A SEMISMOOTH NEWTON METHOD WITH MULTI-DIMENSIONAL FILTER GLOBALIZATION FOR $L_1$-OPTIMIZATION

ANDRE MILZAREK∗ AND MICHAEL ULBRICH†

Abstract. Due to their property of enhancing the sparsity of solutions, $l_1$-regularized optimization problems have developed into a highly dynamic research area with a wide range of applications. We present a class of methods for $l_1$-regularized optimization problems that are based on a combination of semismooth Newton steps, a filter globalization, and shrinkage/thresholding steps. A multidimensional filter framework is used to control the acceptance and to evaluate the quality of the semismooth Newton steps. If the current Newton iterate is rejected, a shrinkage/thresholding-based step with quasi-Armijo stepsize rule is used instead. Global convergence and transition to local q-superlinear convergence for both convex and nonconvex objective functions are established. We present numerical results and comparisons with several state-of-the-art methods that show the efficiency and competitiveness of the proposed method.

Key words. $l_1$ optimization, fixed point method, semismooth Newton method, multi-dimensional filter, global convergence, superlinear convergence

AMS subject classifications.

1. Introduction. In this paper, we propose and investigate a globally and locally superlinearly convergent algorithm for solving $l_1$-regularized optimization problems of the form

$$\min_x f(x) + \mu \|x\|_1,$$

(1.1)

where $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable, possibly nonconvex, $\mu > 0$ is a parameter, and $\|x\|_1 = |x_1| + \ldots + |x_n|$. The proposed method combines semismooth Newton steps, a multidimensional filter globalization, and shrinkage steps. The presented convergence theory covers both the convex and the nonconvex case. Recent years showed a significant interest in $l_1$-regularized optimization problems of the form (1.1), which is particularly due to the sparsity enhancing properties of the $l_1$-regularization. The remarkably universal role of sparse solutions stems from the fact that in many applications, such as signal or image processing, there exist canonical sparse representations of the relevant data. A strong focus has been so far on the case where $f$ is smooth and convex [3, 11, 28, 30, 49]. Even more specifically, and motivated by compressive sensing, the majority of research has been concentrated on the convex quadratic case $f(x) = \frac{1}{2} \|Ax - b\|^2$, [6, 14, 20, 27, 29, 46], with $\|\cdot\| = \|\cdot\|_2$, which is closely related to the basis pursuit denoising problem

$$\min_{x \in \mathbb{R}^n} \|x\|_1 \text{ s.t. } \|Ax - b\| \leq \sigma.$$

(1.2)

In fact, it is easy to show that any solution $x^*$ of (1.1) with $f(x) := \frac{1}{2} \|Ax - b\|^2$ solves also the basis pursuit denoising problem for $\sigma = \|Ax^* - b\|$, which supports the high relevance of the problem class (1.1). For $\sigma = 0$, the problem (1.2) becomes the basis pursuit problem and the constraint then is written as $Ax = b$. Basis pursuit can be

∗Technische Universität München, Department of Mathematics, Boltzmannstr. 3, D-85747 Garching b. München, Germany (milzarek@ma.tum.de)
†Technische Universität München, Chair of Mathematical Optimization, Department of Mathematics, Boltzmannstr. 3, D-85747 Garching b. München, Germany (mulbrich@ma.tum.de).
A. MILZAREK AND M. ULBRICH

interpreted as a convex relaxation of the (in general NP-hard) problem of finding the sparsest solution of the linear system $Ax = b$:

$$\min_{x \in \mathbb{R}^n} \|x\|_0 \quad \text{s.t.} \quad Ax = b. \quad (1.3)$$

Here, the $l_0$-quasi norm $\|x\|_0 := |\{i : x_i \neq 0\}|$ counts the number of nonzeros. Under suitable conditions on the matrix $A$ and on the sparsity of a solution $x^*$ of (1.3), $x^*$ can be shown to be the unique solution of (1.2). This together with the close relation between (1.1) and (1.2) gives one of several possible explanations why $l_1$-regularizations favor sparse solutions. A typical situation is that $x \mapsto a_i^T x$, where $a_i^T$ denotes the $i$-th row of $A$, corresponds to a linear measurement functional and $b_i$ is the corresponding data. If it is known that the signal $\bar{x}$ giving rise to the data $b$ is sparse, i.e., $p := \|\bar{x}\|_0 \ll n$, this knowledge can be explored to reconstruct $\bar{x}$ from far less than $n$ measurements. Details and further information can be found in the work of Candès, Romberg, Tao [7, 8, 9] and Donoho [15]. Compressive sensing significantly extended the class of existing data and signal acquisition methods and has been applied to a wide variety of fields, such as compressive imaging [17, 45], magnetic resonance and computed tomography imaging [1, 34, 35], seismics [32], or communication [2]. This and many further potential applications show that the sparsity enhancing property of the $l_1$ norm makes (1.1) an important class of optimization problems. Many of them are large scale [30, 41] and often the matrix $A$ or more generally the evaluation of $f$ and its derivatives contains (forward or inverse) discrete cosine, wavelet, or other transforms, such that only matrix-free operations like $Av$ or $A^T w$ are available [3, 6, 19, 20, 46].

For $l_1$-optimization problems, especially those centered around basis pursuit denoising, several different classes of algorithmic approaches have been developed over the past years.

Among many others, Orthogonal Matching Pursuit (OMP) [43], StOMP [16] Subspace Pursuit [13] and CoSaMP [37] are examples for greedy algorithms that try to reconstruct a sparse signal via subspace optimization techniques. In each iteration a least squares problem is solved on an appropriately fixed subspace to obtain an approximate solution. The subspaces are determined by a selection of certain components of the current approximation and are changed iteratively depending on the respective method.

A large class of first order $l_1$-minimization algorithms for problem (1.3) is based on the iterative shrinkage/thresholding (IST) method [14, 18, 19, 47] or on variants of IST that incorporate proximal thresholding [10] or forward-backward splitting schemes [11, 28]. Recently, many extensions and modifications of these base algorithms have been introduced and investigated. Examples of such variants include the two-staged TwIST [6], continuation methods such as FPC [28] and SpaRSA [47], the spectral gradient projection method GPSR [20] and the FPC-AS [46] algorithm that combines shrinkage, subspace optimization, and continuation techniques.

Further examples are NESTA [4] and SPGL1 [5], which both solve the basis pursuit denoising problem (1.2), the interior-point method $l_1J_a$ [29], FISTA [3], a block coordinate gradient descent method [49] and a primal-dual algorithm [12] that was studied by Chambolle and Pock and is designed for minimizing general, convex composite functions. Based on the alternating direction method (ADM) [24], Yang and Zhang proposed the method YALL1 [48], that provides an algorithmic framework for basis pursuit (denoising) and related problems. We refer to section 5.2 for a more detailed description of the algorithms FPC, FPC-AS, GPSR, NESTA, SpaRSA and
SPGL1.
In [27] Griesse and Lorenz considered a semismooth Newton method SSN for \( l_1 \)-regularized optimization in an infinite-dimensional setting and established local superlinear convergence. Stadler [42] considered semismooth Newton methods for elliptic optimal control problems with \( L^1 \)-control cost.

1.1. Overview of the method. We propose and analyze an algorithm that combines the efficiency of filter globalization techniques with the fast local convergence properties of the semismooth Newton method [38, 39]. Our approach is primarily based on the idea to obtain trial steps from semismooth Newton steps for a nonsmooth formulation

\[
F_\tau(x) = 0,
\]

of the first order optimality conditions. The acceptance of these steps is controlled by a multi-dimensional filter globalization technique. If the semismooth Newton step is not accepted, then a suitably chosen descent step is performed. The main requirement is that these alternative steps ensure global convergence in the case where only finitely many semismooth Newton steps are taken. In this paper, we choose soft thresholding shrinkage steps with an Armijo-type line search for this purpose. The piecewise continuously differentiable function \( F_\tau : \Omega \rightarrow \mathbb{R}^n \) arising in (1.4) will be derived in section 2. It is closely related to the fixed point equation that underlies the shrinkage iteration.

We use a globalization technique that is based on a multidimensional filter framework. Originally, the filter concept was developed by Fletcher and Leyffer [21] in order to globalize SQP methods for nonlinear programming problems without using penalty functions. The original version of the filter method works with a two dimensional filter, where each entry consists of the objective function value and a measure for the constraint violation at a given point. The filter globalization concept has rapidly established itself as one of the most important and efficient globalization techniques in nonlinear programming. For further details we refer to [22, 23, 44]. Gould, Leyffer and Toint modified this concept in [25] and proposed a multidimensional filter to globalize (Gauss-) Newton-based methods for nonlinear equations and least squares problems. In [26] Gould, Sainvitu and Toint adapted this approach to an unconstrained minimization problem by applying the method to the gradient of the objective function. Our method can be viewed as an extension of this idea to the nonsmooth setting of \( l_1 \) optimization.

Under assumptions comparable to those of other state-of-the-art methods, we prove for our algorithm that every accumulation point of the generated sequence is a stationary point. Furthermore, under suitable second order conditions, transition to q-superlinear local convergence is shown. In contrast to many other analyses, we consider not only the case of convex \( f \), but also address the general situation of a nonconvex function \( f \).

In the numerical implementation, following the strategy in [4, 28, 46], we accelerate our method by using a continuation scheme for the regularization parameter \( \mu \).

1.2. Organization and notation. By \( \| \cdot \| := \| \cdot \|_2 \) we denote the 2-norm and the gradient and Hessian of \( f \) at \( x \) are written as \( g(x) := \nabla f(x) \) and \( H(x) := \nabla^2 f(x) \), respectively. If \( V \in \mathbb{R}^{n \times n} \) is a symmetric matrix, then its maximum and minimum eigenvalues are denoted as \( \lambda_{\text{max}} \) and \( \lambda_{\text{min}} \). For a matrix \( M \in \mathbb{R}^{n \times n} \) and index sets
the matrix \( M_{I,J} := (M_{ij})_{i \in I, j \in J} \in \mathbb{R}^{|I| \times |J|} \) denotes the submatrix of \( M \) corresponding to the rows \( I \) and columns \( J \).

This paper is organized as follows. In section 2 we specify different optimality conditions for the \( l_1 \)-minimization problem and derive the nonsmooth equation (1.4). In section 3 we state the assumptions under which we prove convergence and discuss some preliminaries concerning the properties of the generalized fixed point method as well as the theoretic introduction and examination of the multidimensional filter method. We conclude this section with the presentation of the main approach. In section 4 we present and prove our results on global and local convergence of the algorithm. Concentrating on the sole convergence analysis we move some intermediate results which mainly consist of local properties to section 5. In section 6 we propose a continuation scheme for the main approach and compare the completed algorithm with other state-of-the-art methods.

2. Optimality Conditions. We now derive first order optimality conditions for the \( l_1 \)-problem

\[
\min_{x \in \mathbb{R}^n} f(x) + \mu \|x\|_1, \quad (2.1)
\]

where \( f : \mathbb{R}^n \to \mathbb{R} \) is a twice continuously differentiable function and \( \mu \geq 0 \) is a parameter. Denoting by \( \phi(x) := f(x) + \mu \|x\|_1 \) the objective function of (2.1), we call \( x \in \mathbb{R}^n \) a stationary point of \( \phi \) if it satisfies

\[
\phi'(x, d) \geq 0 \quad \forall d \in \mathbb{R}^n, \quad (2.2)
\]

where \( \phi'(x, d) := \lim_{t \to 0^+} \frac{\phi(x + td) - \phi(x)}{t} \) is the directional derivative of \( \phi \). By computing \( \phi'(x, d) \) explicitly, we obtain the following equivalent formulation of (2.2):

\[
 g_i(x) \begin{cases} 
 - \mu, & x_i > 0, \\
 \in [-\mu, \mu], & x_i = 0, \\
 + \mu, & x_i < 0, 
\end{cases} \quad i = 1, \ldots, n. \quad (2.3)
\]

It is straightforward to see that (2.3) is equivalent to the following nonsmooth equation:

\[
 F_\tau(x) := g(x) - P_{[-\mu, \mu]} \left( g(x) - \frac{x}{\tau} \right) = 0. \quad (2.4)
\]

Here, \( \tau > 0 \) is arbitrarily fixed, \( P_{[-\mu, \mu]} : \mathbb{R} \to [-\mu, \mu] \) denotes the projection onto the set \([-\mu, \mu]\), i.e., \( P_{[-\mu, \mu]}(t) := \min\{\max\{-\mu, t\}, \mu\} \), and is applied component-wise.

**Remark 2.1.** The function \( F_\tau \) is piecewise \( C^1 \) and thus semismooth on \( \mathbb{R}^n \setminus \{0\} \). If \( \nabla^2 f \) is locally Lipschitz continuous, then \( F_\tau \) is strongly semismooth.

Next, we introduce the two functions

\[
 G_\tau(z) := z - \tau g(z) \quad \text{and} \quad S_\nu(z) := z - P_{[-\nu, \nu]}(z).
\]

Here, \( G_\tau \) is a scaled gradient step for \( f \) with stepsize \( \tau > 0 \) and \( S_\nu \) is the shrinkage operator or soft-thresholding function, which preserves the sign of all components \( z_i \) and reduces \( |z_i| \) to \( \max\{0, |z_i| - \nu\} \). In particular, there holds

\[
 ||S_\nu(z)||_i \leq |z_i| - \min\{\nu, |z_i|\} \leq |z_i| \quad \forall \ i. \quad (2.5)
\]
It is elementary to verify the identity

\[ x - \tau F(x) = S_{\tau \mu}(G_{\tau}(x)). \tag{2.6} \]

Hence, we obtain that (2.4) is equivalent to the fixed point equation

\[ x = S_{\tau \mu}(G_{\tau}(x)). \tag{2.7} \]

We collect the derived three equivalent formulations of (2.2):

**Lemma 2.2.** Let \( \tau > 0 \) be arbitrarily fixed. Then \( x \) is a stationary point of \((2.1)\) if and only if it satisfies one (and thus all) of the mutually equivalent conditions \((2.2)\), \((2.3)\), \((2.4)\), or \((2.7)\).

The formulation \((2.7)\) induces the following fixed point iteration:

\[ x^{k+1} = S_{\tau \mu}(G_{\tau}(x^k)) \text{ with } \tau > 0. \tag{2.8} \]

The fixed point iteration \((2.8)\) and its associated algorithm were intensively discussed and used by various authors \([6, 14, 18, 19, 20, 28, 47]\). It is also known under the name of iterative shrinkage or thresholding method or forward-backward splitting approach \([3, 11, 14, 28, 47]\), and forms the basis of a general descent method that will be presented in section 3.1.

The shrinkage step \( S_{\tau \mu}(G_{\tau}(x)) \) can be interpreted as the solution of the following separable \(\ell_1\)-regularized quadratic problem:

\[ S_{\tau \mu}(G_{\tau}(x)) = \arg\min_{z \in \mathbb{R}^n} g(x)^T(z - x) + \frac{1}{2\tau} \|z - x\|_2^2 + \mu \|z\|_1. \]

In fact, setting \( f_x(z) = g(x)^T(z - x) + \frac{1}{2\tau} \|z - x\|_2^2 \), the problem \( \min_z f_x(z) + \mu \|z\|_1 \) is of the form \((2.1)\) with \( x \) and \( f(x) \) replaced by \( z \) and \( f_x(z) \). Hence, writing down the optimality condition \((2.7)\) for the solution \( z \) of this problem, we obtain:

\[ z = S_{\tau \mu}(z - \tau \nabla f_x(z)) = S_{\tau \mu}(z - \tau g(x) - (z - x)) = S_{\tau \mu}(x - \tau g(x)) = S_{\tau \mu}(G_{\tau}(x)). \]

### 3. A Globalized Semismooth Newton Method

The following conditions summarize our assumptions for proving that every accumulation point of the proposed algorithm is a stationary point. In contrast to most other convergence analyses for \(\ell_1\)-optimization algorithms, we consider both the general nonconvex case and the convex case. We first present the assumptions \((A1)-(A3)\) tailored to the convex case, then the requirements \((B1), (B2)\) for the nonconvex situation. Later, it will be shown that, for convex \( f \), \((A1)-(A3)\) imply \((B1), (B2)\) in the setting of our algorithm.

**Assumption.** There exists an open, convex set \( \Omega \subset \mathbb{R}^n \) such that

- (A1) The function \( f : \Omega \to \mathbb{R} \) is convex and twice continuously differentiable.
- (A2) The level sets \( N(\gamma) := \{ x \in \Omega : \phi(x) \leq \gamma \} \), \( \gamma \in \mathbb{R} \) are closed in \( \mathbb{R}^n \).
- (A3) The solution set \( X^* \) is nonempty and compact.

Alternatively if we do not assume \( f \) to be convex, we demand the following:

**Assumption.** There exists an open convex set \( \Omega \subset \mathbb{R}^n \) such that

- (B1) The function \( f : \Omega \to \mathbb{R} \) is twice continuously differentiable on \( \Omega \).
- (B2) The sequence of iterates \((x^k)\) stays in a compact set \( \Omega_0 \subset \Omega \).

Besides those two blocks of conditions we also need:
Assumption.

(C) The Hessian of the function $f$ is uniformly bounded on $\Omega$, i.e.,

$$\lambda_{\max} := \sup_{x \in \Omega} \|H(x)\| < \infty.$$  

We have introduced the set $\Omega$ to allow for functions $f$ that are defined only on a subset of $\mathbb{R}^n$. The assumptions (A2) and (B2), respectively, ensure that the algorithm will not interfere with the boundary of $\Omega$. Hence, in the following, we target on finding stationary points of $\phi$ on the open convex set $\Omega$, i.e., of the problem:

$$\min_x \phi(x) := f(x) + \mu \|x\|_1 \quad (x \in \Omega). \quad \text{(3.1)}$$

We next show that for a general class of algorithms comprising those investigated in this paper, the assumptions (A1)–(A3) imply the conditions (B1) and (B2).

Lemma 3.1. Let $\Omega \subset \mathbb{R}^n$ and $f : \Omega \to \mathbb{R}$ satisfy the conditions (A1)–(A3) and consider a sequence $(x^k) \subset \Omega$. Let $K_S \cup K_F$ be a disjoint partitioning of $\mathbb{N}$ such that $\phi(x^k) \leq \phi(x^{k-1})$ for all $k \in K_S$ and either $K_F$ is finite (or empty) or $F_\tau(x^k) \to 0$ for $K_F \ni k \to \infty$. Then the following holds:

(i) All level sets $N(\gamma)$, $\gamma \in \mathbb{R}$, are compact.

(ii) The sequence of iterates $(x^k)$ remains in a compact set $\Omega_0 \subset \Omega$.

Proof. Due to the compactness of $X^*$ and the theorem of Weierstrass, we can expand $X^*$ by a small strip without leaving $\Omega$, i.e., there exists $\delta > 0$ such that

$$X^*_\delta := \{x + d : x \in X^*, \ d \in \mathbb{R}^n, \ \|d\| < \delta\}$$

is still contained in $\Omega$. Let $\Gamma := \partial X^*_\delta$ denote the boundary of $X^*_\delta$ and fix an arbitrary $x^* \in X^*$. Using the compactness of $\Gamma$, we can find $\epsilon > 0$ such that

$$\phi(x) > \phi(x^*) + \epsilon \quad \forall x \in \Gamma.$$

Further, there exists $r > 0$ with $\max_{x \in \Gamma} \|x - x^*\|_1 = r$. By combining these properties with the convexity of $f$ we can establish the following growth rate:

$$\phi(y) > \phi(x^*) + \frac{\epsilon}{r} \|y - x^*\|_1 \quad \forall y \in \Omega \setminus \overline{X^*_\delta}. \quad \text{(3.2)}$$

This implies the boundedness of all level sets $N(\gamma)$ and thus their compactness since $N(\gamma)$ is by assumption closed in $\mathbb{R}^n$. Hence, it suffices to show that all iterates are contained in an appropriate level set $N(\gamma)$. By (3.2), there holds

$$f(x) - f(x^*) \geq \mu(\vartheta - 1) \|x - x^*\|_1 \quad \text{(3.3)}$$

with a suitable constant $\vartheta > 0$ and for all $x \in \Omega$, $\|x - x^*\|_1 > r$. If $K_F$ is a finite set, then, due to $\phi(x^k) \leq \phi(x^{k-1})$ for all $k \in K_S$, we obtain $x^k \in N(\gamma)$ for all $k \geq 0$, where $\gamma := \max \{\phi(x^r) ; r \in \{0\} \cup K_F\}$.

Next, to prove that $(x^k)_{K_F}$ is bounded in the case $|K_F| = \infty$, assume in contrary that there exists a subsequence $(x^k)_K$ of $(x^k)_{K_F}$ with $\|x^k\| \to \infty$ as $K \ni k \to \infty$. Then $(F_\tau(x^k))_K \to 0$ and there exists a subsequence $(x^k)'_{K'}$ of $(x^k)_K$ with

$$g(x^k) = F_\tau(x^k) + \mathcal{P}_{[-\mu, \mu]} \left( g(x^k) - \frac{x^k}{\tau} \right) \to g^*$$

and $v^k := \frac{x^k - x^*}{\|x^k - x^*\|_1} \to v^*$.
as \( K' \ni k \to \infty \). This leads to the following contradiction

\[
(\vartheta - 1)\mu \leq \frac{f(x^k) - f(x^*)}{\|x^k - x^*\|_1} \leq \sum_{v^*_i < 0} g_i(x^k)v^*_i + \sum_{v^*_i = 0} g_i(x^k)v^*_i + \sum_{v^*_i > 0} g_i(x^k)v^*_i \\
\to \sum_{v^*_i < 0} \mu v^*_i - \sum_{v^*_i > 0} \mu v^*_i = -\mu,
\]

where we used (3.3) and the convexity of \( f \). Hence, the sequence \((x^k)_{K_F}\) is bounded. Further, there holds

\[
\|g(x^k)\| = \|F_\tau(x^k) + P_{[-\mu,\mu]} (g(x^k) - \tau^{-1}x^*)\| \leq \|F_\tau(x^k)\| + \mu\sqrt{n}.
\]

Using the gradient inequality for the convex function \( f \), we obtain

\[
\phi(x^k) - \phi(x^*) \leq g(x^k)^T(x^k - x^*) + \mu(\|x^k\|_1 - \|x^*\|_1) \\
\leq \|g(x^k)\|\|x^k - x^*\| + \mu\|x^k - x^*\|_1 \leq (\|F_\tau(x^k)\| + 2\mu\sqrt{n})\|x^k - x^*\|.
\]

Since the sequence \((x^k)_{K_F}\) is bounded, we see that \( \gamma := \max_{k \in \{0\} \cup K_F} \phi(x^k) < \infty \). From \( \phi(x^k) \leq \phi(x^{k-1}) \) for all \( k \in K_F \) we thus conclude that \( x^k \in N(\gamma) \) for all \( k \geq 0 \). \( \square \)

### 3.1. Fixed Point Method with Quasi-Armijo Rule

We now consider a globalized fixed point method that we use for the step generation whenever the semismooth Newton step is not accepted. It is advantageous to analyze this method separately before proceeding the development and investigation of the final overall algorithm.

Let \( x^k \) denote the current iterate and let \( d^k \) be a direction that is generated by the fixed point method (2.8). Then the globalized fixed point method calculates \( x^{k+1} = x^k + \sigma_d d^k \), where the step size \( \sigma_d \) is controlled by a quasi-Armijo rule. The details are formulated in Algorithm 1. This approach is closely related to a method proposed by Wen, Yin, Goldfarb, and Zhang in \[46\]. Instead of adopting their nonmonotone step size rule, we work with a monotone quasi-Armijo rule and give a compact convergence analysis in the general nonconvex setting. We use the following notation: \( g^k := \nabla f(x^k) \), \( \Delta^k := (g^k)^T d^k + \mu(\|S_{x^k}(G_\tau(x^k))\|_1 - \|x^k\|_1) \).

**Algorithm 1**: Fixed Point Method with Quasi-Armijo Rule

1. **Initialization**: Choose \( x^0 \in \Omega \subset \mathbb{R}^n \), \( \tau > 0 \), \( \beta, \gamma \in (0, 1) \).
2. **Calculate the direction**: \( d^k = S_{\tau\mu}(G_\tau(x^k)) - x^k \).
3. **If** \( d^k = 0 \), **then STOP**.
4. **Choose a maximal quasi-Armijo stepsize** \( \sigma_d \in \{1, \beta, \beta^2, \ldots\} \) with \( x^k + \sigma_d d^k \in \Omega \) and \( \phi(x^k + \sigma_d d^k) - \phi(x^k) \leq \sigma_d \gamma \Delta^k \).
5. **Set** \( x^{k+1} = x^k + \sigma_d d^k \).

In the following, we prove that Algorithm 1 is a globally convergent descent method. Further properties of Algorithm 1 will be discussed later together with the convergence analysis of our main approach.

**Lemma 3.2.** Let the assumptions (A1) or (B1) hold and let the sequences \((x^k)\) and \((d^k)\) be generated by Algorithm 1. Then for all \( k \geq 0 \) there holds

\[
\phi'(x^k, d^k) \leq \Delta^k \leq -\frac{1}{\tau} \|d^k\|^2.
\]
Proof. The first inequality follows from the convexity of the $l_1$-norm. For the second inequality, we use the characterization
\[
S_{\tau \mu}(G_{\tau}(x^k)) = \arg \min_{z \in \mathbb{R}} g(x^k)^T (z - x^k) + \frac{1}{2\tau} \|z - x^k\|^2 + \mu \|z\|_1.
\]
Setting $q_k(z) := g(x^k)^T (z - x^k) + \frac{1}{2\tau} \|z - x^k\|^2$, the point $z^k = S_{\tau \mu}(G_{\tau}(x^k))$ minimizes $q_k(z) + \mu \|z\|_1$ and thus its directional derivative is nonnegative in every direction $d$. Due to the convexity of $\| \cdot \|_1$, this directional derivative is a lower bound for $\nabla q_k(x_k)^T d + \mu(\|z_k + d\|_1 - \|z_k\|_1)$. For the direction $d = -d^k = x^k - z^k$, we thus obtain:
\[
0 \leq -\nabla q_k(z^k)^T d^k + \mu(\|z^k - d^k\|_1 - \|z^k\|_1)
= -g(x^k)^T d^k - \frac{1}{\tau} \|d^k\|^2 + \mu(\|x^k\|_1 - \|z^k\|_1) = -\Delta_k - \frac{1}{\tau} \|d^k\|^2.
\]
\[\Box\]

Remark 3.3. The quasi-Armijo condition in step (3) of Algorithm 1 and Lemma 3.2 directly imply $\phi(x^k) \leq \phi(x^{k-1})$ for all $k \in \mathbb{N}$. Thus, under the assumptions (A1)–(A3), Lemma 3.1 yields that the sequence $(x^k)$ stays in a compact set $\Omega_0 \subset \Omega$.

We next prove the boundedness of the directions $d^k$.

Lemma 3.4. Let the assumptions (A1)–(A3) or the conditions (B1), (B2) be satisfied and let $(d^k)$ be generated by Algorithm 1. Then there exist $C_x > 0$ and $C_d > 0$ with $\|x^k\| \leq C_x$ and $\|d^k\| \leq C_d$ for all $k \geq 0$.

Proof. The existence of a compact set $\Omega_0 \subset \Omega$ such that $x^k \in \Omega_0$ for all $k \geq 0$ follows either from (B2) or from Remark 3.3 and (A1)–(A3). Hence, there exists $C_x > 0$ with $\|x^k\| \leq C_x$ for all $k$. Since $g$ is continuous, there exists $g_{\text{max}} > 0$ with $\|g(x)\| \leq g_{\text{max}}$ for all $x \in \Omega_0$. We obtain

\[
\|d^k\| = \|S_{\tau \mu}(G_{\tau}(x^k)) - x^k\| \leq \|S_{\tau \mu}(x^k - \tau g^k)\| + \|x^k\| \leq \|x^k - \tau g^k\| + C_x \leq \tau g_{\text{max}} + 2C_x,
\]

where we used the component-wise monotonicity (2.5) of the shrinkage operator. \[\Box\]

Lemma 3.5. Let the assumptions (A1)–(A3) or (B1)–(B2) and the condition (C) hold and let the sequences $(x^k)$ and $(d^k)$ be generated by Algorithm 1. Then there exists a constant $\zeta > 0$ for all $k \in \mathbb{N}$ with
\[
\phi(x^k + \sigma d^k) - \phi(x^k) \leq \sigma \gamma \Delta^k \quad \text{for all } \sigma \in [0, \zeta].
\]

Proof. Apparently $\sigma = 0$ fulfills (3.4). So, let us consider $\sigma > 0$ sufficiently small, then we obtain for arbitrary but fixed $k \in \mathbb{N}$
\[
\frac{\phi(x^k + \sigma d^k) - \phi(x^k)}{\sigma} - \gamma \Delta^k \leq \frac{f(x^k + \sigma d^k) - f(x^k)}{\sigma} - g(x^k)^T d^k + (1 - \gamma) \Delta^k
\leq \frac{1}{2} \sigma \|d^k\|^2 H(x^k + \xi \sigma d^k) d^k - \frac{(1 - \gamma)}{\tau} \|d^k\|^2,
\]
where we used the convexity of the $l_1$-norm, Lemma 3.2 and a Taylor expansion with suitable $\xi \in [0, 1]$. In order to apply assumption (C), we have to ensure $x^k + \xi \sigma d^k \in \Omega$. As in the proof of Lemma 3.1 the compact set $\Omega_0$ can be expanded by a small $\epsilon_0$-strip without leaving $\Omega$:
\[
\{x + d : x \in \Omega_0, \ d \in \mathbb{R}^n, \ |d| \leq \epsilon_0\} \subset \Omega.
\]
Since \( \|d_k\| \leq C_d \) for all \( k \) by Lemma 3.4, we have \( x_k + \sigma d_k \in \Omega \) for all \( \sigma \in (0, \epsilon_0/C_d) \) and all \( k \geq 0 \). Using assumption (C) we now obtain, for all \( \sigma \leq \epsilon_0/C_d \),
\[
\frac{\phi(x_k + \sigma d_k) - \phi(x_k)}{\sigma} - \gamma \Delta_k \leq \left( \frac{1}{2} \sigma \hat{\lambda}_{\max} - \frac{(1 - \gamma)}{\tau} \right) \|d_k\|^2.
\]
Hence, the quasi-Armijo condition holds whenever
\[
\sigma \leq \zeta := \min \left\{ \frac{2(1 - \gamma)}{\tau \hat{\lambda}_{\max}}, \epsilon_0 \right\}.
\]

**Remark 3.6.** Lemma 3.5 shows that every step size sequence \((\sigma_k)\) generated by Algorithm 1, is uniformly bounded from below (whenever our assumptions hold). To be more precise, we have
\[
\sigma_k \geq \beta \zeta > 0 \quad \text{for all} \quad k \geq 0.
\]

**Theorem 3.7.** Let the assumptions (A1)–(A3) or (B1)–(B2) and condition (C) hold and let the sequence \((x^k)\) be generated by Algorithm 1. Then there holds \( F_{\tau}(x_k) \to 0 \) as \( k \to \infty \). In particular, every accumulation point \( x^* \) of \((x^k)\) satisfies \( F_{\tau}(x^*) = 0 \) and is thus a stationary point.

**Proof.** Since the sequence \((x^k)\) is bounded by Lemma 3.4, we see from Lemma 3.2 that \( \phi(x^k) \) converges monotonically decreasing to a limit \( \phi^* \in \mathbb{R} \). Now, Lemma 3.2 and Lemma 3.5 yield
\[
\phi(x^0) - \phi^* = \sum_{k=0}^{\infty} |\phi(x^k) - \phi(x^{k+1})| \geq \sum_{k=0}^{\infty} -\gamma \sigma_k \Delta_k \geq \sum_{k=0}^{\infty} \gamma \sigma_k \|d_k\|^2 \geq \gamma \beta \zeta \sum_{k=0}^{\infty} \|d_k\|^2.
\]
This shows \( d_k \to 0 \) as \( k \to \infty \). Using (2.6), there holds
\[
d_k = S_{\tau \mu}(G_{\tau}(x_k)) - x_k = -\tau F_{\tau}(x_k),
\]
which implies \( F_{\tau}(x_k) \to 0 \). Since \( F_{\tau} \) is continuous, we obtain \( F_{\tau}(x^*) = 0 \) for every accumulation point \( x^* \) of \((x^k)\). Hence, every accumulation point is a stationary point by Lemma 2.2.

### 3.2. A Multidimensional Filter Framework.
We adopt the multidimensional filter globalization concept of [25, 26] and tailor it to a semismooth Newton method for solving the equation
\[
F_{\tau}(x) = 0.
\]

The filter value corresponding to a point \( x \in \mathbb{R}^n \) is given by \( \theta(x) \), where the filter function \( \theta : \mathbb{R}^n \to \mathbb{R}^p_+ \) is continuous and satisfies
\[
c_0 \|F_{\tau}(x)\|_\infty \leq \|\theta(x)\|_\infty \leq C_0 \|F_{\tau}(x)\|_\infty
\]
with constants \( 0 < c_0 < C_0 \). This ensures that \((\theta(x^k))\) is bounded whenever \((F_{\tau}(x^k))\) is bounded and that every zero of \( \theta \) is also a zero of \( F_{\tau} \) and hence a stationary point of the \( l_1 \)-minimization problem.
At iteration \( k \), the filter \( F_k \subset \mathbb{R}_+^p \) is a finite collection of filter entries \( q \in \mathbb{R}_+^p \), where usually (and in our context always) each \( q \in F_k \) corresponds to a point \( x \in \mathbb{R}^n \), via \( q = \theta(x) \) and the points \( x \) are selected iterates \( x^l, l < k \), of the method to be globalized, in our case the points generated by semismooth Newton steps for (3.6).

A standard approach for choosing \( \theta \) (with many possible variants) is to decompose \( \{1, \ldots, n\} \) into \( p \) possibly overlapping nonempty sets \( I_j \) with \( \bigcup_{j=1}^p I_j = \{1, 2, \ldots, n\} \).

The function \( q \) holds for all \( j \in \{1, \ldots, p\} \).

Given the filter \( F \), the function \( \theta \) is then defined as

\[
\theta_j(x) := \frac{1}{\sqrt{|I_j|}} \| F_{\tau,I_j}(x) \| = \left( \frac{1}{|I_j|} \sum_{i \in I_j} F_{\tau,i}(x)^2 \right)^{1/2} \quad \forall \ j \in \{1, \ldots, p\}.
\]

This choice satisfies condition (3.7) with \( c_0 := 1/\sqrt{\max_j |I_j|} \) and \( C_\theta := 1 \). In the case \( p = 1 \) this yields \( \theta(x) = \frac{1}{\sqrt{n}} \| F_\tau(x) \| \) and in the case \( p = n \), \( I_j = \{j\} \), we obtain

\[
\theta(x) = (|F_{\tau,1}(x)|, |F_{\tau,2}(x)|, \ldots, |F_{\tau,n}(x)|)^T.
\]

Given the filter \( F_k \), similar to [25], we define an acceptance criterion for a point \( x \).

**Definition 3.8.** A point \( x \in \mathbb{R}^n \) is acceptable to the filter \( F_k \subset \mathbb{R}_+^p \setminus \{0\} \) if

\[
\max_{1 \leq j \leq p} (q_j - \theta_j(x)) \geq \gamma_F \delta(q, \theta(x))
\]

holds for all \( q \in F_k \). Here, \( \gamma_F \in (0, 1) \) is fixed and \( \delta : \mathbb{R}_+^p \times \mathbb{R}_+^p \to \mathbb{R}_+ \) is continuous and satisfies for all \( q \in \mathbb{R}_+^p \)

\[
\delta(q, q) = 0 \implies q = 0.
\]

If the new iterate \( x^{k+1} \in \mathbb{R}^n \) is acceptable to the current filter \( F_k \), we can, if we wish, update the filter by adding \( \theta(x^{k+1}) \) to the filter: \( F_{k+1} := F_k \cup \{\theta(x^{k+1})\} \). If the filter is not updated, then we set \( F_{k+1} := F_k \). After each update the filter can be scanned for redundant entries that no longer have influence on the acceptance rule and consequently can be removed. More details can be found in [25]. Returning to the acceptance rule, there are many suitable choices for the function \( \delta \). In this paper, we work with \( \delta(q, \theta(x)) := \|\theta(x)\|_\infty \). The corresponding acceptance test

\[
\max_{1 \leq j \leq p} (q_j - \theta_j(x)) \geq \gamma_F \|\theta(x)\|_\infty \quad \forall q \in F_k
\]

then ensures the uniform boundedness of the filter entries.

The filter ensures convergence in the following sense.

**Lemma 3.9.** Let \( \theta : \mathbb{R}^n \to \mathbb{R}_+^p \) be a filter function and let \( \delta : \mathbb{R}_+^p \times \mathbb{R}_+^p \to \mathbb{R}_+ \) satisfy condition (3.10). Furthermore, let \( (x^k)_K \) be an infinite subsequence of iterates such that \( (\theta(x^k))_{k \in K} \) is bounded, \( x^k \) is acceptable to \( F_{k-1} \) for all \( k \in K, k \geq 1 \), and the filter is updated, i.e., \( F_k = F_{k-1} \cup \{\theta(x^k)\} \), for all \( k \in K \). Then there holds

\[
\lim_{K \ni k \to \infty} \theta(x^k) = 0.
\]

**Proof.** Since the sequence \( (\theta(x^k))_{k \in K} \) is bounded, there exists a subsequence \( (\theta(x^{k_l}))_{l \in K_l} \), \( k_l \in K \), that converges to an accumulation point \( \theta^* \in \mathbb{R}_+^p \). Since \( \theta(x^{k_l}) \) is acceptable to \( F_{k_l-1} \supset F_{k_l-1} \supset \{\theta(x^{k_l-1})\} \), there holds

\[
\max_{1 \leq j \leq p} (\theta_j(x^{k_l-1}) - \theta_j(x^{k_l})) \geq \gamma_F \delta(\theta(x^{k_l-1}), \theta(x^{k_l})).
\]
Taking the limit $l \to \infty$, we obtain

$$0 = \max_{1 \leq j \leq p} (\theta_j^* - \theta_j^+) \geq \gamma_F \delta(\theta^*, \theta^*),$$

where we used the continuity of $\delta$. Applying the second part of condition (3.10), the last equation implies $\theta^* = 0$. \[\square\]

**Remark 3.10.** If we choose $\delta(q, \theta(x)) := \|\theta(x)\|_{\infty}$ as in (3.11), Lemma 3.9 holds without explicitly assuming the boundedness of the filter entries. In fact, since $x^k$ is acceptable to $F_{k-1}$ we then have, for all $q \in F_{k-1}$, that

$$\gamma_F \|\theta(x^k)\|_{\infty} \leq \max_{1 \leq j \leq p} (q_j - \theta_j(x^k)) \leq \|q\|_{\infty}.$$

Lemma 3.9 can be regarded as the essence of the multidimensional filter framework [25, 26]. The general idea to apply this concept is as follows: If a globally convergent base algorithm is given (in our case the fixed point method with quasi-Armijo stepsize rule) and an additional method for computing steps (in our case semismooth Newton) shall be incorporated, then we can use the filter to control acceptance of the latter steps while resorting to steps of the base algorithm, otherwise. Then, any subsequence of points generated by filter steps tends to stationarity. If only finitely many filter steps are taken, then global convergence follows from the properties of the base algorithm. This implies that there exists a subsequence approaching stationarity. To prove that every accumulation point is stationary, the tricky part is the situation where infinitely many filter steps take place but only finitely many iterates resulting from filter steps are contained in the convergent subsequence. In this case it is required to show that the intermediate filter steps do not affect the convergence of the base algorithm.

We will formulate an algorithm of the described type in this section. In the following sections we then will prove that all limit points are stationary along the lines just described.

### 3.3. The full algorithm.

We now derive a semismooth Newton method for the $l_1$-regularized problem (3.1) for both convex and nonconvex $f$. The method uses the following nonsmooth equation form of the optimality conditions:

$$F_\tau(x) = g(x) - \mathcal{P}_{[-\mu, \mu]} \left( g(x) - \frac{x}{\tau} \right) = 0. \quad \text{(3.12)}$$

This results in the nonsmooth Newton system

$$M(x^k) s^k = -F_\tau(x^k) \quad \text{(3.13)}$$

where $M(x^k)$ denotes a generalized derivative of $F_\tau$ in $x^k$. Since $F_\tau$ is piecewise $C^1$, it is semismooth [40]. Choosing

$$M(x) := (I - D(x)) H(x) + \frac{1}{\tau} D(x), \quad \text{(3.14)}$$

where the diagonal matrix $D(x)$ is defined component-wise

$$D_{ii}(x) \begin{cases} 
  = 0, & |g_i(x) - \frac{1}{\tau} x_i| > \mu \\
  = 1, & |g_i(x) - \frac{1}{\tau} x_i| < \mu \\
  \in \{0, 1\}, & |g_i(x) - \frac{1}{\tau} x_i| = \mu 
\end{cases} \text{ for all } i = 1, ..., n,$$
we have that \(M(x)\) is the Jacobian of one of the smooth active pieces that define \(F_r\) at \(x\). Denoting by \(\partial PW F_r(x)\) the collection of all these \(M(x)\), there holds \(\partial F_r(x) \subset \partial PW F_r(x)\) and \(F_r\) is semismooth w.r.t. \(\partial PW F_r\). In particular, there holds

\[
\|F_r(x + s) - F_r(x) - M(x + s)\| = o(s)
\]

uniformly for all \(M(x+s) \in \partial PW F_r(x+s)\) as \(s \to 0\) [31]. We will work with the unique choice for \(M(x)\) that results when we select \(D_{ii}(x) = 1\) in the case \(|g_i(x) - \frac{1}{\tau}x_i| = \mu\).

Following this rule and introducing the index sets

\[
\mathcal{A}(x) := \{i: |g_i(x) - \frac{1}{\tau}x_i| > \mu\} = \{i: D_{ii}(x) = 0\},
\]

\[
\mathcal{I}(x) := \{i: D_{ii}(x) = 1\},
\]

we can write

\[
M(x) = \begin{pmatrix} H_{\mathcal{A}(x)\mathcal{A}(x)(x)} & H_{\mathcal{A}(x)\mathcal{I}(x)(x)} \\ 0 & \frac{1}{\tau}I \end{pmatrix}.
\]

Now, for a given semismooth Newton step \(s^k\) the decision on accepting \(x^k + s^k\) as new iterate is based on the filter framework presented in the last section. Consequently, we accept the trial point \(x^k + s^k\) whenever it is acceptable for the current filter \(F_k\).

In the convex case, if the trial point \(x^k + s^k\) satisfies all conditions and is contained in \(\Omega\), we accept the Newton step, set \(x^{k+1} = x^k + s^k\), update the filter \(F_{k+1} = F_k \cup \{\theta(x^k)\}\) and start the next iteration. Otherwise we reject the Newton step and perform a step of the globalized fixed point method \(x^{k+1} = x^k + \sigma_k d^k\). In the nonconvex case, we require additional conditions for accepting a Newton step. Details are given below.

The resulting Algorithm 2 then looks as follows:

**Algorithm 2:** Semismooth Newton Method with Multi-Dimensional Filter Globalization (SNF)

0. Initialization: Choose an initial point \(x^0 \in \mathbb{R}^n\), \(\tau > 0\), \(\beta, \gamma \in (0, 1)\) (quasi-Armijo parameters), \(\gamma_F \in (0, 1), F_{-1} = \emptyset\) (filter parameters) and \(\alpha_i > 0\) for \(i \in \{1, 2, 3\}\), \(\eta \in (0, 1)\). Set \(k := 0\), \(\phi_0 := \infty\), \(\rho_0 := \infty\).

1. If \(F_r(x^k) = 0\), STOP.

2. If \(k = 0\) or \(x^k\) was obtained in step 4, add \(\theta(x^k)\) to the filter: \(F_k = F_{k-1} \cup \{\theta(x^k)\}\). Otherwise, set \(F_k = F_{k-1}\).

3. Compute the semismooth Newton step \(s^k\) via \(M(x^k)s^k = -F_r(x^k)\).

4. Set \(x^{k+1} = x^k + s^k\) and check if \(x^{k+1}\) lies in \(\Omega\) and is acceptable for the filter \(F_k\):

\[
\max_{1 \leq j \leq p} \{q_j - \theta_j(x^{k+1})\} \geq \gamma_F \max_{1 \leq j \leq p} \theta_j(x^{k+1}) \quad \forall q \in F_k.
\]

If \(x^{k+1}\) is acceptable for \(F_k\) and either \(f\) is convex or (3.17) holds or (3.18) holds, set \(\phi_{k+1} = \phi_k, \rho_{k+1} = \min\{\rho_k, ||F_r(x^{k+1})||\}\), increment \(k\) and go to step 1.

5. Compute the direction \(d^k = S_{\gamma_k}(G_r(x^k)) - x^k\) and choose a maximal quasi-Armijo step \(\sigma_k \in \{1, \beta, \beta^2, \beta^3, \ldots\} \in (0, 1)\) satisfying

\[
\phi(x^k + \sigma_k d^k) - \phi(x^k) \leq \sigma_k \gamma \Delta^k.
\]

If \(x^{k+1}\) is acceptable for \(F_k\) and either \(f\) is convex or (3.17) holds or (3.18) holds, set \(\phi_{k+1} = \phi_k, \rho_{k+1} = \min\{\rho_k, ||F_r(x^{k+1})||\}\), increment \(k\) and go to step 1.

6. Set \(x^{k+1} = x^k + \sigma_k d^k\) and \(\phi_k = \phi(x^{k+1})\), increment \(k\) and go to step 1.

The algorithm contains two conditions (3.17) and (3.18) in step 4, and requires that one of the two has to hold if \(f\) is not convex. We now introduce these condition. Before doing so, however, we stress that these conditions are only required in the nonconvex case and also only if we want to prove that every accumulation point of
(x^k) is stationary. If we are satisfied with the existence of at least one stationary accumulation point of (x^k), then the conditions (3.17) or (3.18) are not required. We now state these two limited growth conditions for \( \| F_r(x^k + s^k) \| \) and for \( \phi(x^k + s^k) \):

\[
\| F_r(x^{k+1}) \| \leq \eta \rho_k \text{ and } \phi(x^{k+1}) \leq \phi(x^k) + \alpha_1 \sqrt{\| F_r(x^k) \| \| F_r(x^{k+1}) \|}, \quad (3.17)
\]

\[
\phi(x^{k+1}) \leq \varphi_k + \alpha_2 \| F_r(x^{k+1}) \|^{2+\alpha_3}. \quad (3.18)
\]

We introduce the sets

\[
\mathcal{K}_S := \{ k : x^k \text{ was generated by the fixed point method} \},
\]

\[
\mathcal{K}_F := \{ k : x^k \text{ was generated by the Newton method} \}.
\]

This means \( k + 1 \in \mathcal{K}_S \) iff \( x^{k+1} = x^k + \sigma_k d^k \) was obtained in step (6) and \( k + 1 \in \mathcal{K}_F \) iff \( x^{k+1} = x^k + s^k \) was obtained in step (4) before going to step (1).

There holds

\[
\rho_k = \min_{i \in \mathcal{K}_F \cap \{ 1, \ldots, k \}} \{ F_r(x^i) \}
\]

and \( \varphi_k = \phi(x^{l_k(k)}) \), where \( l_k(k) := \max(\mathcal{K}_S \cap \{ 1, \ldots, k \}) \) is the index of the last shrinkage iteration. The trial point \( x^k + s^k \) is accepted as new iterate if it is acceptable for the current filter \( F_k \) and additionally satisfies one of the above conditions (3.17) or (3.18).


4.1. Global Convergence. This section focuses on the analysis of the convergence behavior of our main approach.

**Lemma 4.1.** Let the assumption (A1) or (B1) hold and let \( (x^k) \) be generated by Algorithm 2. Consider a subsequence \( (x^k)^{\mathcal{K}} \) that converges to \( x^* \) and contains infinitely many iterates resulting from semismooth Newton steps, i.e., \( |\mathcal{K} \cap \mathcal{K}_F| = \infty \). Then \( x^* \) is a stationary point.

**Proof.** By assumption, there exists an infinite set \( \mathcal{K} \subset \mathcal{K}_F \) such that \( (x^k)^{\mathcal{K}} \to x^* \). Due to the structure of Algorithm 2 and by Remark 3.10, all assumptions of Lemma 3.9 are satisfied. Hence,

\[
\lim_{k \to \infty} \theta(x^k) = 0.
\]

By continuity, we conclude \( \theta(x^*) = 0 \) and (3.7) thus yields \( F_r(x^*) = 0 \). \( \square \)

**Theorem 4.2.** Let the assumptions (A1)–(A3) or (B1)–(B2) and condition (C) hold and let the sequence \( (x^k) \) be generated by Algorithm 2. Then every accumulation point of the sequence \( (x^k) \) is a stationary point.

**Proof.** Let \( x^* \in \mathbb{R}^n \) be any accumulation point of the sequence \( (x^k) \) and let \( (x^k)^{\mathcal{K}} \) be a corresponding subsequence that converges to \( x^* \).

Case 1: \( |\mathcal{K} \cap \mathcal{K}_F| = \infty \). Then the claim follows directly from Lemma 4.1.

Case 2: \( |\mathcal{K} \cap \mathcal{K}_F| < \infty \). In this case we just compute a finite number of Newton iterations, i.e., there exists \( k_0 \in \mathbb{N} \) such that \( k \in \mathcal{K}_S \) holds for all \( k \geq k_0 \). Hence, we can apply the convergence result of Theorem 3.7 for the quasi-Armijo fixed point method to complete the proof in this case.

So far, the proof was completely independent of the convexity properties of \( f \). In contrast, the discussion of the remaining case depends heavily on the acceptance
criteria in step 4 of the algorithm and consequently we have to distinguish two more cases. We start with the nonconvex case.

Case 3: \(|K \cap \mathcal{K}_F| < \infty, |\mathcal{K}_F| = \infty, f\) nonconvex. Since we perform infinitely many Newton steps, the sequence \((F_r(x^k))_{\mathcal{K}_F}\) converges to zero by Lemma 4.1. But since only finitely many such \(x^k\) are contained in the sequence \((x^k)\), the challenge is to show that the convergence of the quasi-Armijo fixed point method is not disrupted by the intermediate Newton steps, which might not always result in \(\phi\)-descent. Using the relationship between optimality condition (2.4) and the fixed point equation (2.7) we obtain

\[
\|d^k\| = \tau\|F_r(x^k)\| \quad \forall k \geq 0. \quad (4.1)
\]

Assumption (C) yields the Lipschitz constant \(\lambda_{\text{max}}\) for \(g\) on \(\Omega\). Thus, we obtain the Lipschitz constant \(1 + \tau\lambda_{\text{max}}\) for \(G_r\) on \(\Omega\). Since \(P_{[-\mu, \mu]}\) is nonexpansive, this yields that \(F_r(x) = g(x) - P_{[-\mu, \mu]}(-\tau^{-1}G_r(x))\) is Lipschitz continuous on \(\Omega\) with modulus \(\lambda_{\text{max}} + \tau^{-1}(1 + \tau\lambda_{\text{max}}) = 2\lambda_{\text{max}} + \tau^{-1}\). For all \(k + 1 \in \mathcal{K}_S\), this and (4.1) yield

\[
\|F_r(x^{k+1})\| \leq \|F_r(x^k)\| + (2\hat{\lambda}_{\text{max}} + \tau^{-1})\sigma_k\|d^k\| \\
\leq 2(\tau\lambda_{\text{max}} + 1)\|F_r(x^k)\| =: C_1\|F_r(x^k)\|. \quad (4.2)
\]

For all \(k + 1 \in \mathcal{K}_F\), using that \(x^{k+1}\) is acceptable to \(\mathcal{F}_k\), we obtain from Remark 3.10:

\[
\|F_r(x^{k+1})\| \leq \sqrt{n}\|F_r(x^{k+1})\|_{\infty} \leq \frac{\sqrt{n}}{c_0}\|\theta(x^{k+1})\|_{\infty} \leq \frac{\sqrt{n}}{\gamma_{G_0}}\|q\|_{\infty} \quad \forall q \in \mathcal{F}_k. \quad (4.3)
\]

Let \(k_0 := \min \mathcal{K}_F\). Then there holds \(\theta(x^{k_0}) \in \mathcal{F}_k\) for all \(k > k_0\), and thus

\[
\|F_r(x^k)\| \leq \frac{\sqrt{n}}{\gamma_{G_0}}\|\theta(x^{k_0})\|_{\infty} =: C_2 \quad \forall k \in \mathcal{K}_F, \ k > k_0. \quad (4.4)
\]

Further, let \((i_j)_{j \geq 0}\) enumerate all elements of the set \(\{k \in \mathcal{K}_S : k > k_0\}\) in increasing order. Then the set \(J := \{j : i_j \in K\}\) contains infinitely many indices. Defining

\[
\Sigma(r) := \sum_{j=0}^{r-1} (\phi(x^{i_j}) - \phi(x^{i_{j+1}}))
\]

we will use the telescope sum

\[
\phi(x^{i_0}) - \phi(x^*) = \phi(x^{i_0}) - \lim_{J \ni r \to \infty} \phi(x^{i_r}) = \lim_{J \ni r \to \infty} \Sigma(r).
\]

Our approach consists in deriving a lower bound for the right hand side that would exceed the left hand side as \(J \ni r \to \infty\) unless \((F_r(x^k))_{\mathcal{K}_F \cap \mathcal{K}_S} \to 0\).

Therefore, we will discuss the difference \(\phi(x^{i_j}) - \phi(x^{i_{j+1}})\) of two consecutive shrinkage iterates (with possibly other iterates in between). We define the index subsets

\[
\mathcal{K}_F^a := \{k \in \mathcal{K}_F : x^k\) satisfies (3.17)\}, \quad \mathcal{K}_F^b := \{k \in \mathcal{K}_F : x^k\) satisfies (3.18)\}.
\]

and the function

\[
n_a : \mathbb{N} \to \mathbb{N} \quad n_a(k) := |\mathcal{K}_F^a \cap \{k_0 + 1, \ldots, k\}|.
\]
Note that $k_0$ is the index of the very first iterate $x^{k_0}$ obtained by a Newton step. We then have

$$\|F_r(x^k)\| \leq \eta^\alpha(k)\rho_0 \quad \forall \ k \in \mathcal{K}_F.$$  

(4.5)

We now consider $j \geq 0$ and derive a lower bound for $\phi(x^{i_j}) - \phi(x^{i_{j+1}})$. To this end, we distinguish several cases. For abbreviation, let $l := i_j, k := i_{j+1} - 1$.

Sub-case 1: $k = l$ (i.e., $k, k + 1 \in \mathcal{K}_S$):

Applying Lemma 3.2, Lemma 3.5, and inequality (4.2), we obtain

$$\phi(x^i) - \phi(x^{i+1}) \geq -\sigma_k \gamma \Delta_k \geq \frac{\sigma_k \gamma}{\rho} \|d_k\|^2 \geq \xi \gamma \|F_r(x^k)\|^2 \geq \frac{\xi \gamma}{C_1^2} \|F_r(x^{i+1})\|^2.$$

Denoting by $J_1$ the set of all $j \geq 0$ for which this case occurs (recall $l = i_j$), we see that

$\Sigma_1(r) := \sum_{j \in J_1, j < r}(\phi(x^{i_j}) - \phi(x^{i_{j+1}}))$ is bounded below as $r \to \infty$. Furthermore, $\lim_{j \to \infty} \Sigma_1(r) < \infty$ requires either $|J_1| < \infty$ or $(F_r(x^{i_{j+1}}))_{j \in J_1} \to 0$.

Sub-case 2: $l < k \in \mathcal{K}_F$:

Using the same arguments as in sub-case 1, (3.18), and $\phi_{k-1} = \phi(x^l)$, we obtain:

$$\begin{align*}
\phi(x^l) - \phi(x^{k+1}) &= \phi(x^l) - \phi(x^k) + \phi(x^k) - \phi(x^{k+1}) \\
&\geq \phi(x^l) - \phi_{k-1} - \alpha_2\|F_r(x^k)\|^{2+\alpha_3} + \xi \gamma \|F_r(x^k)\|^2 \\
&\geq \frac{1}{C_1^2} (\xi \gamma - \alpha_2\|F_r(x^k)\|^{\alpha_3})\|F_r(x^{k+1})\|^2.
\end{align*}$$

Since $(F_r(x^k))_{k \in \mathcal{K}_F} \to 0$, we see that there exists $k_1 \geq k_0$ such that $\alpha_2\|F_r(x^k)\|^{\alpha_3} < \xi \gamma / 2$ for all $k \in \mathcal{K}_F$ with $k \geq k_1$. Hence, if $J_2$ denotes the set of all $j \geq 0$ for which this sub-case 2 occurs, we see that $\Sigma_2(r) := \sum_{j \in J_2, j < r}(\phi(x^{i_j}) - \phi(x^{i_{j+1}}))$ is bounded below for $r \to \infty$ and that $\lim_{j \to \infty} \Sigma_2(r) < \infty$ requires either $|J_2| < \infty$ or $(F_r(x^{i_{j+1}}))_{j \in J_2} \to 0$.

Sub-case 3: $l + 1, \ldots, r_b - 1 \in \mathcal{K}_F$, $r_b = r_b(j) \in \mathcal{K}_F$, $r_b + 1, \ldots, k \in \mathcal{K}_F$:

Let $r_a = r_a(j)$ be defined as $r_a := \max(\{k_0\} \cup (\mathcal{K}_F \cap \{k_0 + 1, \ldots, r_b - 1\}))$. We obtain

$$\begin{align*}
\phi(x^l) - \phi(x^{k+1}) &= \phi(x^l) - \phi(x^{r_a}) + \sum_{i = r_a}^{k-1} (\phi(x^i) - \phi(x^{i+1})) + \phi(x^k) - \phi(x^{k+1}) \\
&\geq \phi(x^l) - \phi_{r_{a-1}} - \alpha_2\|F_r(x^{r_a})\|^{2+\alpha_3} \\
&\quad - \alpha_1 \sum_{i = r_a}^{k-1}\|F_r(x^i)\|^2 \|F_r(x^{i+1})\|^2 + \xi \gamma \|F_r(x^k)\|^2 \\
&\geq -\alpha_2\|F_r(x^{r_a})\|^{2+\alpha_3} - \alpha_1 \sqrt{\rho_0 C_2} \sum_{i = r_a}^{k-1} \eta^\alpha(r_a) \xi \gamma \|F_r(x^{k+1})\|^2,
\end{align*}$$

where we used estimates as in the first case, inequalities (4.3), (4.4), (4.5) and the conditions (3.17), (3.18). Since the iterate $x^{r_a}$ is acceptable for the filter, we can use inequality (4.3) with $q = \theta(x^{r_a})$. This yields

$$\|F_r(x^{r_a})\| \leq \frac{\sqrt{n}}{\gamma_r C_0} \|\theta(x^{r_a})\|_{\infty} \leq \frac{\sqrt{n} C_0}{\gamma_r C_0} \|F_r(x^{r_a})\| \leq \frac{\sqrt{n} C_0}{\gamma_r C_0} \eta^\alpha(r_a) \rho_0,$$
and thus
\[
\phi(x^i) - \phi(x^{i+1}) \geq -\alpha_1 \left( \sqrt{\frac{\eta C}{\gamma \epsilon \epsilon_0}} \eta^{\alpha_{n+1}} \rho \eta_0 \right)^{2+\alpha_3} - \alpha_1 \sqrt{\rho \eta C_2 \sum_{i=r_0}^{k-1} \eta^{\alpha_{n+1}}} + \frac{\epsilon \eta \tau}{C_1^2} \|F_r(x^{k+1})\|^2.
\]

Let $J_3$ denote all $j \geq 0$ that fall into this sub-case 3. Then, we have $n_a(r_a(j)) \neq n_a(r_a(j'))$ for all $j, j' \in J_3, j \neq j'$.

Hence,
\[
\sum_{j \in J_3} \eta^{n_a(r_a(j))} \leq \sum_{k=0}^{\infty} \eta^{k(2+\alpha_3)} = \frac{1}{1-\eta^{2+\alpha_3}}.
\]

Furthermore, $n_a(r_b(j) + 1) < n_a(r_b(j) + 2) < \cdots < n_a(k) = n_a(i_{j+1} - 1)$, and thus,
\[
\sum_{j \in J_3} \sum_{i=r_a(j)+1}^{i_j-1} \eta^{\alpha_{n+1}} \leq \sum_{k=0}^{\infty} \eta^2 = \frac{1}{1-\sqrt{\eta}}.
\]

we thus see that $\Sigma_3(r) := \sum_{j \in J_3, j < r} (\phi(x^j) - \phi(x^{j+1}))$ is bounded below as $r \to \infty$ and that $\lim_{j \to \infty} \Sigma_3(r) < \infty$ requires either $|J_3| < \infty$ or $(F_r(x^{j+1}))_{j \in J_3} \to 0$.

Sub-case 4: $l+1, \ldots, k \in K^b_r$.

This is the same situation as in sub-case 3, except that there does not exist an iterate $i \in \{l+1, \ldots, k\}$ with $i \in K^b_r$. Similarly as before, but easier, we obtain
\[
\phi(x^i) - \phi(x^{i+1}) = \sum_{i=l}^{k-1} (\phi(x^i) - \phi(x^{i+1})) + \phi(x^k) - \phi(x^{k+1}) \geq -\alpha_1 \sum_{i=l}^{k-1} \|F_r(x^i)\|^\frac{1}{2} \|F_r(x^{i+1})\|^\frac{1}{2} + \frac{\epsilon \eta \tau}{C_1^2} \|F_r(x^{k+1})\|^2.
\]

Let $J_4$ denote all $j \geq 0$ with $i_j \geq k_0$ that fall into this sub-case 4. Then, we have $n_a(i_{j+1}) < n_a(i_{j} + 2) < \cdots < n_a(i_{j+1} - 1)$, and thus,
\[
\sum_{j \in J_4} \sum_{i=i_j}^{i_{j+1}-2} \eta^{\alpha_{n+1}} \leq \sum_{k=0}^{\infty} \eta^2 = \frac{1}{1-\sqrt{\eta}}.
\]

This shows that $\Sigma_4(r) := \sum_{j \in J_4, j < r} (\phi(x^j) - \phi(x^{j+1}))$ is bounded below as $r \to \infty$ and that $\lim_{j \to \infty} \Sigma_4(r) < \infty$ requires either $|J_4| < \infty$ or $(F_r(x^{j+1}))_{j \in J_4} \to 0$.

Taking all cases together, it follows from
\[
\phi(x^0) - \phi(x^*) = \lim_{j \to \infty} \sum_{c=1}^{4} \Sigma_c(r)
\]

that $(F_r(x^{j+1}))_{j \geq 0} \to 0$, since otherwise the limit on the right hand side would be $+\infty$ (note that all $\Sigma_c(r)$ were shown to be bounded below as $r \to \infty$). Since $K$ contains infinitely many indices $i_j$, we conclude $F_r(x^*) = 0$. 


Case 4: $|K \cap K_F| < \infty$, $|K_F| = \infty$, $f$ convex.

Since we perform infinitely many Newton steps, the sequence $(F_\tau(x^k))_{K_F}$ converges to zero by Lemma 4.1. By Lemma 3.1, the sequence $(x^k)$ is bounded and there exists an optimal solution $\bar{x} \in X^*$. We now want to show $(\phi(x^k))_{K_F} \to \phi(\bar{x})$.

Assume that $(\phi(x^k))_{K_F}$ does not converge to $\phi(\bar{x})$. Since $\bar{x}$ is a global minimum of the objective function $\phi$, there then exist $\epsilon > 0$ and a subsequence $L \subset K_F$ with

$$\phi(x^l) \geq \phi(\bar{x}) + \epsilon \quad \forall l \in L.$$ 

From the bounded sequence $(x^l)_{l \in L}$ we can choose a further subsequence $\tilde{L} \subset L$ satisfying $(x^l)_{l \in \tilde{L}} \to \tilde{x}$ and $\phi(\tilde{x}) \geq \phi(\bar{x}) + \epsilon$.

The continuity of $F_\tau$ yields $F_\tau(\tilde{x}) = 0$ and thus $\tilde{x}$ is a global solution, which results in the contradiction $\phi(\tilde{x}) = \phi(\bar{x})$. Hence, we have proved $(\phi(x^k))_{K_F} \to \phi(\bar{x})$.

Remark 4.3. The proof does not use any particular properties of the semismooth Newton steps $s^k$, hence the semismooth Newton method for computing $s^k$ could be replaced by other choices. In particular, the Newton system in step (3) could be replaced by a regularized version of it, see (4.6).

4.2. Fast Local Convergence. The semismooth Newton steps achieve locally q-superlinear convergence under suitable conditions. We now will prove that, under appropriate assumptions, Algorithm 2 turns into a semismooth Newton method after finitely many iterations and thus achieves locally an at least q-superlinear rate of convergence.

For formulating second order sufficient conditions that are suitable to ensure an isolated local solution as well as the invertibility of the matrix in the Newton step computation, we define, as in [28], the following index sets.

**Definition 4.4.** Let $x^*$ be a stationary point of $\phi$. We define

$$A_* := \{i : |g_i(x^*)| = \mu\}, \quad I_* := \{i : |g_i(x^*)| < \mu\}.$$ 

By (2.3), there holds $I_* = \{1, \ldots, n\} \setminus A_*$. 

**Remark 4.5.** Note that these index sets in general depend on the stationary point $x^*$. However, it was shown in [28] that there exists a constant optimal gradient and that the sets $A_*$ and $I_*$ can be determined uniquely whenever $f$ is convex. 

By (2.3), the support of every stationary point $x^*$ is a subset of the index set $A_*$ and, hence, $x_i^* = 0$ for all $i \in I_*$. 

The following result gives a sufficient condition for an isolated local minimum.

**Lemma 4.6.** Let the assumption (A1) or (B1) hold and let $x^*$ be a stationary point of $\phi$ that satisfies the condition

$$\lambda_{\min}(H_{A_*A_*}(x^*)) > 0.$$ 

Then $x^*$ is an isolated (thus also strict) local minimum of $\phi$ and also an isolated stationary point of $\phi$.

**Proof.** See Appendix A.1. 

We thus make the following assumption:
Assumption.

(D) The sequence \((x^k)\) has an accumulation point \(x^*\) satisfying
\[
\lambda_{\min}(H_{A,\lambda}(x^*)) > 0.
\]

Since we already proved that accumulation points are stationary, Assumption (D) allows us to invoke Lemma 4.6. Besides a sufficient optimality condition, this condition also ensures the uniformly bounded invertibility of the matrices \(M(x)\) near \(x^*\), see Lemma 4.7 below.

For robustifying the semismooth Newton steps, we consider the more general case where in the Newton system the generalized derivative \(M(x)\) is replaced by a regularized version \(M_\rho\),
\[
M_\rho(x) := (I - D(x))(H(x) + \rho(\|F_\rho(x)\|)I) + \frac{1}{\tau}D(x).
\]

Here, the function \(\rho : \mathbb{R}_+ \to \mathbb{R}_+\) is assumed to be continuous and monotonically increasing with \(\rho(0) = 0\). We are primarily interested in regularizations of the form \(\rho(t) = c t^p\) with \(p \in (0, 1], c > 0\).

**Lemma 4.7.** Let the assumption (A1) or (B1) hold and let \(x^*\) be a stationary point of \(\phi\) that satisfies
\[
\lambda_{\min}(H_{A,\lambda}(x^*)) > 0.
\]

Then there exist \(\delta > 0\) and \(C > 0\) such that, for all choices of \(\rho : \mathbb{R}_+ \to \mathbb{R}_+\) and all \(x \in B_\delta(x^*)\), the matrix \(M_\rho(x)\) is invertible with \(\|M_\rho(x)^{-1}\| \leq C\).

**Proof.** See Appendix A.2. □

**Theorem 4.8.** Let the assumptions (A1)–(A3) or (B1)–(B2) and the condition (C) hold and let the sequence \((x^k)\) be generated by Algorithm 2. Furthermore, let \(x^* \in \mathbb{R}^n\) be an accumulation point of the sequence \((x^k)\) satisfying condition (D), and let step (3) of Algorithm 2 use the regularized matrix \(M_\rho\) instead of \(M(x)\) with \(\rho : \mathbb{R}_+ \to \mathbb{R}_+\) continuous, monotonically increasing, and \(\rho(0) = 0\) (including the standard case \(\rho \equiv 0\), i.e., \(M_\rho = M\)). Then there holds:

(i) The whole sequence \((x^k)\) converges to \(x^*\) and \(x^*\) is an isolated local minimum as well as an isolated stationary point of \(\phi\).

(ii) There exists \(\bar{k} > 0\) such that \(x^k\) results from a (regularized) semismooth Newton step, i.e., \(k \in K_F\), for all \(k \geq \bar{k}\). In particular, \((x^k)\) converges \(q\)-superlinearly to \(x^*\).

(iii) If, in addition, the Hessian \(H(x)\) is Lipschitz continuous near \(x^*\), then the order of convergence is 2 if \(\rho \equiv 0\) and 1 + \(p\) if \(\rho(t) = O(t^p)\) as \(t \to 0^+\) for some \(p \in (0, 1]\).

**Proof.**

(i): Let \(x^* \in \mathbb{R}^n\) be an accumulation point of the sequence \((x^k)\) with the stated properties. Then, by Theorem 4.2, \(x^*\) is a stationary point, since it also holds true if \(M\) is replaced by \(M_\rho\), see Remark 4.3. Lemma 4.6 yields that \(x^*\) is an isolated local minimum of \(\phi\) and also an isolated stationary point. Since every accumulation point of \((x^k)\) is stationary, \(x^*\) is an isolated accumulation point of \((x^k)\).

Now consider an arbitrary subsequence \((x^k)^K\) that converges to the isolated accumulation point \(x^*\). If we can show \((\|x^{k+1} - x^k\|)^K \to 0\) then a well known result of Moré and Sorensen [36, Lemma 4.10] implies the convergence of the whole sequence \((x^k)\) to \(x^*\). Now, by assumption (D) and Lemma 4.7, the matrix \(M_\rho(x)\) is uniformly
boundedly invertible in a neighborhood of \( x^* \). Hence, there exist \( k_1 \geq 0 \) and \( C > 0 \) such that \( \| M_\rho(x^k)^{-1} \| \leq C \) for all \( k \in K, k \geq k_1 \). Together with equation (4.1), this estimate leads to

\[
\| x^{k+1} - x^k \| \leq \max\{C, \tau\} \| F_\tau(x^k) \| \to \max\{C, \tau\} \| F_\tau(x^*) \| = 0 \quad (K \ni k \to \infty).
\]

As intended, [36, Lemma 4.10] now yields that the entire sequence \( (x^k) \) converges to \( x^* \), which concludes the proof of part (i).

(ii): From part (i) we know that \( x^k \to x^* \). As before, there exist \( \delta_1 > 0 \) and \( C > 0 \) such that

- \( \| M_\rho(x)^{-1} \| \leq C \) for all \( x \in \overline{B}_{\delta_1}(x^*) \subset \Omega \),
- \( x^* \) is the unique stationary point of \( \phi \) on \( \overline{B}_{\delta_1}(x^*) \),
- \( \phi(x) > \phi(x^*) \) for all \( x \in \overline{B}_{\delta_1}(x^*) \setminus \{x^*\} \).

Since \( F_\tau \) is piecewise continuously differentiable and the Hessian \( H \) is bounded on \( \overline{B}_{\delta_1}(x^*) \) (even on \( \Omega \) by assumption), we obtain that \( F_\tau \) is Lipschitz on \( \overline{B}_{\delta_1}(x^*) \) with a Lipschitz constant \( L_1 > 0 \). Similarly, \( \phi \) is Lipschitz continuous on \( \overline{B}_{\delta_1}(x^*) \) with a constant \( L_2 > 0 \).

For all \( x \in \overline{B}_{\delta_1}(x^*) \), the Newton step \( s = -M_\rho(x)^{-1}F_\tau(x) \) is well defined and due to the semismoothness of \( F_\tau \) and the bound on \( \| M_\rho(x)^{-1} \| \), there holds for \( x^+ = x + s \):

\[
\| x^+ - x^* \| = \| M_\rho(x)^{-1}[F_\tau(x^*) + M_\rho(x)(x - x^*) - F_\tau(x)] \|
\leq C\| F_\tau(x^*) + M(x)(x - x^*) - F_\tau(x) \| + C\rho(\| F_\tau(x) \|)\| (I - D(x))(x - x^*) \| \quad (4.7)
= o(\| x - x^* \|) \quad (\| x - x^* \| \to 0).
\]

Now let

\[
\gamma_f := \min\left\{ \eta, \frac{c_0}{\sqrt{n}C_0(1 + \gamma_f)} \right\}, \quad \gamma_s := \min\left\{ \frac{\gamma_f}{L_1C + \gamma_f}, \frac{\alpha_1^2}{(L_2C + \alpha_1)^2} \right\}.
\]

Then \( 0 < \gamma_f \leq \eta < 1 \) and \( 0 < \gamma_s < 1 \) and there exists \( 0 < \delta \leq \delta_1 \) such that

\[
\| x^+ - x^* \| \leq \gamma_s \| x - x^* \| \quad \forall \ x \in \overline{B}_\delta(x^*).
\]

This shows for all \( x \in \overline{B}_\delta(x^*) \):

\[
\| x - x^* \| \leq \| x^+ - x^* \| + \| s \| = \| x^+ - x^* \| + \| M_\rho(x)^{-1}[F_\tau(x)] \| \leq \gamma_s \| x - x^* \| + C\| F_\tau(x) \|.
\]

Hence,

\[
\| x - x^* \| \leq \frac{C}{1 - \gamma_s} \| F_\tau(x) \| \quad \forall \ x \in \overline{B}_\delta(x^*).
\]

Furthermore, using \( F_\tau(x^*) = 0 \) and the definition of \( \gamma_s \),

\[
\| F_\tau(x^+) \| \leq L_1\| x^+ - x^* \| \leq L_1\gamma_s \| x - x^* \| \leq \frac{L_1C\gamma_s}{1 - \gamma_s} \| F_\tau(x) \| \leq \gamma_f \| F_\tau(x) \| \leq \eta \| F_\tau(x) \|.
\]

Since \( x^k \to x^* \), there exists \( k_1 \geq 0 \) such that \( x^k \in \overline{B}_\delta(x^*) \) for all \( k \geq k_1 \), and hence, with the (regularized) semismooth Newton step \( s^k = -M_\rho(x^k)^{-1}F_\tau(x^k) \) and
\[ x^{k+} = x^k + s^k, \]  

there holds:

\[ \|x^{k+} - x^*\| \leq \gamma_s \|x^k - x^*\|, \quad (4.8) \]

\[ \|x^k - x^*\| \leq \frac{C}{1 - \gamma_s} \|F_\tau(x^k)\|, \quad \|x^{k+} - x^*\| \leq \frac{C}{1 - \gamma_s} \|F_\tau(x^{k+})\|, \quad (4.9) \]

\[ \|F_\tau(x^k)\| = \|F_\tau(x^k) - F_\tau(x^*)\| \leq L_1 \|x^k - x^*\|, \quad (4.10) \]

\[ \|F_\tau(x^{k+})\| \leq \gamma_f \|F_\tau(x^k)\|, \quad (4.11) \]

\[ |\phi(x^k) - \phi(x^*)| \leq L_2 \|x^k - x^*\|. \quad (4.12) \]

Let \( k \) be any index with

\[ k \geq k_1, \quad \|F_\tau(x^k)\| < \min_{0 \leq i \leq k} \|F_\tau(x^i)\|. \quad (4.13) \]

Since the algorithm does not terminate finitely and \( 0 < \|F_\tau(x^k)\| \to 0 \), there exist infinitely many such indices \( k \). We now show that if \( k_2 \) is the smallest such index, then \( k \in K_F \) for all \( k \geq k + 1 \).

Let \( k \) satisfy \((4.13)\). Then we have \( x^{k+} \in B_{\delta}(x^*) \subset B_{\delta}(x^*) \). Further, for all \( q \in F_k \) and the corresponding \( r \) with \( q = \theta(x^r) \), there holds:

\[
\max_j (q_j - \theta_j(x^{k+})) - \gamma_f ||\theta(x^{k+})||_\infty \geq ||q||_\infty - (1 + \gamma_F) \|\theta(x^{k+})\|_\infty
\]

\[
\geq \|\theta(x^r)\|_\infty - C_\theta (1 + \gamma_F) \|F_\tau(x^r)\| \|F_\tau(x^{k+})\|_\infty \geq C_\theta \|F_\tau(x^r)\| - C_\theta (1 + \gamma_F) \|F_\tau(x^{k+})\|
\]

\[
\geq \left( \frac{C_\theta}{\sqrt{n}} - C_\theta (1 + \gamma_F) \gamma_f \right) \|F_\tau(x^r)\| \geq 0
\]

by \((4.11)\) and the definition of \( \gamma_f \). Hence, \( x^{k+} \) is acceptable for \( F_k \). Also, there holds

\[ \|F_\tau(x^{k+})\| \leq \gamma_f \|F_\tau(x^k)\| \leq \eta \|F_\tau(x^k)\| = \eta \min_{0 \leq i \leq k} \|F_\tau(x^i)\| \leq \eta \rho_k < \rho_k. \quad (4.14) \]

Thus, the first condition in \((3.17)\) is satisfied by \( x^{k+} \) (replacing \( x^{k+1} \)). We show that also the second condition in \((3.17)\) is satisfied: Since \( x^* \) is the unique local minimum on \( B_{\delta}(x^*) \supset \{x^k, x^{k+}\} \), there holds \( \phi(x^k) > \phi(x^*) \) and \( \phi(x^{k+}) > \phi(x^*) \). If we have \( \phi(x^{k+}) \leq \phi(x^k) \), the second condition of \((3.17)\) is satisfied by \( x^{k+} \) replacing \( x^{k+1} \). If we have \( \phi(x^{k+}) > \phi(x^k) \), the following holds:

\[ |\phi(x^{k+}) - \phi(x^k)| \leq |\phi(x^{k+}) - \phi(x^*)| \leq L_2 \|x^{k+} - x^*\|
\]

\[
\leq L_2 \sqrt{\gamma_s} \|x^k - x^*\| \|x^{k+} - x^*\| \leq \frac{L_2 C \sqrt{\gamma_s}}{1 - \gamma_s} \sqrt{\|F_\tau(x^k)\| \|F_\tau(x^{k+})\|}
\]

\[
\leq \frac{L_2 C \sqrt{\gamma_s}}{1 - \gamma_s} \sqrt{\|F_\tau(x^k)\| \|F_\tau(x^{k+})\|} \leq \alpha_1 \sqrt{\|F_\tau(x^k)\| \|F_\tau(x^{k+})\|},
\]

where we used the Lipschitz continuity of \( \phi \) and the inequalities \((4.8)\) and \((4.9)\). Hence, we have shown that for all \( k \) satisfying \((4.13)\), the semismooth Newton iterate satisfies all requirement such that it is chosen as new iterate. Hence, \( x^{k+1} = x^{k+} = x^k + s^k \) and \( k \in K_F \). Furthermore, \((4.14)\) shows that \( \|F_\tau(x^{k+1})\| < \min_{0 \leq i \leq k} \|F_\tau(x^i)\| \) and thus \( x^{k+1} \) satisfies again \((4.13)\) with \( k \) replaced by \( k + 1 \). Hence, inductively, we see...
\{k : k > k_1\} \subset K_F and thus we can choose \(k = k_1 + 1\). The superlinear convergence follows from (4.7).

(iii): If \(H\) is Lipschitz near \(x^*\), then \(F_r\) is strongly semismooth at \(x^*\) and thus
\[
\|F_r(x^*) + M(x)(x - x^*) - F_r(x)\| = O(\|x - x^*\|^2) \quad (\|x - x^*\| \to 0).
\]

For \(\rho(t) = O(t^p)\), we further obtain:
\[
\rho(\|F_r(x)\|)(I - D(x))(x - x^*)\| = O(\|F_r(x)\|^p\|x - x^*\|) = O(\|x - x^*\|^{1+p})
\]
as \(\|x - x^*\| \to 0\). Thus, the asserted order of convergence follows from (4.7). \(\square\)

5. Numerical Results. In this section we present numerical results and discuss the competitiveness of the semismooth Newton method in comparison with several state of the art \(l_1\)-minimization algorithms. From now on, we will refer to Algorithm 2 as SNF (semismooth Newton filter) method. Based on a test framework in [4], we provide a numerical comparison of different \(l_1\)-optimization methods, that are particulary designed to solve either basis pursuit denoising problems of the form
\[
\min_{x \in \mathbb{R}^n} \|x\|_1 \quad \text{s.t.} \quad \|Ax - b\|_2 \leq \sigma \quad (BP_\sigma)
\]
or corresponding \(l_1\)-regularized problems of the form
\[
\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \mu \|x\|_1 \quad (QP_\mu)
\]
We start with implementational aspects of the SNF algorithm.

5.1. Algorithmic details and implementation. We now briefly describe algorithmic and numerical details of the SNF method. We want to point out that the following considerations focus just on the class of convex problems (\(QP_\mu\)).

Newton system. In each iteration we have to solve the system of equations
\[
M(x^k)s^k = -F_r(x^k),
\]
in order to obtain the next Newton step \(s^k\). Thus, the performance of our algorithm highly depends on efficient strategies for solving those systems. By taking advantage of the structure of the generalized derivative \(M(x^k)\) and using a simple block elimination technique we can reduce the computational complexity and end up with the smaller problem
\[
s_T^k = -\tau F_{r,I}(x^k),
\]
\[
[A^T A]_{AA} s_T^k = -F_{r,A}(x^k) - [A^T A]_{AI} s_T^k, \quad (5.1)
\]
where we set \(A = A(x^k)\) and \(I = I(x^k)\). Instead of solving (5.1) directly, we consider a regularized version of the submatrix of the Hessian \([A^T A]_{AA} + \rho I_{AA}\) with \(\rho := \rho(x^k) = \|F_r(x^k)\|\). This leads to the numerically more robust formulation
\[
[A^T A + \rho I]_{AA} s_A^k = -F_{r,A}(x^k) - [A^T A]_{AI} s_T^k \quad (5.2)
\]
and corresponds to a reformulation of the Newton system with the regularized matrix \(M_{\rho}\), which was already introduced in (4.6). The remaining problem (5.2) is approximately solved by an early terminated (preconditioned) CG-method. Since \(l_1\)-minimization algorithms are usually used for large-scale applications and the matrix \(A\)
typically involves direct or inverse discrete cosine, wavelet, or related transforms, the computational effort of every iteration is dominated by the number of $A$- and $A^T$-calls. Let $C_A$ denote the complexity of applying $A$ or $A^T$. Then the complexity of a single, successful Newton iteration of the SNF algorithm is given by $3C_A + 2C_A \cdot \text{cg-iter}$ (two calls are used to evaluate the right-hand side of equation (5.2) and one $A^T$-call is needed to compute $F_{\tau}(x^{k+1})$ for the filter criterion). Furthermore, if the current iterate is not acceptable to the filter, we have to apply $A$ once more to obtain an alternative shrinkage step. This complexity bound motivates us to solve the linear system (5.2) only approximately, in order to keep the number of CG iterations as low as possible. Hence, we choose a rather mild stopping criterion for the CG method and set the relative tolerance to $10^{-1}$ and the maximum number of iterations to 10.

**Filter.** In our implementation, we choose a filter function $\theta : \mathbb{R}^n \to \mathbb{R}^p_+$ of type (3.8) with the following decomposition pattern

$$I_1 = \{1, \ldots, l\}, \quad I_2 = \{l + 1, \ldots, 2l\}, \ldots, \quad I_p = \{(p-1)l + 1, \ldots, n\}, \quad l = \left\lceil \frac{n}{p} \right\rceil.$$  

We use $p = 1000$, but experiments show that the algorithm is quite insensitive to the choice of $p$. Of course, the required filter storage increases proportional to $p$.

**Continuation.** The continuation with respect to $\mu$ [28] has become a common and successful tool to further improve the performance of $l_1$-optimization algorithms. The idea is to solve the problem $(QP_{\mu_j})$ for a sequence of different $\mu$ values. At first, starting with a usually large parameter $\mu_0 > \mu$, an approximate solution $x_0^*$ of the problem $(QP_{\mu_0})$ is computed. We then decrease the regularization parameter, i.e., we choose $\mu_1$ satisfying $\mu_0 > \mu_1 \geq \mu$, and solve $(QP_{\mu_1})$ with $x_0^*$ as initial point. This procedure is repeated until the current regularization parameter $\mu_j$ coincides with our desired parameter $\mu$ or a termination criterion for the problem $(QP_{\mu_j})$ is satisfied. Practical experience and numerical experiments in [4], [20], [28] and [47] showed that this homotopy scheme can enhance the performance of $l_1$-optimization methods significantly. Encouraged by this general observation we embedded the SNF method in a continuation framework. Particularly, we choose

$$\mu_0 = \max \{ C_0^\text{cont} \|b\|_{\infty}, \mu \}, \quad C_0^\text{cont} = \min \left\{ 0.25, 2.2 \cdot (\mu/\|b\|_{\infty})^{\frac{1}{2}} \right\}$$

and decrease the current homotopy parameter $\mu_j$ according to the following update formula

$$\mu_{j+1} = \max \{ \gamma_j \mu_j, \mu \}, \quad (0, 1] \ni \gamma_j = 1 - C_1^\text{cont} \left( \frac{\mu_j}{\mu_0} \right)^{C_2^\text{cont}},$$

$C_1^\text{cont}, C_2^\text{cont} \in (0, 1)$ (see table 5.1). We set $x_j^* := x_j^{k+1}$ and reduce $\mu_j$ whenever we perform a good Newton step, i.e., when the current Newton iterate $x_j^{k+1} = x_j^k + s_j^k$ satisfies the following decrease condition

$$\|F_{\tau}(x_j^{k+1})\| \leq 0.5 \|F_{\tau}(x_j^k)\|, \quad k \geq 0,$$

where $x_j^k$ denotes the $k$-th iterate of the $j$-th subproblem $(QP_{\mu_j})$, $k > 0$ and $x_j^0 := x_{j-1}^*$.  

**Initial point and stopping tolerance.** We choose $x^0 = 0$ as initial point and terminate SNF when the current residual falls below a given tolerance $\epsilon$, i.e.

$$\|F_{\tau}(x_j^k)\| \leq \epsilon,$$  

(5.3)
where we dropped the additional continuation index for convenience. We want to emphasize that the term \( F_\tau \) has to be understood in its original sense, i.e. here \( F_\tau \) depends on the initial regularization parameter \( \mu \). The tolerance \( \epsilon \) influences the level of accuracy, we will work with \( \epsilon \in \{1, 10^{-1}, 10^{-2}, 10^{-4}, 10^{-6}\} \). Table 5.1 summarizes the default setting of the parameters of SNF.

**Table 5.1**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_{\text{cont}}^1 )</td>
<td>factors for the continuation update formula, ( C_{\text{cont}}^1 = 0.535 )</td>
</tr>
<tr>
<td>( C_{\text{cont}}^2 )</td>
<td>( C_{\text{cont}}^2 = -\log_{10}(0.65) )</td>
</tr>
<tr>
<td>( \beta, \gamma )</td>
<td>parameters for the quasi-Armijo condition, ( \beta = 0.1, \gamma = 0.1 )</td>
</tr>
<tr>
<td>( \tau )</td>
<td>parameter for the nonsmooth function ( F_\tau ), ( \tau = 4 )</td>
</tr>
<tr>
<td>( \gamma_F )</td>
<td>factor for the filter acceptance criterion, ( \gamma_F = 10^{-3} )</td>
</tr>
<tr>
<td>( \text{CG-tol}, \text{CG-maxit} )</td>
<td>parameters to control the accuracy of the CG method, ( \text{CG-tol} = 0.1 ), ( \text{CG-maxit} = 10 )</td>
</tr>
</tbody>
</table>

### 5.2. State-of-the-art methods

In this section we state main ideas and basic structural aspects of several state of the art methods, which will be used later in our numerical comparison. We will work with \( l_1 \)-algorithms that are designed for efficient large-scale optimization and can take advantage of fast implementations of the \( A \) or \( A^T \) application.

**Fixed Point Continuation method (FPC)** [28]. The Fixed Point Continuation method is a first-order algorithm for solving the problem \((QP_\mu)\) or the more general problem (2.1) for convex \( f \). It is a direct realization of the fixed point iteration (2.7) with an additional continuation scheme for the \( l_1 \)-regularization parameter \( \mu \). FPC-BB is an advanced version of FPC that uses Barzilai-Borwein steps to improve performance. The code can be found online at [http://www.caam.rice.edu/~optimization/L1/fpc/](http://www.caam.rice.edu/~optimization/L1/fpc/). All parameters were set to default values.

**FPC Active Set (FPC-AS)** [46]. FPC-AS is an extended two-phase version of the FPC method and is designed to solve \((QP_\mu)\). In the first stage a nonmonotone version of algorithm 2 with a Barzilai-Borwein heuristic for the parameter \( \tau \) is used to determine an active set. Motivated by Greedy algorithms for \( l_1 \)-optimization, FPC-AS then solves a smooth subproblem on this active set with a L-BFGS method. The algorithm is embedded in a continuation scheme. The code is available at [http://www.caam.rice.edu/~optimization/L1/FPC_AS/](http://www.caam.rice.edu/~optimization/L1/FPC_AS/). All parameters were set to default values.

**Gradient Projections for Sparse Reconstruction (GPSR)** [20]. GPSR is based on the well-known projected gradient technique for constrained optimization problems. By splitting \( x = v_1 - v_2 \) into its positive and negative part \( v_1 \) and \( v_2 \) the \( l_1 \)-regularized problem \((QP_\mu)\) is smoothed and reformulated as a quadratic program with positivity constraints for \( v_1 \) and \( v_2 \). In our experiments we tested GPSR with continuation and an alternative version with Barzilai-Borwein step sizes (we refer to GPSR-BB). As recommended by the authors and proposed in [4], all parameters were set to default except the number of continuation steps was set to 40, the \( \text{ToleranceA} \) variable was set to \( 10^{-3} \), and the \( \text{MiniterA} \) variable was set to 1. The code is available at [http://www.lx.it.pt/~mtf/GPSR/](http://www.lx.it.pt/~mtf/GPSR/).

**NESTA** [4]. NESTA is built on Nesterov’s smoothing technique for convex and possibly nonsmooth functions and applies this technique to the constrained \( l_1 \)-problem \((BP_\sigma)\). In [4] it was shown that the performance of Nesterov’s framework can be significantly improved by using a continuation scheme on the smoothing parameter.
that characterizes the level of smoothing of the $l_1$-norm. We tested NESTA with different smoothing parameters, $\mu \in \{0.01, 0.02, 10^{-8}\}$ (unfortunately, the meaning of $\mu$ here is different from its standard use in this paper) and two continuation scenarios, where the number of continuation steps was set to either $T = 4$ or $T = 5$. All other parameters were set to default, except, as proposed in [4], the tolerance variable $\delta$ was set to $10^{-7}$. The code can be found at http://www.stat.stanford.edu/~candes/nesta/.

**Sparse reconstruction by separable approximation (SpaRSA)** [47]. The method SpaRSA was developed to solve the general problem (2.1) and extensions with other nonsmooth regularization terms. For $(QP_\mu)$ it is an iterative shrinkage-based algorithm and therefore resembles FPC. SpaRSA also uses Barzilai-Borwein steps and a continuation technique to accelerate its performance. Online code can be obtained at http://www.lx.it.pt/~mtf/SpaRSA/. Again, as recommended, we set all parameter to default and adopt the parameter modifications in GPSR-BB.

**Spectral projected gradient (SPGL1)** [5]. SPGL1 solves the basis pursuit problem $(BP_\sigma)$ via finding roots of a corresponding single-variable nonlinear equation. This procedure involves solving a sequence of so-called LASSO problems

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2 \quad \text{s.t.} \quad \|x\|_1 \leq \tau$$

for different values of $\tau$. In [5] a spectral projected gradient method is used to efficiently compute approximate solutions of the above least-squares problems. The code is available at http://www.cs.ubc.ca/labs/scl/spgl1/. All parameters were set to default values.

**5.3. Numerical comparison.** To compare SNF with several other methods we use a slightly modified test framework from the NESTA package [4]. The problem setting is identical to the one proposed in [4] and is specified as follows. At first, we generate a sparse signal $\bar{x} \in \mathbb{R}^n$ of length $n = 512^2 = 262144$ with $k = \lfloor n/40 \rfloor = 5553$ nonzero entries. Here, the $k$ different indices $i \in \{1, ..., n\}$ are randomly chosen and the magnitude of each nonzero component is determined via

$$\bar{x}_i = \eta_1(i)10^{d_2(i)/20},$$

where $\eta_1(i) \in \{-1, +1\}$ is a symmetric random sign and $\eta_2(i)$ is uniformly distributed in $[0, 1]$. The signal has dynamic range of $d$ dB and we consider $d \in \{20, 40, 60, 80\}$. The matrix $A \in \mathbb{R}^{m \times n}$ takes $m = n/8 = 32768$ random cosine measurements, i.e. $Ax = (\text{dct}(x))_J$, where $J \subset \{1, ..., n\}$, $|J| = m$, is initialized randomly and dct is the discrete cosine transform. Finally, the input data $b \in \mathbb{R}^m$ is obtained by adding Gaussian noise with standard deviation $\bar{\sigma} = 0.1$ to $A\bar{x}$.

Since NESTA and SPGL1 solve the basis pursuit problem $(BP_\sigma)$ while all other mentioned algorithms solve the unconstrained problem $(QP_\mu)$ we need to compute a corresponding pair $(\sigma, \mu)$ to gain comparable results at first. Therefore, we run SPGL1 to generate an approximate solution of the problem $(BP_{\sigma_0})$ with $\sigma_0 = \sqrt{m + 2\sqrt{2m\bar{\sigma}}}$ and to obtain an estimate $\mu(\sigma_0)$ from the dual solution. Afterwards, we use the SNF algorithm with stopping criterion (5.3) and $\epsilon = 10^{-14}$ to compute a high precision solution $x^*$ of the problem $(QP_{\mu(\sigma_0)})$ and set $\sigma = \|Ax^* - b\|$. Then the problems $(QP_\mu)$ and $(BP_\sigma)$ should be almost equivalent. Now, the SNF method is run again with stopping rule (5.3) and different tolerances to create a series of reference solutions. Following the test framework in [4] we modified the stopping criterion of each
algorithm, the other algorithms now terminate at iteration $k$ when the solution $\tilde{x}^k$ satisfies
\[
\frac{1}{2}\|A\tilde{x}^k - b\|_2^2 + \mu \|\tilde{x}^k\|_1 \leq \frac{1}{2}\|Ax^*_{\text{SNF}} - b\|_2^2 + \mu \|x^*_{\text{SNF}}\|_1,
\]
where $x^*_{\text{SNF}}$ denotes the solution of the SNF method. We consider the dynamic ranges $d \in \{20, 40, 60, 80\}$ and the tolerances $\epsilon \in \{10^0, 10^{-1}, 10^{-2}, 10^{-4}, 10^{-6}\}$. Since an application of $A$ or $A^T$ corresponds to the evaluation of a dct or idct function, the total number $N_A$ of $A$- and $A^T$-calls is an important measure of efficiency. Thus, along with the respective total runtime of each algorithm, our numerical comparison is based on a discussion of the different, achieved $N_A$ values. We report DNC (did not converge), when convergence is not reached after $N_A = 20000$ calls. All tests were performed under MATLAB v7.14 (R2012a) on a MacBook Pro with Intel Core i5 2.4 GHz and 4 GB of memory. The tables 5.2 - 5.5 contain the mean values of $N_A$ and CPU-Time over 10 random trials, the table cells of the three best results from each column are marked in grey background color. Table 5.6 contains approximate $l_0$-norms
\[
\|x\|^\delta_0 := |\{i : |x_i| > \delta\}|
\]
for $\delta \in \{10^{-1}, 10^{-4}\}$. Again, each entry represents the mean value over 10 independent runs, the dynamic range is $d = 40$ dB. As a reference, the approximate $l_0$-norms of the corresponding, high precision solutions are also given.

<table>
<thead>
<tr>
<th>Method</th>
<th>$\epsilon : 10^0$</th>
<th>$\epsilon : 10^{-1}$</th>
<th>$\epsilon : 10^{-2}$</th>
<th>$\epsilon : 10^{-4}$</th>
<th>$\epsilon : 10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
</tr>
<tr>
<td>SNF</td>
<td>93 6.3 201 13.0</td>
<td>279 17.8</td>
<td>319 20.2</td>
<td>381 24.0</td>
<td></td>
</tr>
<tr>
<td>NESTA</td>
<td>252 28.8 DNC</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>SPGL1</td>
<td>55 6.1 90 10.2</td>
<td>123 14.4</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>GPSR</td>
<td>557 43.2 584 45.3</td>
<td>758 59.0</td>
<td>1313 102.4</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>GPSR-BB</td>
<td>384 34.6</td>
<td>431 39.1</td>
<td>470 43.0</td>
<td>684 64.4</td>
<td>1032 99.2</td>
</tr>
<tr>
<td>SpaRSA</td>
<td>476 38.5</td>
<td>514 41.7</td>
<td>540 44.0</td>
<td>606 49.4</td>
<td>704 58.0</td>
</tr>
<tr>
<td>FPC</td>
<td>376 26.7</td>
<td>416 29.7</td>
<td>612 44.0</td>
<td>1272 105.9</td>
<td>2327 203.7</td>
</tr>
<tr>
<td>FPC-BB</td>
<td>152 13.2</td>
<td>172 15.0</td>
<td>195 17.2</td>
<td>741 71.8</td>
<td>DNC</td>
</tr>
<tr>
<td>FPC-AS</td>
<td>80 6.5</td>
<td>201 14.6</td>
<td>282 20.5</td>
<td>374 26.1</td>
<td>457 33.1</td>
</tr>
</tbody>
</table>

It was observed in [4, 33] that $l_1$-optimization algorithms react sensitively on changes of the dynamic range and their performances usually deteriorate with increasing dynamic range. Our experiments also confirm this behaviour for the modified test framework.

SPGL1 is very efficient at low and middle precisions. However, SPGL1 does not converge in the high precision examples. In the low precision case, it takes about 4 times as many $A$ applications on the 80 dB signal as on the 20 dB signal (for lower tolerances this factor diminishes to 3).

The Barzilai-Borwein version of GPSR outperforms the regular GPSR version in both number of $A$- and $A^T$-calls and CPU time, though it cannot catch up with the...
Table 5.3
Total number of $A$- and $A^T$-calls $N_A$ and CPU time (in sec.) averaged over 10 independent runs with dynamic range 40 dB (best NESTA configuration was used: $\mu = 10^{-8}, T = 4$)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\epsilon$ : $10^0$</th>
<th>$\epsilon$ : $10^{-1}$</th>
<th>$\epsilon$ : $10^{-2}$</th>
<th>$\epsilon$ : $10^{-4}$</th>
<th>$\epsilon$ : $10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_A$</td>
<td>time</td>
<td>$N_A$</td>
<td>time</td>
<td>$N_A$</td>
</tr>
<tr>
<td>SNF</td>
<td>184</td>
<td>12.3</td>
<td>418</td>
<td>27.6</td>
<td>527</td>
</tr>
<tr>
<td>NESTA</td>
<td>301</td>
<td>34.5</td>
<td>DNC</td>
<td>-</td>
<td>DNC</td>
</tr>
<tr>
<td>SPGL1</td>
<td>93</td>
<td>10.4</td>
<td>164</td>
<td>19.7</td>
<td>226</td>
</tr>
<tr>
<td>GPSR</td>
<td>469</td>
<td>36.4</td>
<td>699</td>
<td>54.6</td>
<td>904</td>
</tr>
<tr>
<td>GPSR-BB</td>
<td>415</td>
<td>37.8</td>
<td>483</td>
<td>44.3</td>
<td>640</td>
</tr>
<tr>
<td>SpaRSA</td>
<td>468</td>
<td>38.0</td>
<td>531</td>
<td>43.1</td>
<td>582</td>
</tr>
<tr>
<td>FPC</td>
<td>370</td>
<td>26.4</td>
<td>699</td>
<td>52.0</td>
<td>1151</td>
</tr>
<tr>
<td>FPC-BB</td>
<td>160</td>
<td>14.0</td>
<td>219</td>
<td>19.5</td>
<td>619</td>
</tr>
<tr>
<td>FPC-AS</td>
<td>122</td>
<td>10.0</td>
<td>381</td>
<td>29.0</td>
<td>464</td>
</tr>
</tbody>
</table>

Table 5.4
Total number of $A$- and $A^T$-calls $N_A$ and CPU time (in sec.) averaged over 10 independent runs with dynamic range 60 dB (best NESTA configuration was used: $\mu = 0.01, T = 5$)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\epsilon$ : $10^0$</th>
<th>$\epsilon$ : $10^{-1}$</th>
<th>$\epsilon$ : $10^{-2}$</th>
<th>$\epsilon$ : $10^{-4}$</th>
<th>$\epsilon$ : $10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_A$</td>
<td>time</td>
<td>$N_A$</td>
<td>time</td>
<td>$N_A$</td>
</tr>
<tr>
<td>SNF</td>
<td>377</td>
<td>25.1</td>
<td>707</td>
<td>46.7</td>
<td>830</td>
</tr>
<tr>
<td>NESTA</td>
<td>404</td>
<td>46.4</td>
<td>DNC</td>
<td>-</td>
<td>DNC</td>
</tr>
<tr>
<td>SPGL1</td>
<td>149</td>
<td>17.1</td>
<td>243</td>
<td>29.7</td>
<td>320</td>
</tr>
<tr>
<td>GPSR</td>
<td>580</td>
<td>45.2</td>
<td>1014</td>
<td>79.7</td>
<td>1306</td>
</tr>
<tr>
<td>GPSR-BB</td>
<td>439</td>
<td>40.1</td>
<td>624</td>
<td>58.4</td>
<td>880</td>
</tr>
<tr>
<td>SpaRSA</td>
<td>527</td>
<td>42.9</td>
<td>604</td>
<td>49.7</td>
<td>667</td>
</tr>
<tr>
<td>FPC</td>
<td>502</td>
<td>36.0</td>
<td>1266</td>
<td>105.1</td>
<td>1911</td>
</tr>
<tr>
<td>FPC-BB</td>
<td>212</td>
<td>18.8</td>
<td>772</td>
<td>74.9</td>
<td>1392</td>
</tr>
<tr>
<td>FPC-AS</td>
<td>158</td>
<td>13.2</td>
<td>599</td>
<td>44.3</td>
<td>695</td>
</tr>
</tbody>
</table>

results of SNF. In general, GPSR-BB requires twice as many $A$- and $A^T$-calls at 80 dB than at 20 dB. The runtimes for 80 dB and 20 dB differ by a factor between 1.25 and 2.75.

SpaRSA needs a comparatively large number of $A$- and $A^T$-calls at low dynamic range and low precision tests. For fixed dynamic range it only requires about 1.5–2 times as many $A$- and $A^T$-calls to compute a very accurate solution as in the low precision case. Besides, SpaRSA shows good performance with large dynamic range and generally requires less $A$- and $A^T$-calls than SNF (see tables 5.4, 5.5). However, concerning computational time, SpaRSA does not succeed in outperforming SNF.

The NESTA algorithm did not converge in most of our tests. This can be traced back to NESTA’s smoothing of the $l_1$-norm that is used in the algorithm and the resulting relatively low sparsity of NESTA’s solutions (see also table 5.6). Thus, with increasing accuracy NESTA seems to fail at sufficiently decreasing the $l_1$-norm of its iterates and satisfying condition (5.4). Nevertheless the results of the low precision problem indicate that NESTA is a very robust method regarding changes of the dynamic range.

At low and middle precision the FPC-BB method outperforms its regular FPC
A SEMISMOOTH NEWTON METHOD WITH M.-D. FILTER GLOBALIZATION

Table 5.5

Total number of $A$- and $A^T$-calls $N_A$ and CPU time (in sec.) averaged over 10 independent runs with dynamic range 80 dB (best NESTA configuration was used: $\mu = 0.01, T = 5$)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\epsilon : 10^0$</th>
<th>$\epsilon : 10^{-1}$</th>
<th>$\epsilon : 10^{-2}$</th>
<th>$\epsilon : 10^{-4}$</th>
<th>$\epsilon : 10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
<td>$N_A$ time</td>
</tr>
<tr>
<td>SNF</td>
<td>658  43.0  935  61.1  1093  71.3  1224  79.3  1288  83.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NESTA</td>
<td>464  53.2  DNC -  DNC -  DNC -  DNC -  DNC -  DNC -  DNC -</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPGL1</td>
<td>243  28.3  320  38.3  404  48.9  DNC -  DNC -  DNC -  DNC -</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPSR</td>
<td>1840 145.1 2422 192.2 2651 210.5 3832 304.6 3832 304.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPSR-BB</td>
<td>498  45.8  815  77.5 1060 102.0 1799 176.0 2135 209.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SpaRSA</td>
<td>540  43.9  676  56.4  760  63.6  964  81.9  1033  87.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC</td>
<td>951  70.1  1934 162.6 2731 238.3 4944 442.6 5982 547.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC-BB</td>
<td>277  25.2  1171 114.9 1966 194.3  DNC -  DNC -  DNC -  DNC -</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC-AS</td>
<td>341  25.6  503  37.8  646  48.2  712  55.3  925  72.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.6

Approximate $l_0$-norms averaged over 10 independent runs with dynamic range 40 dB (best NESTA configuration was used: $\mu = 0.02, T = 4$)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\epsilon : 10^0$</th>
<th>$\epsilon : 10^{-1}$</th>
<th>$\epsilon : 10^{-2}$</th>
<th>$\epsilon : 10^{-4}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$|x|_0^{10^{-1}}$</td>
<td>$|x|_0^{10^{-4}}$</td>
<td>$|x|_0^{10^{-1}}$</td>
<td>$|x|_0^{10^{-4}}$</td>
</tr>
<tr>
<td>Opt. sol. $x^*$</td>
<td>18858 22462 19002 22638 19103 22845 19278 23067</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNF</td>
<td>25945 33355 19364 23331 19108 22882 19278 23067</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NESTA</td>
<td>257916 262140 DNC -  DNC -  DNC -  DNC -  DNC -  DNC -</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPGL1</td>
<td>29234 41164 19616 24621 19167 23242 19278 23068</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPSR</td>
<td>24879 39735 19701 23527 19186 22966 19278 23068</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPSR-BB</td>
<td>16548 24322 19725 24418 19190 23003 19278 23068</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SpaRSA</td>
<td>16324 22487 19599 24611 19160 23320 19278 23074</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC</td>
<td>34845 56340 19772 23635 19188 22964 19278 23068</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC-BB</td>
<td>35134 57485 19772 23635 19188 22964 19278 23068</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPC-AS</td>
<td>22552 31660 19093 24332 19118 22877 19278 23066</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

version and converges much faster. For fixed dynamic range the performance of both methods degrades as the stopping tolerance is reduced, requiring about 6–10 times more iterations. Furthermore, FPC-BB did not converge for all high precision examples. FPC-AS generally performed very well; it takes about 2–4 times as many $A$- and $A^T$-calls at 80 dB than at 20 dB. For dynamic range $d \in \{20, 40\}$ about 6–7 times as many $A$- and $A^T$-calls are used to compute a very accurate solution as to compute a low precision solution, whereas SNF only needs 3–4 times more calls.

The proposed method SNF compares quite positively to the other solvers. It performs especially well at low dynamic range. The tables 5.1 and 5.2 demonstrate the efficiency of SNF and its competitiveness in this regime. Although SNF generally is a bit more sensitive to increasing dynamic range than, e.g., GPSR-BB or SpaRSA, its performance consistently stays competitive with the other methods. Moreover, our experiments in tables 5.1–5.5 as well as table 5.6 show the particular strength of SNF in efficiently computing high accuracy solutions.

6. Conclusions. In this paper, we proposed and investigated a globalized semismooth Newton method for solving large scale $l_1$-optimization problems. A novel combination of the semismooth Newton method, the filter globalization technique, and a
quasi-Armijo fixed point method was presented that converges globally in both the convex and the nonconvex setting. Moreover, under appropriate conditions, transition to fast q-superlinear convergence was established. Finally, the algorithm was embedded in a continuation scheme for the parameter µ and investigated numerically in a class of convex l1-regularized least square problems. Although each iteration incorporates the approximate (matrix-free) solution of a linear system, our experiments indicate the competitiveness of SNF and demonstrated its particular efficiency in obtaining high accuracy solutions.

Appendix.

A.1. Proof of Lemma 4.6. Proof. Due to the continuity of the Hessian $H$ and the additional assumption, there exist $\epsilon > 0$ and suitable constants $\lambda, c_1, c_2, \kappa > 0$, such that, for all $x \in B_\epsilon(x^*)$:

$$(d_{A_i})^T H_{A_iA_i}(x) d_{A_i} \geq \lambda \|d_{A_i}\|^2, \quad \max_{i \in I_*} |g_i(x)| \leq \mu - \kappa.$$  

Now, let $x^* \in B_\epsilon(x^*)$ be any stationary point of $\phi$ (including $x = x^*$). For all $i \in I_*$ we then have $|g_i(x)| \leq \mu - \kappa$ and $x_i = 0$ by (2.3). For all $i \in A_*$ we conclude from (2.3) that

$$g_i(x) d_i + \mu(|x_i + d_i| - |x_i|) \geq 0 \quad \forall d_i \in \mathbb{R}.$$  

Now, Taylor expansion yields for all $d$ with $x + d \in B_\epsilon(x^*)$ and suitable $\tau \in [0, 1]$:

$$\phi(x + d) - \phi(x) = \sum_{i \in A_*} \{g_i(x) d_i + \mu(|x_i + d_i| - |x_i|)\} + \sum_{i \in I_*} \{g_i(x) d_i + \mu d_i\}$$  

$$+ \frac{1}{2} d^T H(x + \tau d)d \geq \kappa \|d_{A_*}\|_1 + \frac{\lambda}{2} \|d_{A_*}\|^2 - \frac{c_1}{2} \|d_{I_*}\|^2 - c_2 \|d_{I_*}\| \|d_{A_*}\|.$$  

We fix $\delta > 0$ satisfying $c_1 \delta^2 + 2 c_2 \delta < \lambda$. In the case $\|d_{I_*}\| \leq \delta \|d_{A_*}\|$ we get

$$\phi(x + d) - \phi(x) \geq \kappa \|d_{I_*}\|_1 + \frac{\lambda}{2} \|d_{A_*}\|^2 - \left(\frac{c_1 \delta^2}{2} + c_2 \delta\right) \|d_{I_*}\| > \kappa \|d_{I_*}\|_1 \geq 0.$$  

Otherwise, if we have $\|d_{I_*}\| > \delta \|d_{A_*}\|$, then

$$\phi(x + d) - \phi(x) \geq \kappa \|d_{I_*}\|_1 + \frac{\lambda}{2} \|d_{A_*}\|^2 - \left(\frac{c_1}{2} \delta + \frac{c_2}{\delta}\right) \|d_{I_*}\|^2.$$  

The last expression is strictly positive whenever $\|d_{I_*}\| < \frac{2 \delta \kappa}{c_1 \delta + 2 c_2}$ is satisfied. Hence, with $\delta > 0$ given above there holds for all $\|d\| < \rho_0 := \frac{2 \delta \kappa}{c_1 \delta + 2 c_2}$ with $x + d \in B_\epsilon(x^*)$:

$$\phi(x + d) - \phi(x) > 0.$$  

This, in particular, holds true for $x = x^*$. We conclude that $x^*$ is a strict local minimum and that on $B_{\rho}(x^*)$ with $\rho := \min\{\epsilon, \rho_0\}$ there does not exist a stationary point of $\phi$ different from $x^*$. As a consequence, $x^*$ is also an isolated stationary point. \[\square\]
A.2. Proof of Lemma 4.7. We use the following result:

**Lemma A.1.** Let symmetric matrices $H, \Lambda \in \mathbb{R}^{n \times n}$ and an index set $J_* \subseteq \{1, \ldots, n\}$ be given. Assume that $\Lambda$ is positive semidefinite and that the submatrix $H_{J, J_*}$ is positive definite with

$$
\lambda_{\min}^* := \lambda_{\min}(H_{J, J_*}) > 0,
$$

Then there exists a constant $C > 0$, only depending on $\lambda_{\min}^*, \|H\|$, and $\tau > 0$, such that, for all index sets $J$ satisfying $J \subseteq J_*$ and corresponding complements $L = \{1, \ldots, n\} \setminus J$, the matrix

$$
V_J := \begin{pmatrix}
H_{J, J} + \Lambda_J J & H_{J, L} \\
0 & \frac{1}{\tau} I
\end{pmatrix}
$$

(A.1)

is invertible with $\|V_J^{-1}\| \leq C$.

**Proof.** Let $J$ and $L$ be as stated. In the case $J = \emptyset$ there holds $V_\emptyset = \frac{1}{\tau} I$ and thus we can choose $C = \tau$.

Now let $J \neq \emptyset$. Then, by assumption, $\lambda_{\min}(H_{J, J} + \Lambda_J J) \geq \lambda_{\min}^* > 0$, hence

$$
\|\left( H_{J, J} + \Lambda_J J \right)^{-1} \| \leq \frac{1}{\lambda_{\min}^*}.
$$

For an arbitrary vector $r = \begin{pmatrix} r_1^T, r_2^T \end{pmatrix}^T$, we consider the linear system

$$
\begin{pmatrix}
H_{J, J} + \Lambda_J J & H_{J, L} \\
0 & \frac{1}{\tau} I
\end{pmatrix}
\begin{pmatrix}
s_1 \\
s_2
\end{pmatrix} =
\begin{pmatrix}
r_1 \\
r_2
\end{pmatrix}.
$$

(A.2)

Block backward substitution yields

$$
\|s_2\| = \tau \|r_2\|, \quad \|s_1\| \leq \frac{1}{\lambda_{\min}^*} (\|r_1\| + \tau \|H_{J, L}\| \|r_2\|).
$$

This yields

$$
\|s\| \leq \left( \tau + \frac{1 + \tau \|H\|}{\lambda_{\min}^*} \right) \|r\| =: C \|r\|.
$$

Now we prove Lemma 4.7:

**Proof.** By the definition of $I_*$ and due to the stationarity condition (2.3) that holds at $x = x^*$, we conclude $|g_i(x^*)| < \mu$ and $x_i^* = 0$ for all $i \in I_*$. Hence, by continuity, there exists $\delta_1 > 0$, such that

$$
\left| g_i(x) - \frac{x_i}{\tau} \right| < \mu \quad \forall x \in B_{\delta_1}(x^*), \forall i \in I_*.
$$

In terms of the index sets defined in (3.15), this just means

$$
I_* \subseteq I(x), \quad A(x) \subseteq A_* \quad \forall x \in B_{\delta_1}(x^*). \quad (A.3)
$$

For every $J \subset A_*$ and $L = \{1, \ldots, n\} \setminus J$, we define

$$
M_{\rho, J}(x) := \begin{pmatrix}
H_{J, J}(x) + \rho(\|F_\tau(x)\|)I & H_{J, L}(x) \\
0 & \frac{1}{\tau} I
\end{pmatrix}, \quad x \in B_{\delta_1}(x^*).
$$
Thus exists $0 < \delta < \delta_1$ such that, for all $J \subset A$, $L = \{1, \ldots, n\} \backslash J$ and all choices of $\rho : \mathbb{R}_+ \to \mathbb{R}_+$, the matrix $M_{\rho,J}(x^*)$ is invertible with $\|M_{\rho,J}(x^*)^{-1}\| < C$. By continuity of $x \mapsto M_{\rho,J}(x)$, there thus exists $0 < \delta < \delta_1$ such that, for all $J \subset A$, $L = \{1, \ldots, n\} \backslash J$, all choices of $\rho : \mathbb{R}_+ \to \mathbb{R}_+$, and all $x \in B_\delta(x^*)$, there holds $\|M_{\rho,J}(x)^{-1}\| < C$. Since $A(x) \subset A$, and $I(x) = \{1, \ldots, n\} \backslash A(x)$ for all $x \in B_\delta(x^*)$, this yields, since $M_{\rho}(x) = M_{\rho,A(x)}(x)$:

$$\|M_{\rho}(x)^{-1}\| \leq C \quad \forall \ x \in B_\delta(x^*).$$

\[\square\]

REFERENCES


A SEMISMOOTH NEWTON METHOD WITH M.-D. FILTER GLOBALIZATION


