_{N_C(\bar{y})} b \in ri(N_C(\bar{y}))$. For this point define $\bar{d} \equiv b - P_{N_C(\bar{y})} b$ and the index sets $\mathbb{J} = \{j \mid a_j^T \bar{y} = L_j\}$*

$$\mathbb{J}_+ \equiv \left\{ j \mid a_j^T \bar{d} > 0 \right\}, \qquad \mathbb{J}_- \equiv \left\{ j \mid a_j^T \bar{d} < 0 \right\}.$$

Then $(\mathbb{J}_+ \cup \mathbb{J}_-) \cap \mathbb{J} = \emptyset$ and for the step length given by (1.27) *the set of active indices set is increasing, that is, $\mathbb{J} \subset \mathbb{J}' = \{j \mid a_j^T (\bar{y} + \bar{\lambda} \bar{d}) = L_j\}$.*

In the special case that $\bar{y} = 0$, then the step length $\bar{\lambda}$ is given by

$$\bar{\lambda} \equiv \min_{j \notin \mathbb{J}} \left\{ \frac{L_j}{|a_j^T \bar{d}|} \right\}. \tag{1.35}$$

Proof. By Proposition 1.3.5 and Remark 1.3.6, if $P_{N_C(\bar{y})} b \in ri(N_C(\bar{y}))$ then $\bar{d}$ is orthogonal to the columns of A corresponding to the set of active indices $\mathbb{J}$. Thus $(\mathbb{J}_+ \cup \mathbb{J}_-) \cap \mathbb{J} = \emptyset$ as claimed. It follows immediately from (1.27) that $\mathbb{J} \subset \mathbb{J}' = \{j \mid a_j^T (\bar{y} + \bar{\lambda} \bar{d}) = L_j\}$ since $\bar{\lambda}$ is computed from the elements belonging to $\mathbb{J}_+ \cup \mathbb{J}_-$, and, again by Proposition 1.3.5, the active constraints corresponding to $\mathbb{J}$ remain unchanged in the direction $\bar{d}$.

When $\bar{y} = 0$ the step length given by (1.27) simplifies to

$$\bar{\lambda} = \min_{j \in \bar{\mathbb{J}}_+ \cup \bar{\mathbb{J}}_-} \left\{ \frac{L_j}{|a_j^T \bar{d}|} \right\} > 0. \tag{1.36}$$

Hence (1.36) is equivalent to (1.35). This completes the proof. ■

We are now ready to state and prove the main result of this section, the convergence of Algorithm 1.3.4 for a particular choice of initial weights $L_j^0 = \|a_j\|$ for $j = 1, 2, \ldots, n$. Theorem 1.4.5 says that the algorithm finds a point y^* and a weight L^* for which $0 \in \partial \left(\iota_{R_{L^*}} (A^T y^*) - (y^*)^T b \right)$ exactly (tolerance $\tau = 0$), as opposed to finding a point where the chosen subgradient is smaller than some tolerance. Since the problem is convex, this is sufficient for optimality. Of course, this is only possible with exact arithmetic.

Theorem 1.4.5 (exact recovery of sufficiently sparse solutions) *Let $A \in \mathbb{R}^{m \times n}$ $(m < n)$ be full rank and denote the jth column of A by a_j. Initialize Algorithm 1.3.4 with initial guess y^0 and weight L^0 such that $y_j^0 = 0$ and $L_j^0 = \|a_j\|$ for $j = 1, 2, \ldots, n$.*

If an element $x^* \in \mathbb{R}^n$ with $Ax^* = b$ satisfies (1.28)*, then, with tolerance $\tau = 0$, Algorithm 1.3.4 converges in finitely many steps to a point y^* and a weight L^* where,*

$$argmin\{\|Aw-b\|^2 \mid w \in N_{R_{L^*}}(y^*)\} = x^*,$$

the unique sparsest solution to $Ax = b$.

Proof. The proof is by induction and follows a pattern similar to the convergence proof of the orthogonal matching pursuit algorithm [4, Theorem 6], though the details are more technical. (Indeed, we show in Section 1.4.1 below that this is no coincidence.) In order to facilitate the proof, we will in fact prove convergence of a slightly more general procedure than Algorithm 1.3.4. The difference is in the initialization. Rather than initializing $y^0 = 0$, as any practical method would do, we will choose an arbitrary $y^0 = \gamma^0 y$ for any fixed vector y with $\gamma^0 \geq 0$ small enough. This allows us to establish the pattern for later iterations at the very beginning.

Let $C^0 \equiv \{y \in \mathbb{R}^m \mid A^T y \in R_{L^0}\}$. The open unit ball lies in the (relative) interior of C^0 since, for any y with $\|y\| < 1$, we have $|(A^T y)_j| \leq \|a_j\|\|y\| \leq \|a_j\| = L_j^0$ with the last inequality strict if $a_j \neq 0$. (Without loss of generality, we can assume that A has no zero columns.) Then $N_{C^0}(y^0) = \{0\}$, so that $P_{N_{C^0}(y^0)}b = 0$ and $d^0 = b$ is in fact a direction of (subgradient) descent according to Proposition 1.3.5 for any y^0 small enough.

Identifying the active constraints.

Computing the step length, by (1.27) we have

$$\lambda_0 \equiv \min\left\{ \min_{j\in\mathbb{J}_+^0}\left\{\frac{\|a_j\| - \gamma^0 a_j^T y}{a_j^T b}\right\}, \min_{j\in\mathbb{J}_-^0}\left\{\frac{-\|a_j\| - \gamma^0 a_j^T y}{a_j^T b}\right\}\right\} > 0. \tag{1.37}$$

where, recall, $\gamma^0 y = y^0$, and

$$\mathbb{J}_+^0 = \{j \mid a_j^T b > 0\} \quad \text{and} \quad \mathbb{J}_-^0 = \{j \mid a_j^T b < 0\}.$$

Let j_0 be the index of a minimum element of the set above. We show that, for any choice of minimum element (in the case that there is more than one) $j_0 \in \mathbb{J}^* \equiv \{j \mid x_j^* \neq 0\}$. In other words, we show that

$$|a_k^T y^0 + \lambda_0 a_k^T b| < \|a_k\| \quad \text{for all } k \notin \mathbb{J}^*. \tag{1.38}$$

By the triangle inequality, (1.38) holds if

$$|a_k^T y^0| + |\lambda_0 a_k^T b| < \|a_k\| \quad \text{for all } k \notin \mathbb{J}^*. \tag{1.39}$$

Expanding λ_0 and rearranging terms in (1.39) yields, for γ^0 small enough,

$$\frac{|a_k^T b|}{\|a_k\| - \gamma^0 |a_k^T y|} < \frac{1}{\lambda_0} = \begin{cases} \frac{|a_{j_0}^T b|}{\left|\|a_{j_0}\| - \gamma^0 a_{j_0}^T y\right|}, & \text{if } j_0 \in \mathbb{J}_+^0 \\ \frac{|a_{j_0}^T b|}{\left|-\|a_{j_0}\| - \gamma^0 a_{j_0}^T y\right|}, & \text{if } j_0 \in \mathbb{J}_-^0 \end{cases} \quad \text{for all } k \notin \mathbb{J}^*. \quad (1.40)$$

Let $J \in \mathbb{J}^*$ be the index of an element of x^* satisfying

$$|x_J^*| \|a_J\| \geq |x_j^*| \|a_j\| \quad \text{for all} \quad j = 1, 2, \ldots, n.$$

By definition of λ_0,

$$\frac{|a_J^T b|}{\|a_J\| + \gamma^0 |a_J^T y|} \leq \left\{ \begin{array}{l} \frac{|a_J^T b|}{\left|\|a_J\| - \gamma^0 a_J^T y\right|}, \quad \text{if } J \in \mathbb{J}_+^0 \\ \frac{|a_J^T b|}{\left|-\|a_J\| - \gamma^0 a_J^T y\right|}, \quad \text{if } J \in \mathbb{J}_-^0 \end{array} \right\} \leq \frac{1}{\lambda_0}.$$

By Lemma 1.4.3, the sparsity condition (1.28) implies (1.29) which immediately yields (1.40), and hence (1.38), for γ^0 small enough.

Letting $y' = \gamma^0 y + \lambda_0 b$, we conclude that, as (1.28) holds, then for γ^0 small enough (as it certainly would be for the initial guess of zero)

$$\mathbb{J}^1 \equiv \{ j \mid |a_j^T y'| = \|a_j\| = L_j^0 \} \cap \mathbb{J}^* \neq \emptyset$$

where $\mathbb{J}^\nu$ is defined by (1.20).

The question remains as to how small γ^0 need be. For this we refer to the index set $\mathbb{I}^0(\gamma)$ defined by (1.24) with $L^{-1} \equiv L^0$. Note that this is just the set of indices of active faces in $\mathbb{J}_+^0 \cup \mathbb{J}_+^0$ corresponding to the exact line search step length λ_0 computed by (1.37). Viewed as a function, λ^0 is the minimum of a finite collection of affine functions of γ^0 and is thus a continuous function of γ^0. Moreover, the set of indices corresponding to the affine functions at which the minimum is attained, $\mathbb{I}(\gamma^0)$, satisfies $\mathbb{I}(\gamma^0) \subset \mathbb{I}(0)$ on a neighborhood of 0. In other words, the index j_0 of the minimum element at which the exact step length λ_0 is attained belongs to $\mathbb{I}^0(0)$ for all γ^0 small enough. This yields an implementable strategy for determining the proper scaling in subsequent iterations by checking the coincidence of the set of active indices $\mathbb{I}^\nu(\gamma)$ with the set of faces reached from the origin, $\mathbb{I}^\nu(0)$.

Subgradient selection. There always exists $\gamma' \geq 0$ with $y'' = \gamma' y'$ such that $P_{N_{C(L^0, \mathbb{J}^1, \gamma')}(y'')} b \in ri(N_{C(L^0, \mathbb{J}^1, \gamma')}(y''))$ since for $\gamma' = 0$ the normal cone to $C(L^0, \mathbb{J}^1, 0)$ at $y'' = 0$ defined by (1.22) is the subspace

$$N_{C(L^0, \mathbb{J}^1, 0)}(0) = \left\{ Aw \;\middle|\; \begin{cases} w_j \in \mathbb{R} & \text{for } j \in \mathbb{J}^1 \\ w_j = 0 & \text{for } j \notin \mathbb{J}^1 \end{cases} \right\}.$$

This follows from the fact that the only active faces of the polyhedron $C(L^0, \mathbb{J}^1, 0)$ at the origin are the ones corresponding to the point [0] (the degenerated interval). Thus, at least for $\gamma' = 0$, the projection of b onto the subspace spanned by

the columns of A corresponding to $\mathbb{J}^1$ is equivalent to $P_{N_{C(L^0,\mathbb{J}^1,0)}(0)}b$. By Proposition 1.3.5, then, for γ' small enough (possibly zero) the direction of descent $d^1 \equiv b - P_{N_{C(L^0,\mathbb{J}^1,\gamma')}(y'')}b$ is orthogonal to the columns of A corresponding to the index set $\mathbb{J}^1$.

Rescaling. For the choice of γ' above, we have $\left(\mathbb{J}^1_+ \cup \mathbb{J}^1_-\right) \cap \mathbb{J}^1 = \emptyset$ where

$$\mathbb{J}^1_+ \equiv \left\{ j \mid a_j^T d^1 > 0 \right\}, \qquad \mathbb{J}^1_- \equiv \left\{ j \mid a_j^T d^1 < 0 \right\}.$$

There are two cases to consider: $\gamma' = 0$ and $\gamma' > 0$. If $\gamma' = 0$, then $\gamma^1 = 0$ and by Lemma 1.4.4

$$\mathbb{I}^1(0) = \underset{j \notin \mathbb{J}^1_0}{\operatorname{argmin}} \left\{ \frac{L^0_j}{|a_j^T d^1|} \right\},$$

so that

$$L^1_j = \begin{cases} 0 & \text{for all } j \in \mathbb{I}^1(0) \\ L^0_j & \text{else} \end{cases}$$

and $y^1 = 0$.

If, on the other hand, $\gamma' > 0$, the previous argument shows that there exists at least *some* $\gamma^1 \in [0, \gamma']$ such that $\mathbb{I}^1(\gamma^1) \subset \mathbb{I}^1(0)$, which is sufficient for our purposes.

With γ^1 in hand, we set the weights

$$L^1_j = \begin{cases} \gamma^1 L^0_j & \text{for all } j \in \mathbb{I}^1(\gamma^1) \\ L^0_j & \text{else.} \end{cases}$$

and update the iterate $y^1 = \gamma^1 y'$ as prescribed.

Note that y^1 is feasible and the set of active faces $\mathbb{J}^1$ is unchanged since $a_j^T y^1 = a_j^T \gamma^1 y'$ with $a_j^T \gamma^1 y' = \gamma^1 L^0_j = L^1_j$ for all $j \in \mathbb{J}^1$, and $a_j^T \gamma^1 y' < a_j^T y' < L^1_j$ otherwise.

Induction. Proceeding now by induction, we suppose for $\nu \geq 0$ that $a_j^T y^\nu = L^\nu_j$ for all $j \notin \mathbb{J}^\nu \subset \mathbb{J}^*$ and that $|a_j^T y^\nu| < L^\nu_j = \|a_j\|$ for all $j \notin \mathbb{J}^\nu$ where $\nu \leq |\mathbb{J}^\nu| \leq |\mathbb{J}^*|$. We show that there are only two possibilities for the next iteration: either $d^{\nu+1} = 0$, in which case $\mathbb{J}^{\nu+1} = \mathbb{J}^*$ and $w^{\nu+1} = x^*$; or $d^{\nu+1} \neq 0$, in which case $\mathbb{J}^{\nu+1} \subset \mathbb{J}^*$ with $|\mathbb{J}^{\nu+1}| < |\mathbb{J}^{\nu+2}| \leq |\mathbb{J}^*|$ and $|a_j^T y^{\nu+1}| = L^{\nu+1}_j$ for $j \in \mathbb{J}^{\nu+2}$ and $|a_j^T y^\nu| < L^{\nu+1}_j$ for $j \notin \mathbb{J}^{\nu+2}$.

In either case, in a somewhat awkward consequence of our indexing, note that for γ^ν satisfying (1.25) and the induction hypothesis we have that $\mathbb{J}^{\nu+1} \subset \mathbb{J}^*$. Our task is to show that $\mathbb{J}^{\nu+2} \subset \mathbb{J}^*$

Case 1: $d^{\nu+1} = 0$**.** In this case, we have

$$b = P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \in ri(N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y''))$$

for $y'' = \gamma' y'$ with $y' = y^\nu + \lambda^\nu d^\nu$ and, by assumption (1.28),

$$\mathbb{J}^{\nu+1} = \{ j \mid |a_j^T(y^\nu + \lambda^\nu d^\nu)| = L_j^\nu \} \subset \mathbb{J}^*.$$

Also note that $\mathbb{J}_+^{\nu+1} = \emptyset$, $\mathbb{J}_-^{\nu+1} = \emptyset$ because $d^{\nu+1} = 0$ and hence $\mathbb{I}^{\nu+1}(\gamma) = \emptyset$ for all $\gamma \geq 0$. So without any calculation one can choose $\gamma^{\nu+1} = \gamma'$ and determine $L^{\nu+1}$ according to (1.26) and $y^{\nu+1} = \gamma^{\nu+1} y'$. Define $C^* \equiv \{ y \mid A^T y \in R_{L^{\nu+1}} \}$. Then $d^{\nu+1} = 0 \in \partial \left(\iota_{C^*}(A^T y^{\nu+1}) - b^T y^{\nu+1} + \iota_{\mathbb{R}^n_+}(L^{\nu+1}) \right)$ and $y^{\nu+1}$ for $L^{\nu+1}$ defined by (1.26) is a fixed point of the iteration. By the definition of the subdifferential (1.4), $y^{\nu+1}$ is an optimal solution to (1.14). The corresponding subgradient

$$w^{\nu+1} \equiv \operatorname{argmin} \left\{ \|Aw - b\|^2 \mid w \in N_{R_{L^{\nu+1}}}(y^{\nu+1}) \right\}$$

satisfies $Aw^{\nu+1} = b$ and is supported on $\mathbb{J}^{\nu+1} \subset \mathbb{J}^*$. Lemma 1.4.2 shows that x^* is the unique sparsest solution to $Ax = b$. Thus, $\mathbb{J}^{\nu+1} = \mathbb{J}^*$ and $w^{\nu+1} = x^*$ as claimed.

Case 2: $d^{\nu+1} \neq 0$**.** In this case $b \notin N_{C(L^\nu, \mathbb{J}^{\nu+1}, \gamma')}(y'')$, and it must be that $|\mathbb{J}^{\nu+1}| < |\mathbb{J}^*|$. By the induction hypothesis $\mathbb{J}^{\nu+1} \subset \mathbb{J}^*$. By the choice of γ' we have

$$P_{N_{C(L^\nu, \mathbb{J}^{\nu+1}, \gamma')}(y'')} b \in ri(N_{C(L^\nu, \mathbb{J}^{\nu+1}, \gamma')}(y''))$$

and thus by Lemma 1.4.4 $\left(\mathbb{J}_+^{\nu+1} \cup \mathbb{J}_-^{\nu+1} \right) \cap \mathbb{J}^{\nu+1} = \emptyset$ and the active set is monotonically increasing, so we must show that $\mathbb{J}^{\nu+2} \subset \mathbb{J}^*$.

We continue to the rescaling step to find $\gamma^{\nu+1}$ satisfying (1.25). Since by construction $d^{\nu+1}$ is orthogonal to the columns a_j with $j \in \mathbb{J}^{\nu+1}$, we can deflate the matrix A to contain only those columns with indices not in $\mathbb{J}^{\nu+1}$. The weights corresponding to the remaining indices, denoted $\bar{L}^{\nu+1}$, are unchanged from the initialization, that is, $L_j^\nu = \|a_j\|$ for $j \notin \mathbb{J}^{\nu+1}$ and so the elements of $\bar{L}^{\nu+1}$ are just the norms of the remaining columns of the deflated matrix $A^{\nu+1}$. Repeating the argument for the first iteration with b replaced by $d^{\nu+1}$, condition (1.28) with $\gamma^{\nu+1}$ satisfying (1.25) guarantees that $|y^{\nu+1} + \lambda_{\nu+1} d^{\nu+1}| = \|a_j\| = \bar{L}_j^{\nu+1}$ for some j corresponding to an element of $\mathbb{J}^* \setminus \mathbb{J}^{\nu+1}$, while $|y^{\nu+1} + \lambda_{\nu+1} d^{\nu+1}| < \|a_j\| = L_j^\nu$ for j corresponding to the complement of $\mathbb{J}^*$. (Note that because of the deflation technique, the correspondence between these indices is not direct.) Defining $y' = y^{\nu+1} + \lambda_{\nu+1} d^{\nu+1}$ $\mathbb{J}^{\nu+2} \equiv \{ j \mid |a_j^T y'| = L^{\nu+1} \}$, by orthogonality and rescaling of the previous weights we have that $\mathbb{J}^{\nu+2} \subset \mathbb{J}^*$ and $|\mathbb{J}^{\nu+1}| < |\mathbb{J}^{\nu+2}| \leq |\mathbb{J}|$, as claimed.

Since the cardinality of the active set increases strictly monotonically with each iteration, the algorithm is finitely terminating as asserted. ■

The next corollary is an immediate consequence of Theorem 1.4.5. We will show in the next section that the corollary is actually a statement of finite termination of the orthogonal matching pursuit algorithm [4, Theorem 6].

Corollary 1.4.6 (greedy rescaling) *Let $A \in \mathbb{R}^{m \times n}$ ($m < n$) be full rank and denote the jth column of A by a_j. Initialize Algorithm 1.3.4 with initial guess $y_j^0 = 0$ and*

$L_j^0 = \|a_j\|$ *for* $j = 1,2,\ldots,n$, *and at the rescaling step choose* $L_j^{\nu+1} = \gamma^{\nu+1} = 0$ *for all* $j \in \mathbb{J}^{\nu+1}$.

If a point x^* *solves* $Ax = b$ *and satisfies* (1.28) *then, with tolerance* $\tau = 0$, *Algorithm 1.3.4 converges in finitely many steps to* $y^* = 0$ *with the weight* L^* *where,*

$$argmin\{\|Aw - b\|^2 \mid w \in N_{R_{L^*}}(0)\} = x^*,$$

the unique sparsest solution to $Ax = b$.

Remark 1.4.7 We have called the rescaling strategy of Corollary 1.4.6 *greedy* to conform with precedent, however in light of the variational derivation that we have developed here, we would prefer to use the descriptor *dogmatic*. To see why we prefer this, note that when the scaling of the active indices is set to zero, these elements are forever "committed" to the active set, even if in later iterations it might be determined that this was an error for some elements. In our algorithm the detection of a possible error would occur in the determination of the preliminary scaling stage. If $P_{N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y'')}b \in ri(N_{C(L^\nu,\mathbb{J}^{\nu+1},\gamma')}(y''))$ only for $\gamma' = 0$ this is an indication that the direction of descent will cause a sign change in one of the active elements.

If the scaling is bounded away from 0, then the orthogonality of the descent directions with the active columns of A, see Proposition 1.3.5, is no longer guaranteed and the strict monotonicity of the cardinality of the active set Lemma 1.4.4 is also lost. This reflects the fact that, in this case, the algorithm can "change its mind" about the active set, that is, it has *recourse*. The more general Algorithm 1.3.4 is, in fact, no less dogmatic than the greedy variant since we enforce orthogonality of the descent direction with the active columns of A. It can be modified to include recourse by simply not enforcing orthogonality of the descent direction with the active constraints. The analysis of this implementation, however, is beyond the scope of this work. ■

1.4.1 Greedy Algorithms

As promised above, we now show that the greedy rescaling of Algorithm 1.3.4 specified in Corollary 1.4.6, is equivalent to a well-known *greedy algorithm* (see [4] and references therein). The prototype greedy algorithm is formulated in [4] as follows:

Algorithm 1.4.8 (Orthogonal Matching Pursuit) Input the matrix A, the vector b and a solution tolerance $\tau > 0$.

Initialization: Let $\nu = 0$, $y^0 = 0$, $r^0 = b$, and the support set $\mathbb{J}^0 = \emptyset$.
Main iteration: For a given tolerance $\tau > 0$ do

- (Sweep.) For $j = 1,2,\ldots,n$ compute the errors $\iota(j) = \min_{z_j} \|a_j z_j - r^{\nu-1}\|^2$ where a_j denotes the jth column of A.

- (Update support.) Compute $J^\nu \equiv \operatorname{argmin}\{\iota(j) \mid j \notin \mathbb{J}^{\nu-1}\}$ and update $\mathbb{J}^\nu \equiv \mathbb{J}^{\nu-1} \cup \{J^\nu\}$.
- (Compute provisional solution and residual.) Compute

$$x^\nu \equiv \operatorname{argmin}\{\|Ax-b\|^2 \mid \operatorname{support}(x) = \mathbb{J}^\nu\} \quad \text{and} \quad r^\nu \equiv b - Ax^\nu. \tag{1.41}$$

- (Increment or stop.) If $\|r^\nu\| < \tau$, stop; otherwise set $\nu = \nu + 1$ and repeat.

Note that the calculation of the provisional solution (1.41) is almost the same as the calculation of the normal cone projection in Lemma 1.3.3, the only difference being that x^ν in (1.41) is the projection onto the *subspace* corresponding to the index set $\mathbb{J}^\nu$ while the subgradient w^ν in Lemma 1.3.3 is the projection onto the associated *normal cone* mapping.

Lemma 1.4.9 (provisional solution/subgradient equivalence) *Let $\bar{\mathbb{J}} \subset \mathbb{J}$ where $\mathbb{J} \equiv \{j \mid x^*_j \neq 0\}$ for x^* a solution to* (1.1) *with the counting objective $\varphi(x) = \|x\|_0$. Let $L = (L_1, L_2, \ldots, L_n)$ and choose any $\bar{y} \in \mathbb{R}^m$ such that $|a_j^T \bar{y}| \leq L_j$ with equality holding only for $j \in \bar{\mathbb{J}}$, and such that $\bar{w}_j \neq 0$ for any $j \in \bar{\mathbb{J}}$ where $\bar{w} = \operatorname{argmin}\{\|Aw - b\|_2^2 \mid w \in N_{R_L}(A^T\bar{y})\}$. Then $\bar{w} = \bar{x} \equiv \operatorname{argmin}\{\|Ax-b\|^2 \mid x_j = 0 \quad \forall j \notin \bar{\mathbb{J}}\}$.*

Proof. If $\bar{w}_j \neq 0$ for all $j \in \bar{\mathbb{J}}$ then the minimizer of $\|Aw-b\|^2$ is in the relative interior to $N_{R_L}(A^T\bar{y})$, an orthant of the subspace containing the support of $\bar{x}$. Hence minimizers of $\|Aw-b\|^2$ over the orthant and the entire subspace are equivalent, that is $\bar{w} = \bar{x}$. ■

Less obvious is the fact that the active index selection in Algorithm 1.4.8 is equivalent to an exact line search with a dynamically reweighted ℓ_1 norm.

Lemma 1.4.10 (step length/active index selection) *Define $\bar{\mathbb{J}} \subset \{1, 2, \ldots, n\}$ and $\bar{L} = (\bar{L}_1, \bar{L}_2, \ldots, \bar{L}_n)$ with*

$$\bar{L}_j \equiv \begin{cases} \|a_j\| & \text{for } j \notin \bar{\mathbb{J}} \\ 0 & \text{for } j \in \bar{\mathbb{J}}. \end{cases}$$

and the sets $R_{\bar{L}}$ and $\bar{C} \equiv \{y \in \mathbb{R}^m \mid A^T y \in R_{\bar{L}}\}$ accordingly. Let $\bar{d} = b - P_{N_{\bar{C}}(0)} b$. Then

$$\bar{J} \equiv \operatorname*{argmin}_{j \notin \bar{\mathbb{J}}} \left\{ \min_{z_j} \|a_j z_j - \bar{d}\|^2 \right\} \tag{1.42}$$

is the index set corresponding to the step length $\bar{\lambda}$ given by (1.35)*, that is,*

$$\min_{k \notin \bar{\mathbb{J}}} \left\{ \frac{\|a_k\|}{|a_k^T \bar{d}|} \right\} = \frac{\|a_j\|}{|a_j^T \bar{d}|} \quad \forall j \in \bar{J}.$$

Proof. We work forward from the definition of $\bar{J}$. Substituting

$$\frac{a_j^T \bar{d}}{\|a_j\|^2} = \min_{z_j} \|a_j z_j - \bar{d}\|^2$$

into (1.42) yields

$$\begin{aligned}
\bar{J} &= \operatorname*{argmin}_{j \notin \bar{\mathbb{J}}} \left\{ \left\| \frac{a_j^T \bar{d}}{\|a_j\|^2} a_j - \bar{d} \right\|^2 \right\} \\
&= \operatorname*{argmin}_{j \notin \bar{\mathbb{J}}} \left\{ \frac{|a_j^T \bar{d}|^2}{\|a_j\|^2} \left(\frac{\|a_j\|^2 \|\bar{d}\|^2}{|a_j^T \bar{d}|^2} - 1 \right) \right\} \\
&= \operatorname*{argmin}_{j \notin \bar{\mathbb{J}}} \left\{ \|\bar{d}\|^2 - \frac{|a_j^T \bar{d}|^2}{\|a_j\|^2} \right\} \\
&= \operatorname*{argmax}_{j \notin \bar{\mathbb{J}}} \left\{ \frac{|a_j^T \bar{d}|^2}{\|a_j\|^2} \right\} \\
&= \operatorname*{argmin}_{j \notin \bar{\mathbb{J}}} \left\{ \frac{\|a_j\|}{|a_j^T \bar{d}|} \right\}.
\end{aligned}$$

This completes the proof. ■

We conclude that orthogonal matching pursuit is equivalent to the dynamically reweighted steepest subgradient descent method with exact line search.

Proposition 1.4.11 *Algorithm 1.4.8 is equivalent to Algorithm 1.3.4 initialized with* $y^0 = 0$ *and* $L^0 = (\|a_1\|, \|a_2\|, \ldots, \|a_n\|)$*, and with the rescaling* $\gamma^\nu = 0$ *for all* ν*.*

Proof. This follows immediately from Lemmas 1.4.9 and 1.4.10. ■

1.5 Numerical Examples

The equivalence of Algorithm 1.3.4 with $\gamma^\nu = 0$ for all ν to the orthogonal matching pursuit algorithm 1.4.8 makes the wealth of numerical experience with orthogonal matching pursuit immediately available to our more general algorithm. We only demonstrate in this section that the greedy version of the algorithm and the more general version behave similarly on sufficiently sparse problems.

Remark 1.5.1 Before presenting our numerical examples, a few comments about practical implementations are in order. As pointed out earlier, in the absence of exact arithmetic, practical implementations cannot directly apply the most general form of Algorithm 1.3.4. However, even without exact arithmetic, we can determine precisely the operative quantities as long as the numerical error is below the threshold needed to discriminate between certain discrete cases.

For example, suppose we have 14 digits of accuracy and $|a_j^T y^\nu|$ is to within 10^{-15} of L_j^ν: would it be equal to L_j^ν if we had exact arithmetic? If $L_j^\nu = 0$, then it must be that $a_j^T y^\nu = 0$ with exact arithmetic since it was proved in Propositions 1.3.5 and 1.3.7 that the iterates are generated from feasible directions with step length chosen so that the iterates are always feasible. If the dynamic reweighting were chosen so that $L_j^\nu > 0$, then it is impossible to determine whether $a_j^T y^\nu$ *should* equal, say, $-L_j^\nu$, unless it is known that $a_j^T d^\nu = 0$, in which case it *should* hold that $a_j^T y^{\nu-1} = a_j^T y^\nu$, where it has been determined from previous iterations that $a_j^T y^{\nu-1} = -L_j^{\nu-1}$. Again, by Proposition 1.3.5, if $w_j^\nu \neq 0$ for j in the active set $\mathbb{J}^\nu$ and

$$w^\nu = \operatorname{argmin} \{ \|Aw - b\|^2 \mid w \in N_{R_{L^\nu}}(A^T y^\nu) \}$$

then $a_j^T d^\nu = 0$. Let δ be the numerical accuracy of the computation. If $|w_j^\nu| > \delta$ then we are certain that $w_j^\nu \neq 0$, and thus $a_j^T d^\nu = 0$ so that $a_j^T y^\nu = a_j^T y^{\nu-1} = L_j^{\nu-1}$. If instead $|w_j^\nu| \leq \delta$, then we cannot be sure that $|w_j^\nu| \neq 0$ and consequently we cannot be certain that d^ν is orthogonal to the active columns of A.

This numerical uncertainty is related to the *ill-posedness* of the problem $Ax = b$: if the sparsest signal x^* has elements whose magnitude is below the numerical noise level, then the algorithm must be regularized. We will have more to say about this in the conclusion. For our numerical study we only take examples for which the signal is above the numerical noise level, and so our exact arithmetic algorithm is still implementable. ■

We turn to our numerical illustration:

Our "Toy" problem. For our numerical example, we construct a real signal of length 128^2 ($n = 2 \times 16,384$ to account for real and imaginary parts) with 70 nonzero components ($|\mathbb{J}^*| = 70$), chosen at random, and randomly sample the discrete Fourier transform of this signal at a rate of about $1/8$. Since the true signal is real-valued, our effective sampling rate is about $1/4$ due to symmetry in the Fourier coefficients ($m = 2 \times 3588$ for the real and imaginary parts). Since we are dealing with the Fourier transform, the scaling of columns of

$$A \in \mathbb{R}^{(2*16384)\times(2*3588)}$$

is just $\|a_j\| = 1/\sqrt{2*3588}$.

Algorithm illustrations. We illustrate the theory with two different implementations of Algorithm 1.3.4, the first with scaling parameter $\gamma^\nu > 0$ for each iteration ν (in fact, we need only take $\gamma^\nu = 1$ to satisfy the requirements of the algorithm) and the second with $\gamma^\nu = 0$ for all iterations corresponding to the "greedy" implementation. The complexity of the two implementations is identical. Both instances converge in 70 iterations and require the same work to compute the subgradient.

Although the normal equations provide an explicit closed-form expression for the calculation of the subgradient w in (1.18), this still involves the inversion of a matrix, albeit small relative to the overall problem size. As we are interested in applications for which the sparsity is on the order of 10^3 to 10^4 nonzero elements, in-

stead, we solve (1.18) iteratively using the Relaxed Average Alternating Reflection (RAAR) algorithm [10, 11] for finding *best approximation pairs* between the sets $N_{R_L}(A^T y'')$ and $B \equiv \{x \mid Ax = b\}$. (It is important to note that we can only find best approximation pairs since for all but the last iteration $N_{R_L}(A^T y'') \cap B = \emptyset$.) Ordinary alternating projections would have also sufficed to solve this subproblem, however we found that the RAAR algorithm required, on average, 33% fewer iterations with the proper choice of relaxation parameter.

Both of our implementations of Algorithm 1.3.4 require exactly the same number of iterations of the RAAR algorithm to compute (1.18) since they both solve the exact same subproblem at each iteration. The subproblems require, on average, 82.6 iterations to get to within the numerical tolerance (10^{-12}).

Complexity. Rather than explicitly forming the partial Fourier matrix A we take advantage of the fast Fourier transform. The FFT is the most complex computation in the algorithm. The RAAR algorithm requires 2 FFT computations per iteration on a complex-valued vector of length 128^2 and the main loop of Algorithm 1.3.4 requires 3 FFT computations of the same complexity. For the example reported here, over all the iterations, the algorithm required in total $821,871$ FFT computations on complex-valued vectors of length 128^2, or on the order of 10^{11} floating point operations. On a 2.2 GHz Intel Core 2 Duo processor with 2GB 667 MHz memory this takes 32 seconds of CPU time.

If instead of using the FFT we had used the normal equations to explicitly compute the subgradients we would have needed only 211 FFT computations, and the matrix inversions required in the normal equations would have required, at the worst, inversion of a 70×70 real-valued matrix. The computational complexity of this approach is estimated to be on the order of 10^7 floating point operations. For problems with sparsity < 700 elements the normal equation approach will probably be faster; thereafter iterative methods, such as RAAR, using the FFT become competitive.

Figure 1.1(a) shows the error between the reconstructed signal and the true signal. The reconstruction for both implementations are identical. Figure 1.1(b) shows the weights corresponding to the implementation with scaling $\gamma^\nu = 0$ for all ν. The weights for the implementation with $\gamma^\nu = 1$ for all ν are not shown since these are all identical and unchanged from the initialization. Note that $\gamma^\nu = 1$ for all ν is then the behavior of the algorithm for solving the fixed, reweighted ℓ_1 optimization problem *for this problem*. These will not, in general be the scalings chosen by the algorithm on different problems. Finally, in Figure 1.1(c) we give a comparison of the step lengths at each iteration of the two implementations.

1.6 Comments and Conclusion

Our goals herein were to apply convex analysis to the nonconvex problem of sparse signal recovery and to take notions that have evolved from different approaches and incorporate these advances into convex analysis and algorithms. With this work we have made a first step in this direction.

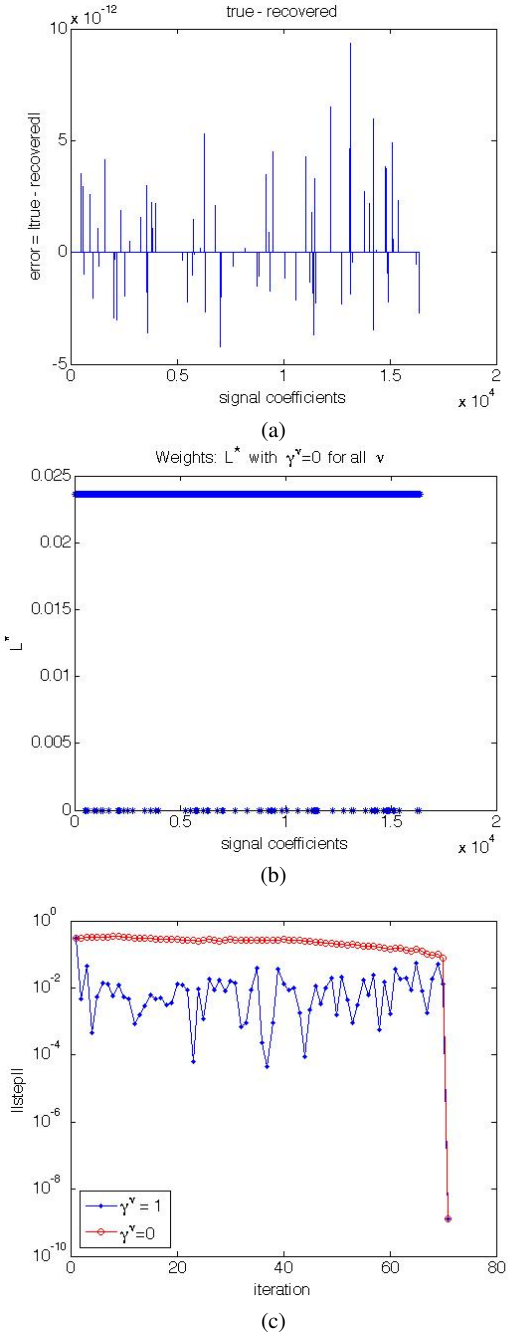

Fig. 1.1 (a) Pointwise reconstruction error. (b) Weights at the optimal solution for the implementation with $\gamma^\nu = 0$ for all ν. (c) Comparison of magnitude of steps between $\gamma^\nu = 0$ and $\gamma^\nu = 1$ implementations at each iteration

We proposed convex dual-space relaxations of the original nonconvex problem and have analyzed one extreme of possible relaxations. We have proved convergence in finitely many steps of a nonsmooth steepest descent method with exact line search and dynamically reweighted ℓ_1 norms when applied to problems satisfying the mutual coherence condition.

An instance of our algorithm is shown to be equivalent to orthogonal matching pursuit, which has been well-studied in the literature, though we are unaware of any identification of this method to dual-space linesearch methods as presented here. This explicit connection of orthogonal matching pursuit to reweighted ℓ_1 minimization in the dual opens the door to a greater synthesis of algorithms and a better understanding of the behavior of these algorithms.

Indeed, the proof of the coincidence of the solution to the ℓ_1 minimization problem to the solution of the corresponding minimization of the counting metric $\|\cdot\|_0$ is usually given indirectly. Here, under the assumption of mutual coherence and certain interiority qualifications on the projection of the data onto the normal cone associated with the active constraints, we have an explicit proof of the equivalence of the solutions to the ℓ_1 and $\|\cdot\|_0$ problems. An instance of this equivalence was demonstrated in the numerical example.

Our numerical examples do not extend to circumstances not covered by the theory developed here. There are two sources of failure of the algorithm, one due to the sparsity conditions not being met, and the other due to numerical error. We emphasized the importance of recognizing algorithms that implicitly rely on exact arithmetic and how implementations can succeed or fail without it. We are unaware of a numerical study that distinguishes between instances where the sparsity conditions are not met and instances where the numerical tolerance is not precise enough for a practical implementation. This is a topic worthy of greater attention than we have space for here.

The next step in this research will be to investigate the other relaxations, $\varepsilon > 0$ of (1.8). For this instance the objective is smooth (infinitely differentiable) in its domain R_L, and the gradient can be written in closed-form. We conjecture that the corresponding steepest descent, exact linesearch algorithm with dynamic reweighting will behave much like an interior point algorithm since the effect of the parameter ε is to keep the iterates on the interior of the feasible region.

Another direction that needs to be addressed is sparse *approximate* solutions to the model $Ax = b$. This is more appropriate for applications where the image b is corrupted by noise, or, as we have seen, numerical error. There has been a lot of very good work in this direction by other researchers. Our approach is appropriate for fast (finitely terminating), highly accurate exact solutions. It remains to be seen whether this basic program extends to fast (polynomial time), reasonably accurate approximate solutions.

References

1. Borwein, J., Vanderwerff, J.: Convex Functions: Constructions, Characterizations and Counterexamples, *Encyclopedias in Mathematics*, vol. 109. Cambridge University Press, New York (2010)
2. Borwein, J.M., Lewis, A.S.: Convex analysis and nonlinear optimization : theory and examples, 2nd edn. Springer Verlag, New York (2006)
3. Borwein, J.M., Luke, D.R.: Duality and Convex Programming. In: O. Scherzer (ed.) Handbook of Mathematical Methods in Imaging. Springer (to appear)
4. Bruckstein, A.M., Donoho, D.L., Elad, M.: From sparse solutions of systems of equations to sparse modeling of signals and images. SIAM Rev. **51**(1), 34–81 (2009)
5. Candès, E., Recht, B.: Exact matrix completion via comvex optimization. Found. of Comput. Math. **9**, 717–772 (2009)
6. Candes, E., Tao, T.: Near-optimal signal recovery from random projections: Universal encoding strategies? IEEE Trans. Inform. Theory **52**(12), 5406–5425 (2006)
7. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20**, 33–61 (1998)
8. Donoho, D.L., Elad, M.: Optimally sparse representation in general (non-orthogonal) dictionaries via l1 minimization,. Proc. Natl. Acad. Sci. **100**, 2197–2202 (2003)
9. Gribonval, R., Nielsen, M.: Sparse decompositions in unions of bases. IEEETrans. Inform. Theory **49**, 3320–3325 (2003)
10. Luke, D.R.: Relaxed averaged alternating reflections for diffraction imaging. Inverse Problems **21**, 37–50 (2005)
11. Luke, D.R.: Finding best approximation pairs relative to a convex and a prox-regular set in Hilbert space. SIAM J. Optim. **19**, 714–739 (2008)
12. Natarajan, B.K.: Sparse approximate solutions to linear systems. SIAM J. Comput. **24**, 227–234 (1995)
13. Recht, B., Fazel, M., Parrilo, P.: Guaranteed minimum rank solutions of matrix equations via nuclear norm minimization. SIAM Review (to appear)
14. Rockafellar, R.T., Wets, R.J.: Variational Analysis. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, Berlin (1998)
15. Santosa, F., Symes, W.W.: Linear inversion of band-limited reflection seismograms. SIAM J. Sci. Statist Comput. **7**, 1307–1330 (1986)