Les Cahiers du GERAD ISSN: 0711–2440

A Levenberg-Marquardt method for nonsmooth regularized least squares

A. Y. Aravkin, R. Baraldi, D. Orban

G–2022–58

December 2022

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2022
– Bibliothèque et Archives Canada, 2022

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2022
– Library and Archives Canada, 2022

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

A Levenberg-Marquardt method for nonsmooth regularized least squares

Aleksandr Y. Aravkin [a]

Robert Baraldi [b]

Dominique Orban [c]

[a] *Department of Applied Mathematics, University of Washington, Seattle, WA, 98195, USA*

[b] *Optimization and Uncertainty Quantification, Sandia National Laboratories, Albuquerque, NM, 87125, USA*

[c] *GERAD and Department of Mathematics and Industrial Engineering, Polytechnique Montréal, Montréal (Qc), Canada, H3C 3A7*

saravkin@uw.edu
rjbaral@sandia.gov
dominique.orban@gerad.ca

December 2022
Les Cahiers du GERAD
G–2022–58

Abstract : We develop a Levenberg-Marquardt method for minimizing the sum of a smooth nonlinear least-squares term $f(x) = \frac{1}{2}\|F(x)\|_2^2$ and a nonsmooth term h. Both f and h may be nonconvex. Steps are computed by minimizing the sum of a regularized linear least-squares model and a model of h using a first-order method such as the proximal gradient method. We establish global convergence to a first-order stationary point of both a trust-region and a regularization variant of the Levenberg-Marquardt method under the assumptions that F and its Jacobian are Lipschitz continuous and h is proper and lower semi-continuous. In the worst case, both methods perform $O(\epsilon^{-2})$ iterations to bring a measure of stationarity below $\epsilon \in (0, 1)$. We report numerical results on three examples: a group-lasso basis-pursuit denoise example, a nonlinear support vector machine, and parameter estimation in neuron firing. For those examples to be implementable, we describe in detail how to evaluate proximal operators for separable h and for the group lasso with trust-region constraint. In all cases, the Levenberg-Marquardt methods perform fewer outer iterations than a proximal-gradient method with adaptive step length and a quasi-Newton trust-region method, neither of which exploit the least-squares structure of the problem. Our results also highlight the need for more sophisticated subproblem solvers than simple first-order methods.

Keywords : Regularized optimization, nonsmooth optimization, nonconvex optimization, nonlinear least squares, Levenberg-Marquardt method, proximal gradient method

Acknowledgements: R. Baraldi acknowledge that this research was sponsored by the Department of Energy Office of Science, Office of Advanced Scientific Computing Research's John von Neumann Fellowship. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE–NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. D. Orban acknowledge that the research was partially supported by an NSERC Discovery Grant.

1 Introduction

We consider the problem

$$\underset{x}{\text{minimize}}\ \ f(x) + h(x), \qquad f(x) = \tfrac{1}{2}\|F(x)\|_2^2, \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable and $h : \mathbb{R}^n \to \mathbb{R}$ is proper and lower semi-continuous; we allow h to be nonsmooth and nonconvex. In practice, f is often a data-misfit term while h is a regularizer designed to promote desirable properties in the solution, such as sparsity. Numerous applications investigated in the nonsmooth regularized optimization literature actually have the structure (1), including basis pursuit denoising [14, 28], sparse factorization and dictionary learning [2], and sparse total least squares [30]. Yet nonsmooth numerical methods do not exploit the least-squares structure, nor accommodate general nonsmooth regularizers.

We describe two methods for (1): a quadratic regularization variant and trust-region variant inspired by the method of Levenberg [19] and Marquardt [21], denoted `LM` and `LMTR` respectively. Steps are computed by approximately minimizing simpler nonsmooth iteration-dependent Gauss-Newton-type models. Our algorithmic realizations utilize first-order methods, such as the proximal gradient method or the quadratic regularization method of Aravkin et al. [1], to solve the subproblems. The trust-region approach allows for any arbitrary trust-region norm, which, in practice, is influenced by nonconvex subproblem tractibility. For both algorithms, we establish global convergence in terms of an optimality measure describing achievable decrease by a single proximal gradient step. Additionally, we derive a worst-case complexity bound of $\mathcal{O}(1/\epsilon^2)$ iterations to bring the stationarity measure below a tolerance of $\epsilon \in (0, 1)$ for `LM` and `LMTR`, i.e., the presence of a nonsmooth term in the objective yields a complexity bound of the same order as in the smooth case.

We provide implementation details and illustrate the performance of our methods on several numerical examples, including basis pursuit denoise with group-lasso regularization, nonlinear support vector machine with $\ell_{1/2}^{1/2}$-norm regularization, and a sparse parameter estimation example taken from the Fitzhugh-Nagumo model of neuron firing. Our methods exhibit favorable performance under certain conditions with respect to previous work Aravkin et al. [1]. We additionally provide efficient, open-source software implementations of `LM` and `LMTR` as a package in the Julia language [3]. We find that exploiting the least-squares structure yields few `LM` and `LMTR` outer iterations, a well-known benefit in smooth optimization. The cost incurred is a large number of inner iterations, i.e, spent solving the subproblem. Thus, the results highlight the need for more sophisticated methods to minimize the sum of a linear least-squares term and a nonsmooth regularizer.

Related research

The present research is based on the framework laid out by Aravkin, Baraldi, and Orban [1]. The convergence and complexity of our trust-region Levenberg-Marquardt implementation follow directly from the general results of [1]. To the best of our knowledge, the trust-region literature does not explicitly cover the case of a nonlinear least-squares smooth objective with a nonsmooth regularizer other than a penalty term even though numerous applications exhibit that structure. See [13] for background and an extensive treatment.

A large portion of the literature focuses on h convex and/or globally Lipschitz continuous, e.g., Cartis et al. [11], Grapiglia et al. [17] and references therein. We do not attempt to give a comprehensive account of that literature here as we focus on significantly weaker assumptions. While many methods exist in the first-order literature, e.g., [12], few can effectively utilize any significant curvature information. Proximal Newton methods [18] require solutions to nontrivial proximal operators and positive semi-definiteness of the Hessian. The small number of references that allow both f and h to be nonconvex that we are aware of include: Li and Lin [20], who design accelerations of the proximal gradient method under the assumption that $f + h$ is coercive; Bolte et al. [8] who design an alterating method for cases where

$h(x) = h_1(x_1) + h_2(x_2)$ and (x_1, x_2) is a partition of x; Stella et al. [26] who propose a linesearch limited-memory BFGS method named PANOC; Themelis et al. [27] who propose a nonmonotone linesearch proximal quasi-Newton method named ZeroFPR based on the forward-backward envelope; and Boţ et al. [9], who study a proximal method with momentum. The last three converge if $f + h$ satisfies the Kurdyka-Łojasiewicz (KŁ) assumption. Moreover, while all include (1) as a special case, few exploit any curvature information and none are specific to the least-squares structure. The algorithms presented here, like those of [1], require no such coercivity or KŁ assumptions.

Notation

We use $\|\cdot\|$ to represent a generic, but fixed, norm on $\mathbb{R}^n$ or $\mathbb{R}^m$. The unit ball defined by that norm is $\mathbb{B}$, and $x + \Delta\mathbb{B}$ is the ball centered at x of radius $\Delta > 0$. For an integer $q \geq 1$, $\|\cdot\|_q$ is the ℓ_q-norm and $\mathbb{B}_q$ is the unit ball in the ℓ_q-norm. If $A \subseteq \mathbb{R}^n$, $\chi(\cdot \mid A)$ is the indicator of A, i.e., the function whose value is 0 if $x \in A$ and $+\infty$ otherwise. Unless otherwise noted, if A is a matrix, $\|A\|$ denotes the spectral norm of A, i.e., its largest singular value. We use $J(x) : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ to denote the Jacobian of F at x.

2 Background

Definition 1 (Limiting subdifferential)**.** Consider $\phi : \mathbb{R}^n \to \overline{\mathbb{R}}$ and $\bar{x} \in \mathbb{R}^n$ with $\phi(\bar{x}) < \infty$. We say that $v \in \mathbb{R}^n$ is a *regular subgradient* of ϕ at $\bar{x}$, and we write $v \in \hat{\partial}\phi(\bar{x})$ if

$$\liminf_{x \to \bar{x}} \frac{\phi(x) - \phi(\bar{x}) - v^T(x - \bar{x})}{\|x - \bar{x}\|_2} \geq 0.$$

The set of regular subgradients is also called the *Fréchet subdifferential*. We say that v is a *general subgradient* of ϕ at $\bar{x}$, and we write $v \in \partial\phi(\bar{x})$, if there are sequences $\{x_k\}$ and $\{v_k\}$ such that

$$x_k \to \bar{x}, \quad \phi(x_k) \to \phi(\bar{x}), \quad v_k \in \hat{\partial}\phi(x^k) \text{ and } \; v^k \to v.$$

The set of general subgradients is called the *limiting subdifferential*.

Proposition 1 (25, Theorem 10.1)**.** *If $\phi : \mathbb{R}^n \to \overline{\mathbb{R}}$ is proper and has a local minimum at $\bar{x}$, then $0 \in \hat{\partial}\phi(\bar{x}) \subseteq \partial\phi(\bar{x})$. If ϕ is convex, the latter condition is also sufficient for $\bar{x}$ to be a global minimum. If $\phi = f + h$ where f is continuously differentiable on a neighborhood of $\bar{x}$ and h is finite at $\bar{x}$, then $\partial\phi(\bar{x}) = \nabla f(\bar{x}) + \partial h(\bar{x})$.*

If $0 \in \hat{\partial}\phi(\bar{x})$, we say that $\bar{x}$ is *first-order stationary* for ϕ. Under our assumptions,

$$x \text{ is first-order stationary for (1)} \quad \Longleftrightarrow \quad 0 \in J(x)^T F(x) + \partial h(x). \tag{2}$$

The proximal gradient method [16] applied to a regularized objective $f(x) + h(x)$ where f is differentiable is defined by the iteration

$$x_{k+1} \in \operatorname*{prox}_{\nu h}(x_k - \nu\nabla f(x_k)) \qquad (k \geq 0), \tag{3}$$

where $\nu > 0$ is a steplength and the *proximal operator* is defined as

$$\operatorname*{prox}_{\nu h}(y) := \operatorname*{argmin}_u \tfrac{1}{2}\|u - y\|_2^2 + \nu h(u). \tag{4}$$

Without further assumptions on h, (4) is a set that may be empty, or contain one or more elements. The iteration (3) has the following descent property

Lemma 1 (8, Lemma 2)**.** *Let ∇f be Lipschitz continuous with Lipschitz constant $L \geq 0$, h be proper lower semi-continuous and $\inf h > -\infty$. Let $x_k \in \operatorname{dom} h$, $0 < \nu < 1/L$, and x_{k+1} be defined according to (3). Then,*

$$(f + h)(x_{k+1}) \leq (f + h)(x_k) - \tfrac{1}{2}(\nu^{-1} - L)\|x_{k+1} - x_k\|_2^2. \tag{5}$$

3 Linear least squares

For fixed $\sigma \geq 0$ and $x \in \mathbb{R}^n$, define

$$\varphi(s;x) := \tfrac{1}{2}\|J(x)s + F(x)\|_2^2, \tag{6a}$$

$$\psi(s;x) \approx h(x+s) \quad \text{with} \quad \psi(0;x) = h(x), \tag{6b}$$

$$m(s;x,\sigma) := \varphi(s;x) + \tfrac{1}{2}\sigma\|s\|_2^2 + \psi(s;x). \tag{6c}$$

Consider the parametric problem and its optimal set

$$p(x,\sigma) := \min_s \; m(s;x,\sigma) \leq \varphi(0;x) + \psi(0;x) = f(x) + h(x) \tag{7a}$$

$$P(x,\sigma) := \operatorname*{argmin}_s \; m(s;x,\sigma). \tag{7b}$$

The form of (7) is representative of a Levenberg-Marquardt subproblem for (1) in which f and h are modeled separately.

In particular, $\varphi(0;x) = f(x)$ and $\nabla_s\varphi(0;x) = \nabla f(x)$. We make the following additional assumption. *Model Assumption* 3.1. For any $x \in \mathbb{R}^n$, $\psi(\cdot;x)$ is proper, lsc and prox-bounded, i.e., there exists $\lambda_x \in \mathbb{R}_+ \cup \{+\infty\}$ such that $\psi(\cdot;x) + \frac{1}{2}\lambda_x^{-1}\|\cdot\|_2^2$ is bounded below. In addition, $\psi(0;x) = h(x)$, and $\partial\psi(0;x) = \partial h(x)$.

In Model Assumption 3.1, we assume that our choice of λ_x is the supremum of all possible choices, and we refer to it as the *threshold of prox-boundedness* of $\psi(\cdot;x)$. In particular, $\psi(\cdot;x)$ is bounded below if and only if $\lambda_x = +\infty$.

By Proposition 1, if $\sigma \geq \lambda_x^{-1}$,

$$s \in P(x,\sigma) \quad \Longrightarrow \quad 0 \in \nabla\varphi(s;x) + \sigma s + \partial\psi(s;x).$$

We define

$$\xi(x,\sigma) := (f+h)(x) - p(x,\sigma). \tag{8}$$

The following stationarity criterion follows directly from the definitions above.
Lemma 2. *Let Model Assumption 3.1 be satisfied and $\sigma \geq \lambda_x^{-1}$. Then $\xi(x,\sigma) = 0 \iff 0 \in P(x,\sigma) \Longrightarrow x$ is first-order stationary for (1). In addition, x is first-order stationary for (1) if and only if $s = 0$ is first-order stationary for (6c).*

Proof. Note first that $\xi(x,\sigma) = 0 \iff p(x,\sigma) = (f+h)(x) = \varphi(0;x) + \psi(0;x)$, which occurs if and only if $0 \in P(x,\sigma)$. Proposition 1 then implies $0 \in \partial m(0;x,\sigma) = \nabla\varphi(0;x) + \partial\psi(0;x)$ and is equivalent to (2). □

The next result states some properties of (7).
Proposition 2. *Let Model Assumption 3.1 be satisfied.* $\operatorname{dom} p = \operatorname{dom} P = \operatorname{dom}\psi \times \{\sigma \mid \sigma \geq \lambda_x^{-1}\}$. *In addition, for any $x \in \mathbb{R}^n$,*

1. *$p(x,\cdot)$ is proper lsc and for each $\sigma > \lambda_x^{-1}$, $P(x,\sigma)$ is nonempty and compact;*
2. *if $\{\sigma_k\} \to \bar{\sigma} > \lambda_x^{-1}$ in such a way that $\{p(x,\sigma_k)\} \to p(x,\bar{\sigma})$, and for each k, $s_k \in P(x,\sigma_k)$, then $\{s_k\}$ is bounded and all its limit points are in $P(x,\bar{\sigma})$;*
3. *$p(x,\cdot)$ is continuous at any $\bar{\sigma} > \lambda_x^{-1}$ and $\{p(x,\sigma_k)\} \to p(x,\bar{\sigma})$ holds in part 2 if $\bar{\sigma} > 0$.*

Proof. Parts 1–2 follow from applying [25, Theorem 1.17] by noting that (6c) is level-bounded in s locally uniformly in (x,σ) because $\psi(\cdot;x) + \frac{1}{2}\lambda_x^{-1}\|s\|_2^2$ is bounded and $\varphi(s;x) + \frac{1}{2}(\sigma - \lambda_x^{-1})\|s\|_2^2$ is level bounded in s locally uniformy in (x,σ). Part 3 also follows from [25, Theorem 1.17] by noting that (6c) is continuous in σ at any $\bar{\sigma} > \lambda_x^{-1}$. □

By Proposition 2 part 3, $\xi(x, \cdot)$ is continuous at any $\bar{\sigma} > \lambda_x^{-1}$.

Although (6a) is a natural model of f about x, convergence properties may be stated in terms of the simpler first-order model

$$\varphi_1(s; x) := f(x) + \nabla f(x)^T s = \tfrac{1}{2}\|F(x)\|_2^2 + \left(J(x)^T F(x)\right)^T s, \tag{9a}$$
$$m_1(s; x, \sigma) := \varphi_1(s; x) + \tfrac{1}{2}\sigma\|s\|^2 + \psi(s; x). \tag{9b}$$

The first step of the proximal gradient method (3) applied to the minimization of both $\varphi(s; x)+\psi(s; x)$ and $\varphi_1(s; x) + \psi(s; x)$ with steplength $\nu > 0$ is

$$\begin{aligned} s_1 &\in \operatorname*{prox}_{\nu\psi(\cdot;x)} \left(-\nu J(x)^T F(x)\right) \\ &= \operatorname*{argmin}_s \ \tfrac{1}{2}\|s + \nu J(x)^T F(x)\|_2^2 + \nu\psi(s; x) \\ &= \operatorname*{argmin}_s \ \left(J(x)^T F(x)\right)^T s + \tfrac{1}{2}\nu^{-1}\|s\|_2^2 + \psi(s; x) \\ &= \operatorname*{argmin}_s \ m_1(s; x, \nu^{-1}). \end{aligned} \tag{10}$$

If $\nu^{-1} \geq \sigma$, then $m_1(s; x, \sigma) \leq m_1(s; x, \nu^{-1})$. Therefore, if s_1 results from (10), it also induces decrease in (9b).

In parallel to Lemma 2 and Proposition 2, we may define

$$p_1(x, \sigma) := \min_s \ m_1(s; x, \sigma) \leq \varphi_1(0; s) + \psi(0; x) = f(x) + h(x) \tag{11a}$$
$$P_1(x, \sigma) := \operatorname*{argmin}_s \ m_1(s; x, \sigma), \tag{11b}$$
$$\xi_1(x, \sigma) := (f + h)(x) - p_1(x, \sigma) \geq 0, \tag{11c}$$

and we have the following results, stating corresponding properties of p_1 and ξ_1. The proofs replicate those in Proposition 2 and Lemma 3.

Lemma 3. *Let Model Assumption 3.1 be satisfied and $\sigma \geq \lambda_x^{-1}$. Then $\xi_1(x, \sigma) = 0 \iff 0 \in P_1(x, \sigma) \implies x$ is first-order stationary for (1). In addition, x is first-order stationary for (1) if and only if $s = 0$ is first-order stationary for (9b).*

Proposition 3. *Let Model Assumption 3.1 be satisfied.* $\operatorname{dom} p_1 = \operatorname{dom} P_1 = \operatorname{dom} \psi \times \{\sigma \mid \sigma \geq \lambda_x^{-1}\}$. *In addition, for any $x \in \mathbb{R}^n$,*

1. *$p_1(x, \cdot)$ is proper lsc and for each $\sigma > \lambda_x^{-1}$, $P_1(x, \sigma)$ is nonempty and compact;*
2. *if $\{\sigma_k\} \to \bar{\sigma} > \lambda_x^{-1}$ in such a way that $\{p_1(x, \sigma_k)\} \to p_1(x, \bar{\sigma})$, and for each k, $s_k \in P_1(x, \sigma_k)$, then $\{s_k\}$ is bounded and all its limit points are in $P_1(x, \bar{\sigma})$;*
3. *$p_1(x, \cdot)$ is continuous at any $\bar{\sigma} > \lambda_x^{-1}$ and $\{p_1(x, \sigma_k)\} \to p_1(x, \bar{\sigma})$ holds in part 2 if $\bar{\sigma} > 0$.*

Because $L = 0$ for φ_1, Lemma 1 implies that the decrease achieved by s_1 is $(\varphi_1 + \psi)(s_1; x) \leq (\varphi_1 + \psi)(0; x) - \frac{1}{2}\nu^{-1}\|s_1\|^2$, which can be rearranged as

$$(f + h)(x) - (\varphi_1 + \psi)(s_1; x) \geq \tfrac{1}{2}\nu^{-1}\|s_1\|^2 \geq \tfrac{1}{2}\sigma\|s_1\|^2. \tag{12}$$

In the special case where $\psi = 0$, $s_1 = -\nu^{-1}\nabla f(x)$, so that (12) reduces to

$$\xi_1(x, \sigma) \geq \xi_1(x, \nu^{-1}) \geq f(x) - \varphi_1(s_1; x) \geq \tfrac{1}{2}\sigma\nu^{-1}\|\nabla f(x)\|^2 \geq \tfrac{1}{2}\sigma^2\|\nabla f(x)\|^2,$$

which suggests that $\sigma^{-1}(\xi_1(x, \nu^{-1}))^{1/2}$ may be used as stationarity measure.

4 Nonlinear least squares

4.1 A regularization approach

We first examine the formulation of the method of Levenberg and Marquardt in which the model (6c) is employed to compute a step. Specifically, consider Algorithm 1. The step s_k is computed by approximately minimizing (6c) in stage 7 but the quality of the step is measured without taking the regularization term $\frac{1}{2}\sigma_k\|s_k\|^2$ into account in stage 8. The subproblem step s_k may be computed by continuing the iterations of the proximal gradient method initialized at $s_{k,1}$. This gives rise to one possible implementation of Algorithm 1.

Algorithm 1 Nonsmooth regularized Levenberg-Marquardt method.

```
1: Choose constants $0 < \eta_1 \le \eta_2 < 1$ and $0 < \gamma_3 \le 1 < \gamma_1 \le \gamma_2$.
2: Choose $x_0 \in \mathbb{R}^n$ where $h$ is finite, $\sigma_0 > 0$, compute $F(x_0)$ and $h(x_0)$.
3: for $k = 0, 1, \dots$ do
4:     Choose a steplength $\nu_k < 1/(\|J(x_k)\|^2 + \sigma_k)$.
5:     Compute $s_{k,1}$ as defined in (10) and $\xi_1(x_k, \nu_k^{-1})$ as defined in (11c).
6:     Define $m(s; x_k, \sigma_k)$ as in (6c).
7:     Compute an approximate solution $s_k$ of (7b).
8:     Compute the ratio
           $$\rho_k := \frac{f(x_k) + h(x_k) - (f(x_k + s_k) + h(x_k + s_k))}{\varphi(0; x_k) + \psi(0; x_k) - (\varphi(s_k; x_k) + \psi(s_k; x_k))}.$$
9:     If $\rho_k \ge \eta_1$, set $x_{k+1} = x_k + s_k$. Otherwise, set $x_{k+1} = x_k$.
10:    Update the regularization parameter according to
           $$\sigma_{k+1} \in \begin{cases} [\gamma_3\sigma_k,\ \sigma_k] & \text{if } \rho_k \ge \eta_2, \\ [\sigma_k,\ \gamma_1\sigma_k] & \text{if } \eta_1 \le \rho_k < \eta_2, \\ [\gamma_1\sigma_k,\ \gamma_2\sigma_k] & \text{if } \rho_k < \eta_1. \end{cases}$$
11: end for
```

It may occur that $\sigma_k \le \lambda_{x_k}^{-1}$. In such a case, $\psi(s_k; x_k) = -\infty$ so that the rules of extended arithmetic imply $\rho_k = 0$, whether $h(x_k + s_k) = +\infty$ or is finite. Thus s_k will be rejected at stage 9 and σ_{k+1} will be chosen larger than σ_k at stage 10. After a finite number of such increases, σ_k will exceed $\lambda_{x_k}^{-1}$ and a step with finite $\psi(s_k; x_k)$ will result.

Our main working assumption is the following.

Problem Assumption 4.1. The residual F and its Jacobian J are bounded and Lipschitz continuous on $\Omega := \{x \in \mathbb{R}^n \mid (f + h)(x) \le (f + h)(x_0)\}$ and h is proper and lower semi-continuous.

While Problem Assumption 4.1 is a strong demand on all of $\mathbb{R}^n$ and, in particular, rules out the case of linear least squares, it is a common assumption in the convergence analysis of the Levenberg-Marquardt method. If Ω is a compact set, then F is Lipschitz continuous on Ω if it is $\mathcal{C}^1$ on Ω, and J is Lipschitz continuous on Ω if F is $\mathcal{C}^2$ on Ω.

Under Problem Assumption 4.1, ∇f is Lipschitz continuous on Ω, i.e., there exists $L > 0$ such that

$$|f(x + s) - (f(x) + \nabla f(x)^T s)| \le \tfrac{1}{2} L \|s\|_2^2 \quad \text{for all } x,\ x + s \in \Omega. \tag{13}$$

We emphasize that in what follows, knowledge of L, or an estimate thereof, is not required. Our next assumption on the model is the following.

Model Assumption 4.1. There exists a constant $\kappa_{\mathrm{m}} > 0$ such that for all x and $s \in \mathbb{R}^n$, $|(f + h)(x + s) - (\varphi + \psi)(s; x)| \le \kappa_{\mathrm{m}} \|s\|^2$.

Model Assumption 4.1 is essentially an assumption on the nonsmooth part ψ of the model. Indeed, (6a) and (13) combine to yield

$$|f(x + s) - \varphi(s; x)| \le |f(x + s) - (f(x) + \nabla f(x)^T s)| + \tfrac{1}{2}\|J(x)s\|^2|$$

$$\leq \tfrac{1}{2}(L + \|J(x)\|^2)\|s\|^2.$$

where we used the definition of $f(x)$, the identity $\nabla f(x) = J(x)^T F(x)$, and (13). Thus if J is bounded on Ω, we obtain

$$|f(x+s) - \varphi(s;x)| \leq \tfrac{1}{2}(L + \sup_{x \in \Omega} \|J(x)\|^2)\|s\|^2.$$

In particular, Model Assumption 4.1 is satisfied with $\kappa_{\mathrm{m}} = \frac{1}{2}(L + \sup_{x \in \Omega} \|J(x)\|^2)$ if we select $\psi(s;x) := h(x+s)$.

We make the following additional assumption and say that $\{\psi(\cdot;x_k)\}$ is *uniformly prox-bounded.*

Model Assumption 4.2. There exists $\lambda > 0$ such that $\lambda_{x_k} \geq \lambda$ for all $k \in \mathbb{N}$.

Model Assumption 4.2 is satisfied if h itself is prox-bounded and we select $\psi(s;x_k) := h(x_k + s)$ at each iteration.

Our first result ensures that σ_k is bounded above in Algorithm 1.

Theorem 1. *Let Problem Assumption 4.1 and Model Assumptions 3.1, 4.1 and 4.2 be satisfied, and let*

$$\sigma_{\mathrm{succ}} := \max(2\kappa_{\mathrm{m}}/(1-\eta_2),\, \lambda^{-1}) > 0. \tag{14}$$

If x_k is not first-order stationary and $\sigma_k \geq \sigma_{\mathrm{succ}}$, then iteration k is very successful and $\sigma_{k+1} \leq \sigma_k$.

Proof. Let s_k be the step computed at iteration k of Algorithm 1. If $\sigma_k < \lambda_{x_k}^{-1}$, $\rho_k = 0$ as explained above, s_k is rejected and σ_k is increased. Hence, we assume that $\sigma_k \geq \lambda^{-1} \geq \lambda_{x_k}^{-1}$. Because x_k is not first-order stationary, $s_k \neq 0$. Because s_k is an approximate solution of (7b), we must have

$$\varphi(0;x_k) + \psi(0;x_k) \geq \varphi(s_k;x_k) + \tfrac{1}{2}\sigma_k\|s_k\|^2 + \psi(s_k;x_k)$$

and therefore,

$$\varphi(0;x_k) + \psi(0;x_k) - (\varphi(s_k;x_k) + \psi(s_k;x_k)) \geq \tfrac{1}{2}\sigma_k\|s_k\|^2. \tag{15}$$

Model Assumption 4.1 and (15) combine to yield

$$|\rho_k - 1| = \frac{|f(x_k+s_k) + h(x_k+s_k) - (\varphi(s_k;x_k) + \psi(s_k;x_k))|}{\varphi(0;x_k) + \psi(0;x_k) - (\varphi(s_k;x_k) + \psi(s_k;x_k))} \leq \frac{2\kappa_{\mathrm{m}}\|s_k\|^2}{\sigma_k\|s_k\|^2}.$$

After simplifying by $\|s_k\|^2$, we obtain $\sigma_k \geq \sigma_{\mathrm{succ}} \implies \rho_k \geq \eta_2$. □

Note that Theorem 1 does not explicitly include Problem Assumption 4.1 in its assumptions, though it is likely to be required for Model Assumption 4.1 to hold.

Interestingly, Theorem 1 holds without assuming that the step s_k satisfies a sufficient decrease condition. Upon examination of the proof, the reason turns out to be that any step that results in simple decrease in $m(s;\sigma,x)$ results in sufficient decrease in $\varphi(\cdot;x) + \psi(\cdot;x)$, independently of the method used to compute s_k.

Theorem 1 ensures existence of a constant $\sigma_{\max} > 0$ such that

$$\sigma_k \leq \sigma_{\max} := \min(\sigma_0, \gamma_2\sigma_{\mathrm{succ}}) > 0 \quad \text{for all } k \in \mathbb{N}. \tag{16}$$

Our next result concerns the situation where a finite number of successful iterations occur. The proof is almost identical to that of [13, Theorem 6.4.4] and [1, Theorem 3.5] and is omitted.

Theorem 2. *Let Problem Assumption 4.1 and Model Assumptions 3.1 and 4.1 be satisfied. If Algorithm 1 only generates finitely many successful iterations, then $x_k = x^*$ for all sufficiently large k and x^* is first-order critical.*

By Rockafellar and Wets [25, Theorem 1.25], $p_1(x, \sigma)$ increases when σ increases, and thus, $\xi_1(x, \sigma)$ decreases when σ increases. Thus, it follows from (16) that

$$\xi_1(x_k, \sigma_k) \geq \xi_1(x_k, \sigma_{\max}) \quad \text{for all } k \in \mathbb{N}. \tag{17}$$

Lemma 2, (17) and the remarks at the end of section 3 suggest using $\xi_1(x_k, \sigma_{\max})^{\frac{1}{2}}$ as stationarity measure. Indeed, for given $\epsilon > 0$, $\xi_1(x_k, \sigma_{\max}) \leq \epsilon/\sigma_{\max} \Longrightarrow \sigma_k \xi_1(x_k, \sigma_{\max}) \leq \epsilon$.

Because we must choose the steplength ν_k as in Step 4 of Algorithm 1, we compute $\xi_1(x_k, \nu_k^{-1})$ rather than $\xi_1(x_k, \sigma_k)$. Concretely, for given $0 < \theta < 1$, we set

$$\nu_k := \theta / (\|J_k\|^2 + \sigma_k). \tag{18}$$

Under Problem Assumption 4.1, there exists $\kappa_J > 0$ such that $\|J(x)\| \leq \kappa_J$ for all $x \in \Omega$. Because Algorithm 1 only generates $x_k \in \Omega$, the above and (16) yield

$$\nu_k \geq \theta / (\kappa_J^2 + \sigma_{\max}) := \nu_{\min} > 0 \quad \text{for all } k \in \mathbb{N}. \tag{19}$$

Therefore, $\nu_k^{-1} \leq \nu_{\min}^{-1}$ for all $k \geq 0$, and

$$\xi_1(x_k, \nu_k^{-1}) \geq \xi_1(x_k, \nu_{\min}^{-1}) \quad \text{for all } k \in \mathbb{N}. \tag{20}$$

For a stopping tolerance $\epsilon \in (0, 1)$, we seek to determine $k(\epsilon) \in \mathbb{N}$ such that

$$\xi_1(x_k, \nu_{\min}^{-1})^{\frac{1}{2}} > \epsilon \quad \text{for all } k < k(\epsilon) \quad \text{and} \quad \xi_1(x_{k(\epsilon)}, \nu_{\min}^{-1})^{\frac{1}{2}} \leq \epsilon. \tag{21}$$

Define the sets

$$\mathcal{S} := \{k \in \mathbb{N} \mid \rho_k \geq \eta_1\}, \tag{22a}$$
$$\mathcal{S}(\epsilon) := \{k \in \mathcal{S} \mid k < k(\epsilon)\}, \tag{22b}$$
$$\mathcal{U}(\epsilon) := \{k \in \mathbb{N} \mid k \notin \mathcal{S} \text{ and } k < k(\epsilon)\}. \tag{22c}$$

In order to conduct the complexity analysis, it is necessary to assume that the step computation at stage 7 of Algorithm 1 is related to $\xi_1(x_k, \sigma_k)$. We make the following assumption.

Step Assumption 4.1. There exists $\kappa_{\text{mdc}} \in (0, 1)$ such that s_k computed at stage 7 of Algorithm 1 satisfies

$$\varphi(0; x_k) + \psi(0; x_k) - (\varphi(s_k; x_k) + \psi(s_k; x_k)) \geq \kappa_{\text{mdc}} \xi_1(x_k, \nu_k^{-1}). \tag{23}$$

Step Assumption 4.1 is similar to sufficient decrease conditions used in trust-region methods—see [13]. Aravkin et al. [1] provide a concrete use of such condition in a trust-region method for nonsmooth regularized optimization. Clearly, the sufficient decrease assumption is satisfied after a single step of the proximal gradient method applied to (6c). Hence, it is also satisfied at a minimizer of (6c). Thus, in step 7 of Algorithm 1, one strategy is to continue the proximal-gradient iterations until a stopping condition is attained.

The following results parallel those of Aravkin et al. [1], which are in turn inspired from those of Cartis et al. [11] and references therein.

Lemma 4. *Let Problem Assumption 4.1 and Model Assumptions 3.1 and 4.1 be satisfied and s_k be computed according to Step Assumption 4.1, where ν_k is chosen according to (18). Assume there are infinitely many successful iterations and that $f(x) + h(x) \geq (f + h)_{\text{low}}$ for all $x \in \mathbb{R}^n$. Then, for all $\epsilon \in (0, 1)$,*

$$|\mathcal{S}(\epsilon)| \leq \frac{(f + h)(x_0) - (f + h)_{\text{low}}}{\eta_1 \kappa_{\text{mdc}} \epsilon^2} = O(\epsilon^{-2}). \tag{24}$$

Proof. For $k \in \mathcal{S}(\epsilon)$, Step Assumption 4.1 and (20) imply

$$\begin{aligned}(f+h)(x_k) - (f+h)(x_k+s_k) &\geq \eta_1(\varphi(0;x_k)+\psi(0;x_k)-(\varphi(s_k;x_k)+\psi(s_k;x_k)))\\ &\geq \eta_1\kappa_{\mathrm{mdc}}\xi_1(x_k,\nu_k^{-1})\\ &\geq \eta_1\kappa_{\mathrm{mdc}}\xi_1(x_k,\nu_{\min}^{-1})\\ &\geq \eta_1\kappa_{\mathrm{mdc}}\epsilon^2.\end{aligned}$$

The rest of the proof mirrors that of [1, Lemma 3.6]. □

Lemma 5. *Under the assumptions of Lemma 4,*

$$|\mathcal{U}(\epsilon)| \leq \frac{\log(\sigma_{\max}/\sigma_0)}{\log(\gamma_1)} + |\mathcal{S}(\epsilon)|\frac{|\log(\gamma_3)|}{\log(\gamma_1)} = O(\epsilon^{-2}). \tag{25}$$

Proof. For each $k \in \mathcal{U}(\epsilon)$, $\sigma_{k+1} \geq \gamma_1\sigma_k$, while for each $k \in \mathcal{S}(\epsilon)$, $\sigma_{k+1} \geq \gamma_3\sigma_k$. Thus if $k(\epsilon)$ is the iteration for which (21) occurs for the first time,

$$\sigma_0\gamma_1^{|\mathcal{U}(\epsilon)|}\gamma_3^{|\mathcal{S}(\epsilon)|} \leq \sigma_{k(\epsilon)-1} \leq \sigma_{\max}.$$

Taking logarithms, we have

$$|\mathcal{U}(\epsilon)|\log(\gamma_1) + |\mathcal{S}(\epsilon)|\log(\gamma_3) \leq \log(\sigma_{\max}/\sigma_0).$$

Rearranging and recalling that $0 < \gamma_3 < 1$ yields (25). □

Combining Lemmas 4 and 5 yields the overall iteration complexity bound.

Theorem 3. *Under the assumptions of Lemma 4,*

$$|\mathcal{S}(\epsilon)| + |\mathcal{U}(\epsilon)| = O(\epsilon^{-2}). \tag{26}$$

Stated differently, Theorem 3 ensures that either $(f+h)(x_k) \to -\infty$ or that $\liminf_{k\to\infty}\xi_1(x_k,\nu_{\min}^{-1}) = 0$.

4.2 A trust-region approach

We now apply Algorithm 3.1 of Aravkin et al. [1] to (1). We assume that each $f_i : \mathbb{R}^n \to \mathbb{R}$ is $\mathcal{C}^1$, so that their Problem Assumption 3.1 is satisfied. A natural model for f about x is the Gauss-Newton model (6a), which satisfies $\varphi(0;x) = f(x)$ and $\nabla_s\varphi(0;x) = \nabla f(x) = J(x)^T F(x)$. The model $\psi(s;x)$ of $h(x+s)$ is required to satisfy the same Model Assumption 4.1, which holds provided ∇f is Lipschitz continuous or each f_i is $\mathcal{C}^2$ with bounded Hessian. In Aravkin et al. [1, Algorithm 3.1], the first proximal gradient step s_1 is computed by solving

$$\begin{aligned}&\underset{s}{\text{minimize}}\ \tfrac{1}{2}\|F(x)\|_2^2 + (J(x)^T F(x))^T s + \tfrac{1}{2}\nu^{-1}\|s\|^2 + \psi(s;x)\\ &\text{subject to } \|s\| \leq \Delta,\end{aligned} \tag{27}$$

i.e.,

$$s_1 \in \underset{\nu\psi(\cdot;x)+\chi(\cdot|\Delta\mathbb{B})}{\text{prox}}(-\nu J(x)^T F(x)),$$

where $0 < \nu < 1/(\|J(x)\|^2 + \alpha^{-1}\Delta^{-1})$ for a preset constant $\alpha > 0$. Subsequent steps continue the proximal gradient iterations to compute an approximate solution of

$$\underset{s}{\text{minimize}}\ \tfrac{1}{2}\|J(x)s + F(x)\|_2^2 + \psi(s;x) \quad \text{subject to } \|s\| \leq \min(\beta\|s_1\|,\ \Delta), \tag{28}$$

where $\beta \geq 1$. The above describes a trust-region variant of the method of Levenberg [19] and Marquardt [21] for regularized nonlinear least-squares problems. The assumption that $\psi(\cdot; x)$ is prox-bounded can be removed because $\psi(\cdot; x) + \chi(\cdot \mid \Delta\mathbb{B})$ is always bounded below, hence prox-bounded with $\lambda_x = \infty$. An approximate solution of (28) must satisfy Step Assumption 4.1 with $\xi_1(x, \sigma)$ replaced with

$$\hat{\xi}_1(\Delta; x, \nu) := f(x) + h(x) - \hat{p}_1(\Delta; x, \nu),$$

where $\hat{p}_1(\Delta; x, \nu)$ is the optimal value of (27).

Under the above assumptions, Aravkin et al. establish that the trust-region radius Δ never drops below the threshold

$$\Delta_{\text{min}} := \min\left(\Delta_0,\, \hat{\gamma}_1 \frac{\kappa_{\text{mdc}}(1-\eta_2)}{2\kappa_{\text{m}}\alpha\beta^2}\right),$$

where $\Delta_0 > 0$ is the initial trust-region radius, $\hat{\gamma}_1 \in (0, 1)$ is the fraction by which Δ is reduced on rejected steps, $\eta_2 \in (0, 1)$ is the threshold above which Δ is increased on accepted steps, and κ_{mdc} and κ_{m} play similar roles as the constants of the same name in Model Assumption 4.1 and Step Assumption 4.1.

Aravkin et al. use $\hat{\xi}_1(\Delta_{\text{min}}; x, \nu)$ as stationarity measure. They show that for any $\epsilon \in (0, 1)$, the number of iterations necessary to achieve

$$\hat{\xi}_1(\Delta_{\text{min}}; x, \nu)^{\frac{1}{2}} \leq \epsilon$$

is $O(\epsilon^{-2})$ provided that $f + h$ is bounded below. We refer the reader to [1] for complete details.

5 Proximal operators

In Algorithm 1 or the algorithm of Section 4.2, a typical model of the nonsmooth term h is $\psi(s; x) := h(x+s)$. If those algorithms are to use Aravkin et al.'s quadratic regularization method [1, Algorithm 6.1] to compute a step, the latter will in turn form a model of $\psi(\cdot; x)$ at each iteration. In order to simplify notation, let $\psi_k(s) := \psi(s; x_k) = h(x_k + s)$ be the model used at iteration k of Algorithm 1 or the algorithm of Section 4.2.

5.1 General proximal operators

In Algorithm 1, the nonsmooth term in the objective of the subproblem is $\psi_k(s)$. The typical model about s_j reduces to $\omega_j(t) = \psi_k(s_j + t) = h(x_k + s_j + t)$ and, instead of (30), the step computed is

$$t_j \in \underset{t}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}\|t - q\|^2 + h(x_k + s_j + t). \tag{29}$$

The same change of variable as above yields

$$v_j \in \underset{v}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}\|v - \bar{q}\|^2 + h(v) = \underset{\nu h}{\operatorname{prox}}(\bar{q}),$$

whether h is separable or not. Thus we obtain

$$t_j \in \underset{\nu h}{\operatorname{prox}}(\bar{q}) - (x_k + s_j).$$

The nonsmooth term in the objective of the subproblem of the algorithm of Section 4.2 is $\psi_k(s) + \chi(s; \Delta_k)$. About iterate s_j of [1, Algorithm 6.1], the user supplies a model $\omega_j(t) := \omega(t; s_j) \approx \psi_k(s_j + t) + \chi(s_j + t \mid \Delta_k\mathbb{B})$, and the typical choice is $\omega_j(t) = \psi_k(s_j + t) + \chi(s_j + t \mid \Delta_k\mathbb{B}) = h(x_k + s_j + t) + \chi(s_j + t \mid \Delta_k\mathbb{B})$. The step computed is $t_j \in \operatorname{prox}_{\nu\omega_j}(q)$ for certain fixed $\nu > 0$ and $q \in \mathbb{R}^n$, i.e.,

$$t_j \in \underset{t}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}\|t - q\|^2 + h(x_k + s_j + t) + \chi(s_j + t \mid \Delta_k\mathbb{B}). \tag{30}$$

The change of variables $v := x_k + s_j + t$ allows us to rewrite (30) as

$$v_j \in \underset{v}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}\|v - \bar{q}\|^2 + h(v) + \chi(v - x_k \mid \Delta_k \mathbb{B}), \tag{31}$$

where $\bar{q} := x_k + s_j + q$, from which we recover $t_j = v_j - (x_k + s_j)$.

5.2 Separable shifted proximal operators

If h is separable and the trust region is defined by the ℓ_∞-norm, the problem decomposes and the i-th component of v_j is

$$\begin{aligned} v_{j,i} &\in \underset{v_i}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}(v_i - \bar{q}_i)^2 + h_i(v_i) + \chi(v_i - x_{k,i} \mid [-\Delta_k, \Delta_k]) \\ &= \underset{v_i}{\operatorname{argmin}}\ \tfrac{1}{2}\nu^{-1}(v_i - \bar{q}_i)^2 + h_i(v_i) + \chi(v_i \mid [x_{k,i} - \Delta_k, x_{k,i} + \Delta_k]). \end{aligned} \tag{32}$$

Two situations may occur. In the first situation, $x_{k,i} - \Delta_k < v_{j,i} < x_{k,i} + \Delta_k$, so that $v_{j,i} \in \operatorname{prox}_{\nu h_i}(\bar{q}_i)$, i.e.,

$$t_{j,i} \in \underset{\nu h_i}{\operatorname{prox}}(\bar{q}_i) - (x_{k,i} + s_{j,i}).$$

In the second situation, at least one unconstrained solution lies outside of $[x_{k,i} - \Delta_k, x_{k,i} + \Delta_k]$, so that constrained global minima of (32) are either one or both bounds, and/or unconstrained local minima that lie between the bounds.

When h is convex, the constrained solution is the feasible point nearest the unique unconstrained global solution, i.e.,

$$v_{j,i} \in \underset{[x_{k,i} - \Delta_k, x_{k,i} + \Delta_k]}{\operatorname{proj}}(\underset{\nu h_i}{\operatorname{prox}}(\bar{q}_i)),$$

i.e.,

$$t_{j,i} \in \underset{[x_{k,i} - \Delta_k, x_{k,i} + \Delta_k]}{\operatorname{proj}}(\underset{\nu h_i}{\operatorname{prox}}(\bar{q}_i)) - (x_{k,i} + s_{j,i}).$$

Example 1 **($\ell_{1/2}^{1/2}$ pseudonorm).** Consider $\psi(s) = \|s\|_{1/2}^{1/2} = \sum_j |s_j|^{1/2}$. When the trust-region bounds are inactive, Cao et al. [10] express the solution of (32) as

$$v_{j,i} = \begin{cases} \tfrac{2}{3}|\bar{q}_i| \left(1 + \cos\left(\tfrac{2}{3}\pi - \tfrac{2}{3}\mu_\lambda(\bar{q}_i)\right)\right) & \bar{q}_i \ > p(\lambda) \\ 0 & |\bar{q}_i| \le p(\lambda) \\ -\tfrac{2}{3}|\bar{q}_i| \left(1 + \cos\left(\tfrac{2}{3}\pi - \tfrac{2}{3}\mu_\lambda(\bar{q}_i)\right)\right) & \bar{q}_i \ < -p(\lambda) \end{cases}$$

where

$$\mu_\lambda(\bar{q}_i) := \arccos\left(\frac{\lambda}{4}\left(\frac{|\bar{q}_i|}{3}\right)^{-3/2}\right), \qquad p(\lambda) := \frac{54^{1/3}}{4}(2\lambda)^{2/3}.$$

When the trust-region constraint is active, Cao et al. [10] state that the above yields the inflection points of (32). We simply check the inflection points as well as the bounds. If the inflection points are within the bounds, we choose the minimum; if not, we select the minimum value of the cost function at the bounds.

5.3 Nonseparable shifted proximal operators for convex h

In this section we consider examples of nonseparable shifted proximal operators. The starting point is (31) where we assume that h is closed, proper, and convex. We rewrite

$$\chi(v - x \mid \Delta\mathbb{B}) = \sup_z \langle v - x, z \rangle - \sigma_{\Delta\mathbb{B}}(z),$$

where we write x and Δ instead of x_k and Δ_k for simplicity, and where the support function

$$\sigma_{\Delta\mathbb{B}}(z) := \sup_d \langle d, z\rangle + \chi(d \mid \Delta\mathbb{B}).$$

We substitute into (31) and obtain the saddle point problem

$$\min_v \sup_z \tfrac{1}{2}\nu^{-1}\|v - \bar{q}\|^2 + h(v) + \langle v - x, z\rangle - \sigma_{\Delta\mathbb{B}}(z). \tag{33}$$

The objective of (33) is convex in v and concave in z. The saddle-point conditions can be written

$$\begin{aligned} &0 \in \nu^{-1}(v - \bar{q}) + \partial h(v) + z = \nu^{-1}(v - (\bar{q} - \nu z)) + \partial h(v) \\ &0 \in v - x - \partial\sigma_{\Delta\mathbb{B}}(z). \end{aligned}$$

The first condition implies that $v \in \operatorname{prox}_{\nu h}(\bar{q} - \nu z)$. By convexity of h, v is unique so that we are left with

$$0 \in v - x - \partial\sigma_{\Delta\mathbb{B}}(z), \quad \text{where} \quad \operatorname*{prox}_{\nu h}(\bar{q} - \nu z) = \{v\}. \tag{34}$$

5.3.1 Special case: ℓ_2-norm

For $h(\cdot) := \lambda\|\cdot\|_2$,

$$\operatorname*{prox}_{\nu\lambda\|\cdot\|_2}(y) = \begin{cases} 0 & \text{if } \|y\| \le \nu\lambda \\ \left(1 - \frac{\nu\lambda}{\|y\|_2}\right) y & \text{if } \|y\| > \nu\lambda \end{cases}. \tag{35}$$

We now show how to solve (31) by converting (34) to a scalar root finding problem. For given z, let

$$\zeta = \zeta(z) := \|\bar{q} - \nu z\|_2.$$

There are two possibilities.

Case A: If $\zeta \le \nu\lambda$, (35) yields

$$\operatorname*{prox}_{\nu\lambda\|\cdot\|_2}(\bar{q} - \nu z) = \{v\} = \{0\}.$$

The optimal value of (31) in this case is $\frac{1}{2}\nu^{-1}\|\bar{q}\|^2$.

Case B: If $\zeta > \nu\lambda$, (35) yields

$$\operatorname*{prox}_{\nu\lambda\|\cdot\|_2}(\bar{q} - \nu z) = \{v\} = \left\{\left(1 - \frac{\nu\lambda}{\zeta}\right)(\bar{q} - \nu z)\right\}, \tag{36}$$

and (34) becomes

$$\begin{aligned} 0 &\in x - \left(1 - \frac{\nu\lambda}{\zeta}\right)(\bar{q} - \nu z) + \partial\sigma_{\Delta\mathbb{B}}(z) \\ &= (\zeta - \nu\lambda)\frac{\nu}{\zeta}\left(z - \left(\frac{1}{\nu}\bar{q} - \frac{\zeta}{\nu(\zeta - \nu\lambda)}x\right)\right) + \partial\sigma_{\Delta\mathbb{B}}(z), \end{aligned}$$

which we interpret as

$$z = z(\zeta) := \operatorname*{prox}_{\frac{\zeta}{\nu(\zeta-\nu\lambda)}\sigma_{\Delta\mathbb{B}}}\left(\frac{1}{\nu}\bar{q} - \frac{\zeta}{\nu(\zeta - \nu\lambda)}x\right). \tag{37}$$

Recall that [6, Theorem 6.46]

$$\operatorname*{prox}_{\alpha\sigma_{\Delta\mathbb{B}}}(y) = y - \alpha \operatorname*{proj}_{\Delta\mathbb{B}}(\alpha^{-1}y), \quad (\alpha > 0). \tag{38}$$

Therefore, the projection into $\Delta\mathbb{B}$ must be computable. In our implementation, we use $\mathbb{B} = \mathbb{B}_\infty$.

We may now search for ζ such that

$$g(\zeta) := \zeta - \|\bar{q} - \nu z(\zeta)\|_2 = 0. \tag{39}$$

Because projections into convex sets are Lipschitz continuous, so is g over $(\nu\lambda, +\infty)$.

Since (31) is strongly convex, there is a unique solution, and so g has at most one root such that $\zeta > \nu\lambda$. Any such root of g yields v given by (36) and $z(\zeta)$ given by (37) that jointly satisfy (34). If g has no such root, the Case A must occur.

The combination of (37) and (38) yields

$$\bar{q} - \nu z(\zeta) = \frac{\zeta}{\zeta - \nu\lambda}\left[x + \operatorname*{proj}_{\Delta\mathbb{B}}\left(\frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x\right)\right]. \tag{40}$$

As $\zeta \uparrow \infty$, $(\zeta - \nu\lambda)/\zeta \uparrow 1$, and by continuity, the term between square brackets in (40) converges to $x + \operatorname{proj}_{\Delta\mathbb{B}}(\bar{q} - x)$. Therefore, $\|\bar{q} - \nu z(\zeta)\|_2 \to \|x + \operatorname{proj}_{\Delta\mathbb{B}}(\bar{q} - x)\|_2$ and for sufficiently large ζ, we must have $g(\zeta) > 0$.

To study $g(\zeta)$ as $\zeta \downarrow \nu\lambda$, we consider several mutually-exclusive cases.

1. If $x \notin \Delta\mathbb{B}$, then, $\operatorname{proj}_{\Delta\mathbb{B}}(-x) \neq -x$. As $\zeta \downarrow \nu\lambda$, $(\zeta - \nu\lambda)/\zeta \downarrow 0$, and by continuity, the term between square brackets converges to $x + \operatorname{proj}_{\Delta\mathbb{B}}(-x) \neq 0$. Therefore, $\|\bar{q} - \nu z(\zeta)\|_2 \to \infty$ and for sufficiently small ζ, we must have $g(\zeta) < 0$.
2. Consider next the case where $x \in \operatorname{int}\Delta\mathbb{B}$. For ζ sufficiently close to $\nu\lambda$,

$$\operatorname*{proj}_{\Delta\mathbb{B}}\left(\frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x\right) = \frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x, \tag{41}$$

   and $\bar{q} - \nu z(\zeta) = \bar{q}$, i.e., $z(\zeta) = 0$. In this case,

   (a) if $\|\bar{q}\|_2 > \nu\lambda$, then $g(\zeta) < 0$ for ζ close enough to $\nu\lambda$,

   (b) if $\|\bar{q}\|_2 \le \nu\lambda$, then $g(\zeta) > 0$ for all $\zeta > \nu\lambda$;

3. If $\|x\|_\infty = \Delta$ and $\operatorname{proj}_{\Delta\mathbb{B}}(\bar{q} - x) = -x$, then $\operatorname{proj}_{\Delta\mathbb{B}}(\alpha\bar{q} - x) = -x$ for any $\alpha > 0$. In this case, the term between square brackets in (40) is always zero, and $\bar{q} - \nu z(\zeta) = 0$. Thus for all $\zeta > \nu\lambda$, $g(\zeta) = \zeta > 0$.
4. If $\|x\|_\infty = \Delta$ but $\operatorname{proj}_{\Delta\mathbb{B}}(\bar{q} - x) \neq -x$, there are two possible situations. Either the ray $\alpha\bar{q} - x$ intersects $\operatorname{int}\Delta\mathbb{B}$, or it does not. If it does, (41) occurs for all ζ sufficiently close to $\nu\lambda$, $\bar{q} - \nu z(\zeta) = \bar{q}$, and cases 2a–2b apply. If it does not, we have from Lipschitz continuity that

$$\left\|x + \operatorname*{proj}_{\Delta\mathbb{B}}\left(\frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x\right)\right\|_2 = \left\|\operatorname*{proj}_{\Delta\mathbb{B}}\left(\frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x\right) - \operatorname*{proj}_{\Delta\mathbb{B}}(-x)\right\|_2 \le \frac{\zeta - \nu\lambda}{\zeta}\|\bar{q}\|_2.$$

   Thus, $\|\bar{q} - \nu z(\zeta)\|_2 \le \|\bar{q}\|_2$, and

   (a) if $\|\bar{q}\|_2 > \nu\lambda$, then $g(\zeta) \ge \zeta - \|\bar{q}\|_2 > 0$ for $\zeta > \|\bar{q}\|_2$, and so there may exist a root in $(\nu\lambda, \|\bar{q}\|_2]$. By (40), and the fact that $\|y\|_2 \le \sqrt{n}\|y\|_\infty$ for all y, we also have

$$\|\bar{q} \;\, - \nu z(\zeta)\|_2 \le \frac{\zeta}{\zeta - \nu\lambda}\left(\|x\|_2 + \left\|\operatorname*{proj}_{\Delta\mathbb{B}}\left(\frac{\zeta - \nu\lambda}{\zeta}\bar{q} - x\right)\right\|_2\right) \le \frac{(\|x\|_2 + \Delta\sqrt{n})\zeta}{\zeta - \nu\lambda},$$

   so that $g(\zeta) > 0$ for $\zeta > \nu\lambda + 2\Delta\sqrt{n}$. Thus, the search interval may potentially be reduced to $(\nu\lambda, \min(\nu\lambda + \|x\|_2 + \Delta\sqrt{n}, \|\bar{q}\|_2)]$.

   (b) if $\|\bar{q}\| \le \nu\lambda$, then $g(\zeta) > 0$ for all $\zeta > \nu\lambda$.

Thus, in cases 1 and 2a, a root is guaranteed to exist in $(\nu\lambda, +\infty)$ and can be found by a bisection method. The upper bound may be found by observing that (40) implies

$$\|\bar{q} - \nu z(\zeta)\| \leq \frac{\zeta}{\zeta - \nu}(\|x\| + \Delta),$$

so that

$$g(\zeta) = \zeta - \|\bar{q} - \nu z(\zeta)\| \geq \zeta - \frac{\zeta}{\zeta - \nu\lambda}(\|x\| + \Delta),$$

and $g(\zeta) > 0$ as soon as $\zeta > \|x\| + \Delta + \nu\lambda$.

In case 1, a lower bound follows by applying the reverse triangle inequality to (40):

$$\|\bar{q} - \nu z(\zeta)\| \geq \frac{\zeta}{\zeta - \nu\lambda}(\|x\| - \Delta),$$

so that $g(\zeta) < 0$ as soon as $\zeta < \nu\lambda + \|x\| - \Delta$.

In case 2a, the lower bound is simply $\|\bar{q}\|$.

In cases 2b, 3 and 4b, there can be no root in $(\nu\lambda, +\infty)$ and Case A must occur.

Only case 4a requires a root search, with or without sign change. If no root exists in the search interval, Case A must occur.

5.3.2 Special case: Group lasso

The group lasso penalty is a sum of ℓ_2-norms of subvectors:

$$R_g(x) = \sum_i \|x_{[i]}\|_2,$$

where the $x_{[i]}$ partition x into non-overlapping groups. The proximal operator of R_g consists in applying (35) to each subvector:

$$\operatorname*{prox}_{\lambda R_g}(z)_{[i]} = \left(1 - \frac{\lambda}{\|z_{[i]}\|_2}\right)_+ z_{[i]}. \tag{42}$$

Thus, the strategy of the previous section may be applied to each group.

6 Implementation and numerical experiments

Our implementation of Algorithm 3.1 of [1] and Algorithm 1 for (1) employs Aravkin, Baraldi, and Orban's quadratic regularization method, named `R2`, to compute a step. `R2` may be viewed as an implementation of the proximal gradient method with adaptive step size. The trust-region variant uses $\Delta_0 = 1$, terminates the outer iterations as soon as $\xi(\Delta_k; x_k, \nu_k)^{1/2} < \epsilon_a + \epsilon_r \, \xi_{1,0}^{1/2}$, where $\epsilon_a > 0$ and $\epsilon_r > 0$ are an absolute and a relative tolerance, and $\xi_{1,0}$ is the value of ξ_1 observed at the first iteration. A round of inner iterations terminates as soon as

$$\hat{\xi}_1(x_k + s, \hat{\sigma}_k) \leq \begin{cases} 10^{-1} & \text{if } k = 0, \\ \max(\epsilon, \min(10^{-1}, \xi_1(x_k, \sigma_k)/10)) & \text{if } k > 0, \end{cases} \tag{43}$$

where $\hat{\sigma}_k$ and $\hat{\xi}_1$ are the regularization parameter and first-order stationarity measure used inside `R2`. In Algorithm 1, we use $\sigma_0 = 0.01$, and we terminate the outer iterations as soon as $\xi_1(x_k, \sigma_k)^{1/2} < \epsilon$ for a tolerance $\epsilon > 0$ because $\sigma_{\max}$ is unknown. The inner iterations stop in the same manner as (43). All algorithms are implemented in the Julia language [7] version 1.8 as part of the RegularizedOptimization.jl package [3]. The shifted proximal operators are implemented in the ShiftedProximalOperators.jl package [5], while test problems are in the RegularizedProblems.jl package [4]. By contrast with the numerical results of Aravkin et al. [1], test cases are explicitly implemented as nonlinear least-squares problems, with access to the residual $F(x)$ and its Jacobian, and not simply the gradient of $f(x) := \frac{1}{2}\|F(x)\|_2^2$. Jacobian-vector and transposed-Jacobian-vector products are either implemented manually or computed via forward [24] and reverse [23] automatic differentiation, respectively.

We perform comparisons with `R2` and with the quasi-Newton trust-region method of Aravkin et al. [1], named `TR`, and which does not exploit the structure of (1). The trust region is defined in ℓ_∞-norm and the quadratic model uses a limited-memory SR1 Hessian approximation with memory 5. In all experiments, we use $\psi(s; x) := h(x + s)$.

A direct comparison between the four methods is difficult because `LM` and `LMTR` do not utilize the same gradient; they instead take Jacobian-vector and transposed-Jacobian-vector products. To provide a meaningful comparison, in the tables below, we state: 1) the number of objective (or residual) evaluations; 2) the number of gradient evaluations (for `R2` and `TR`) ; 3) the number of transposed-Jacobian-vector products (for `LM` and `LMTR`), listed under gradient evaluations; 4) the solve time in seconds. Our rationale is as follows. `LM` and `LMTR` pass a model to `R2` whose objective evaluation requires one Jv, and whose gradient uses a Jv and a J^Tv. Note however that the latter Jv can be cached and reused. Thus, `R2` requires one Jv at each iteration, and additionally one J^Tv at each successful iteration.

In the figures, we plot descent as a function of residual/objective evaluations.

The summary of the numerical results below is that exploiting the least-squares structure results in a large reduction in outer iterations. However, solving the subproblem with a first-order method such as `R2` consumes many J^Tv. Our experiments thus highlight the need for more sophisticated subproblem solvers dedicated to (6c) and (28).

6.1 Group LASSO

In the group-LASSO problem, we observe noisy data from a linear system $b = Ax_T + \varepsilon$, where $A \in \mathbb{R}^{m\times n}$ has orthonormal rows, and x_T is segmented into g groups with every element in that group set to one of $\{-1, 0, 1\}$. The group-LASSO problem is given by

$$\min_x \tfrac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_{1,2}, \tag{44}$$

where $h(x) = \|x\|_{1,2} = \sum_{i=1}^{g} \|x_{[i]}\|_2$, i.e., the sum of the ℓ_2-norm of the groups. The groups consisting of all zeros are labeled as “inactive”, whereas the groups set to ±1 are “active”. We let $m = 512$, $n = 200$ and $\lambda = 10^{-2}$. We designate $g = 5$ such groups of possible 16 (each with 32 elements) to be “active”. The noise $\varepsilon \sim \mathcal{N}(0, 0.01)$. Thus (44) has the form (1), where $F(x) = Ax - b$. We set the absolute and relative exit tolerances to be 10^{-4} each. The number of subproblem iterations is capped at 100 for each outer iteration.

Figure 1 shows the solutions of each algorithm, and Table 1 reports the statistics. All algorithms arrive at approximately the same solution. `R2` requires the most function evaluations whereas the others require about the same. Table 1 suggests that a tradoff exists between the number of proximal operator evaluations and the number of gradient/Jacobian-vector evaluations. `TR` takes many proximal iterations, whereas `LMTR` and `LM` take far fewer. This tradeoff is further exemplified in the next test cases.

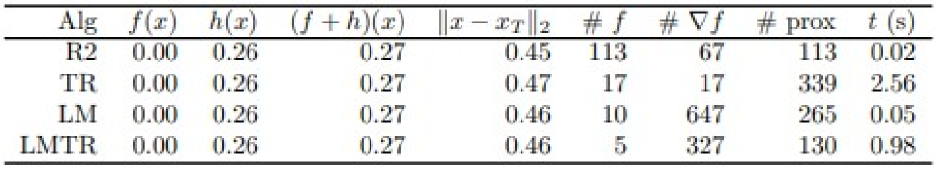
We additionally plot descent history in Figure 4a. The plots are roughly similar, with the trust region methods `TR` and `LMTR` performing the best.

6.2 Nonlinear support vector machine

We now solve an image recognition problem of the form (1), where

$$F(x) = \mathbf{1} - \tanh(b \odot \langle A, x\rangle), \quad \mathbf{1} = [1, \dots, 1]^T, \tag{45}$$

Table 1: Group-LASSO (44) statistics for R2, TR, LM, and LMTR, and $h(x) = \|x\|_{1,2}$. The $\#\nabla f$ is the number of $J^T v$ for LM and LMTR.



Alg	$f(x)$	$h(x)$	$(f+h)(x)$	$\|x - x_T\|_2$	$\# f$	$\# \nabla f$	# prox	t (s)
R2	0.00	0.26	0.27	0.45	113	67	113	0.02
TR	0.00	0.26	0.27	0.47	17	17	339	2.56
LM	0.00	0.26	0.27	0.46	10	647	265	0.05
LMTR	0.00	0.26	0.27	0.46	5	327	130	0.98

$A \in \mathbb{R}^{m \times n}$, $n = 784$ is the vectorized image size, the number of images is $m = 13007$ in the training set and $m = 2163$ in the test set, and $\odot$ denotes the elementwise product between vectors. We wish to use this nonlinear SVM to classify digits of the MNIST dataset as either 1 or 7, with all other digits removed. We additionally impose the condition that the support is sparse, and therefore use $h(x) = \|x\|_{1/2}^{1/2}$ as a regularizer. Hence, our overall problem is

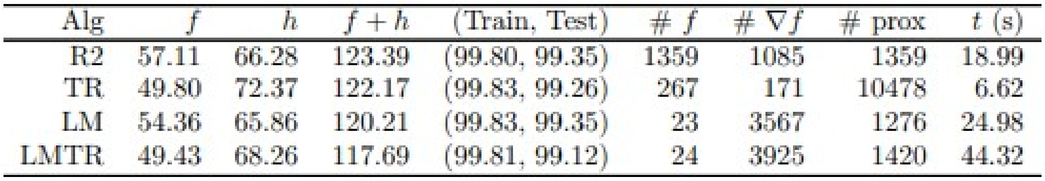
$$\min_x \ \tfrac{1}{2}\|\mathbf{1} - \tanh(b \odot \langle A, x \rangle)\|^2 + \lambda \|x\|_{1/2}^{1/2} \tag{46}$$

with $\lambda = 10^{-1}$. We initialize the problem at $x = \mathbf{1}^n$ so that approximately 50% of the data is misclassified. We set the stopping tolerances again to 10^{-4} and the maximum number of inner iterations to 100.

Figure 2 shows the solution map of each algorithm, which can be interpreted as the pixels most important in determining whether the image is indeed a 1 or 7. All algorithms produce a sparse solution; only about 8% of pixels in the support vector are nonzero. The problem is large and nonconvex; hence, the final solutions share pixels but altogether, they are different. This can be seen in Table 2, which reports the statistics. R2 again requires the most function evaluations. TR requires about 10 times more than LM and LMTR. We again observe that a tradoff exists between number of proximal operator evaluations and the number of gradient/Jacobian-vector evaluations. Here, proximal operator evaluations are cheaper than gradient or Jv evaluations, so wallclock time is higher for LM and LMTR.

We plot descent history against number of function/residual iterations in Figure 4b. Here we can see LM and LMTR performing the best in terms of descent.

Table 2: Nonlinear SVM (46) statistics for R2, TR, LM, and LMTR. Training/test error is with respect to the ℓ_2-norm.



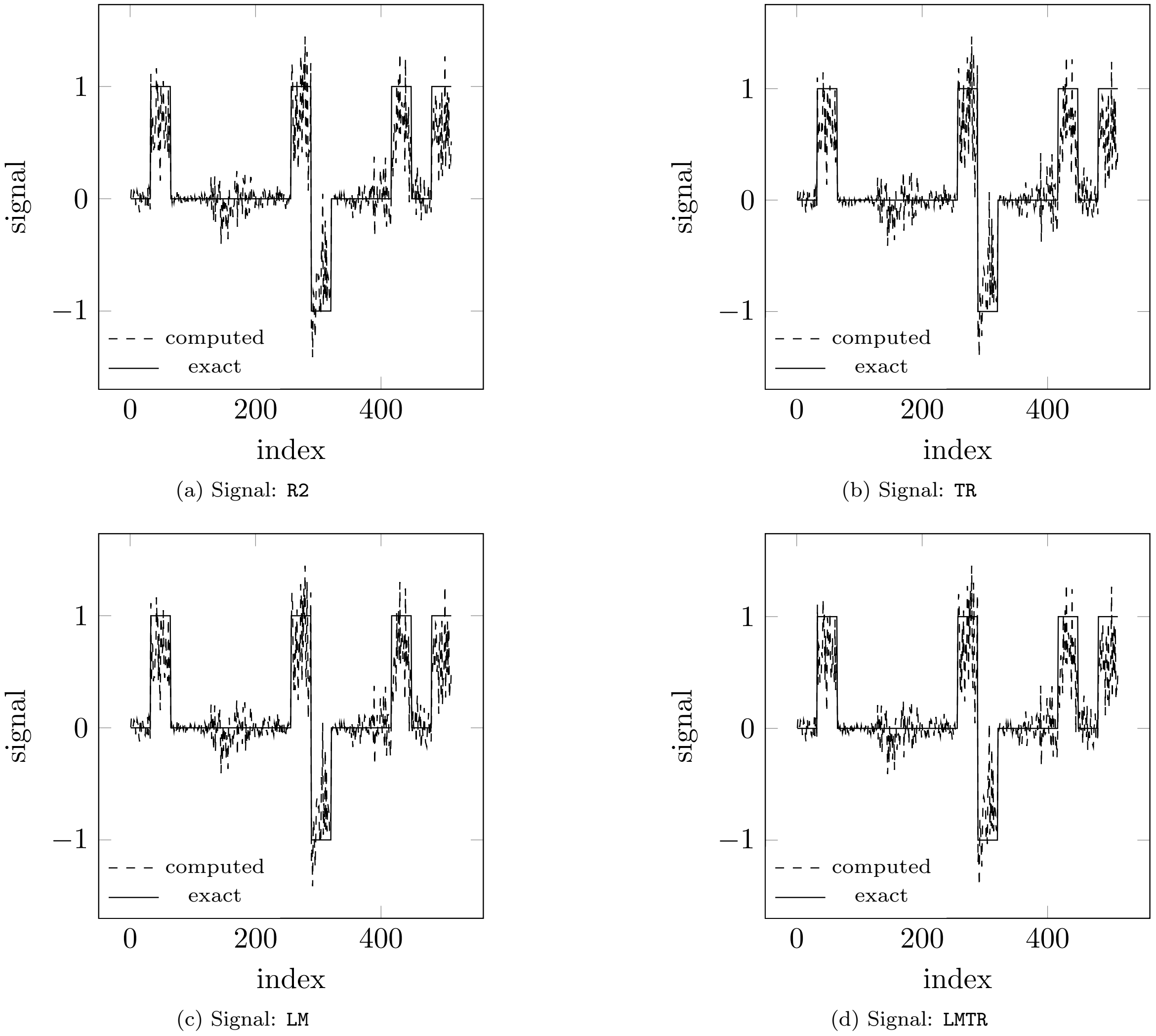
Alg	f	h	$f+h$	(Train, Test)	$\# f$	$\# \nabla f$	# prox	t (s)
R2	57.11	66.28	123.39	(99.80, 99.35)	1359	1085	1359	18.99
TR	49.80	72.37	122.17	(99.83, 99.26)	267	171	10478	6.62
LM	54.36	65.86	120.21	(99.83, 99.35)	23	3567	1276	24.98
LMTR	49.43	68.26	117.69	(99.81, 99.12)	24	3925	1420	44.32

6.3 FitzHugh-Nagumo inverse problem

The problem has the form (1), with $F : \mathbb{R}^5 \to \mathbb{R}^{2n+2}$ defined as $F(x) = (v(x) - \bar{v}(\bar{x}), w(x) - \bar{w}(\bar{x}))$, where $v(x) = (v_1(x), \ldots, v_{n+1}(x))$ and $w(x) = (w_1(x), \ldots, w_{n+1}(x))$ are sampled values of discretized functions $V(t; x)$ and $W(t; x)$ satisfying the FitzHugh [15] and Nagumo et al. [22] model for neuron activation

$$\frac{\mathrm{d}V}{\mathrm{d}t} = (V - V^3/3 - W + x_1)x_2^{-1}, \qquad \frac{\mathrm{d}W}{\mathrm{d}t} = x_2(x_3 V - x_4 W + x_5), \tag{47}$$

parametrized by x. The sampling is defined by a discretization of the time interval $t \in [0, 20]$ and initial conditions $(V(0), W(0)) = (2, 0)$. The data $(\bar{v}(x), \bar{w}(x))$ is generated by solving (47) with



(a) Signal: R2

(b) Signal: TR

(c) Signal: LM

(d) Signal: LMTR

Figure 1: Group-LASSO (44) solutions with R2, TR, LM, and LMTR with $h = \lambda \| \cdot \|_{1,2}$.

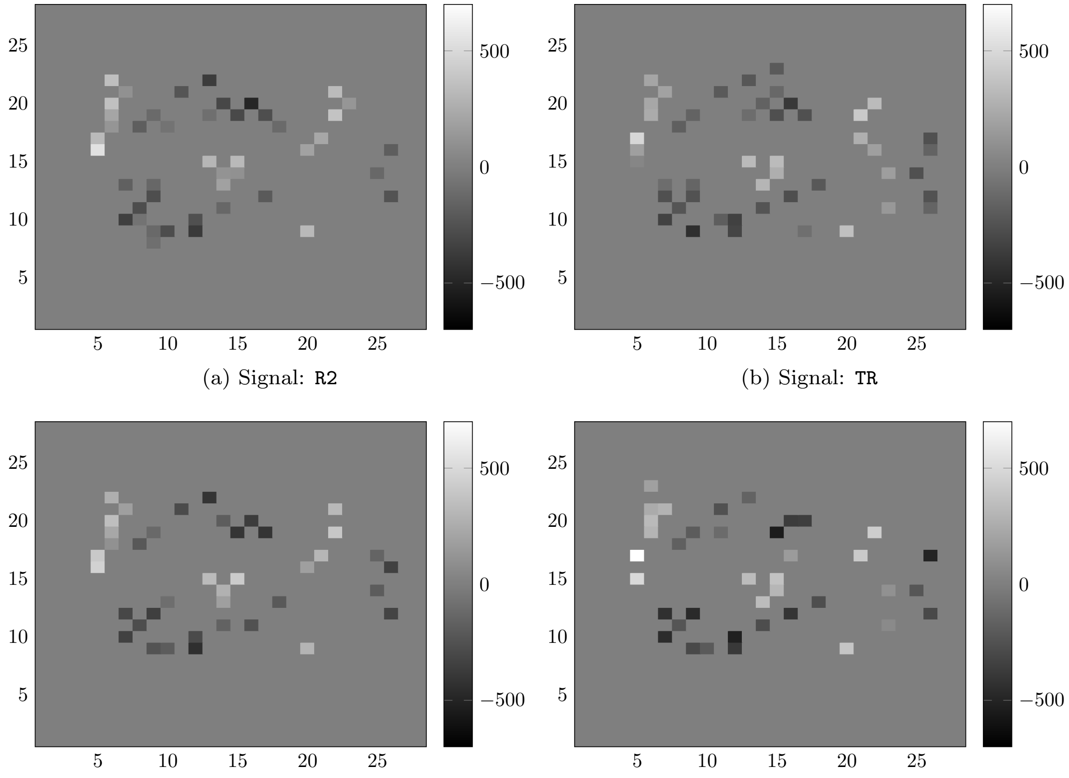
$\bar{x} = (0, 0.2, 1, 0, 0)$, which corresponds to a simulation of the Van der Pol [29] oscillator. In our experiments, we use $n = 100$ and solve

$$\min_x \tfrac{1}{2}\|F(x)\|_2^2 + \lambda\|x\|_1, \tag{48}$$

where $h(x) = \lambda\|x\|_1$ with $\lambda = 10$ to enforce sparsity in the parameters. Our absolute stopping criteria is 10^{-2}, whereas our the relative stopping criteria is set to 10^{-4}.

The solution found by each solver is given in Table 3 TR has the correct nonzero parameters, but the values are farther off. The corresponding simulations are shown in Figure 3; each method is able to fit the data.

Table 4 reports the statistics for each algorithm, which exhibit the same pattern of results as before. The final objective values are fairly similar. LMTR uses the smallest amount of objective evaluations, whereas LM has a harder time solving (48). Because the gradient of the smooth term in (48) is not Lipschitz continuous, we had to set a $\sigma_{\min}$ for both R2 and LM, which increased iteration count. Similar to the SVM example, we can see that LM and LMTR take more time than TR, which again stems from proximal operators being much cheaper to compute than Jv products for this example. Notably, TR seems to fit the data worse but attain a lower value of the regularizer.



(a) Signal: `R2`

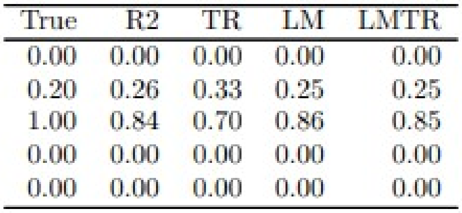
(b) Signal: `TR`

(c) Signal: `LM`

(d) Signal: `LMTR`

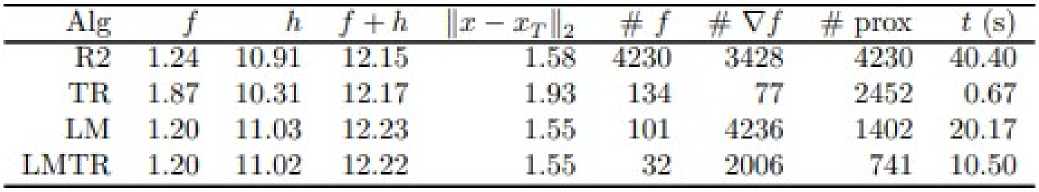
Figure 2: Nonlinear SVM (46) solutions with `TR`, `R2`, `LM`, `LMTR`.

Table 3: Final parameters for the FH problem (48) found by `R2`, `TR`, `LM`, and `LMTR`.



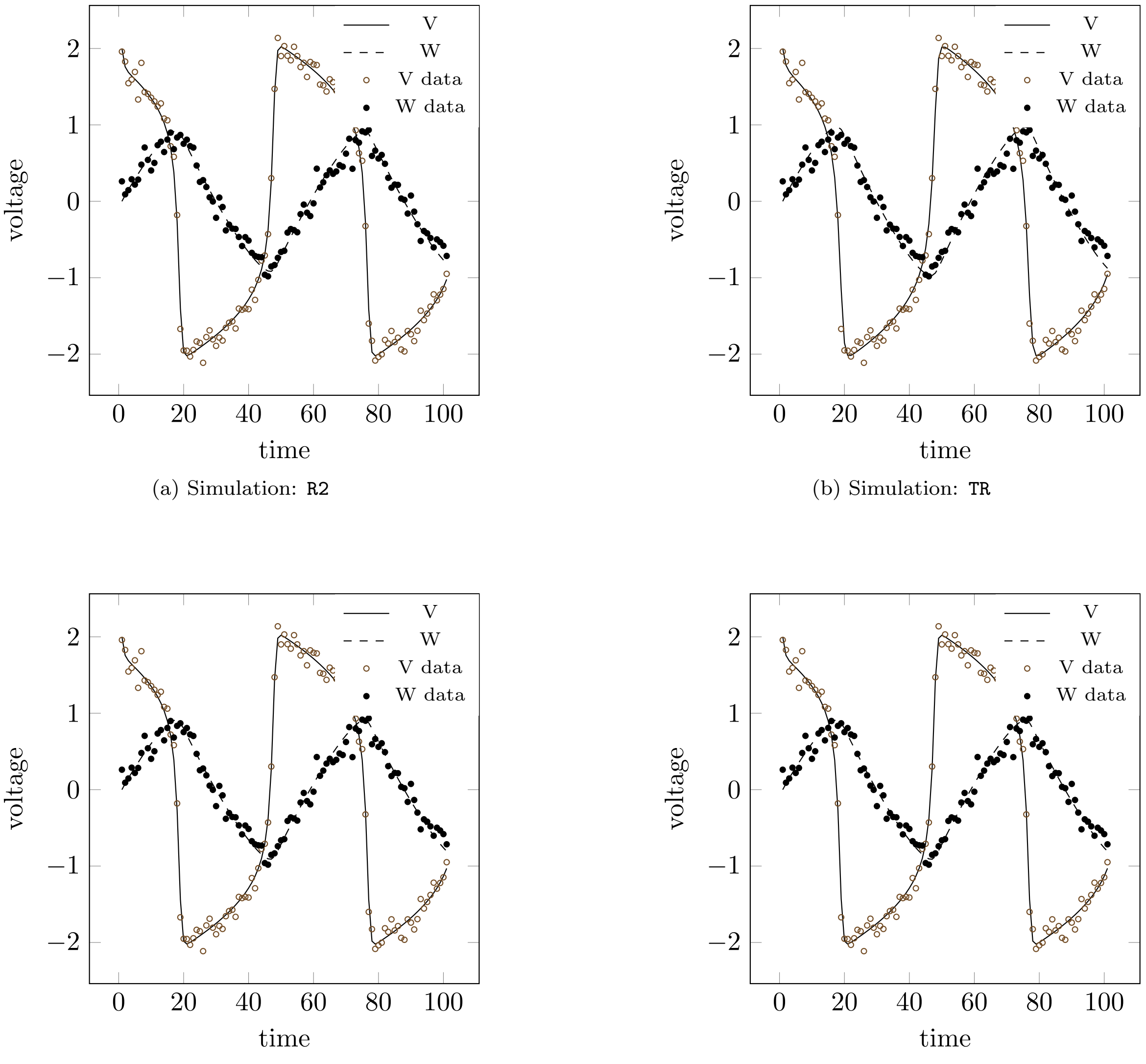
True	R2	TR	LM	LMTR
0.00	0.00	0.00	0.00	0.00
0.20	0.26	0.33	0.25	0.25
1.00	0.84	0.70	0.86	0.85
0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00

Table 4: Statistics for the FH problem (48) for `R2`, `TR`, `LM`, and `LMTR`.



Alg	f	h	$f+h$	$\|x - x_T\|_2$	$\# f$	$\# \nabla f$	# prox	t (s)
R2	1.24	10.91	12.15	1.58	4230	3428	4230	40.40
TR	1.87	10.31	12.17	1.93	134	77	2452	0.67
LM	1.20	11.03	12.23	1.55	101	4236	1402	20.17
LMTR	1.20	11.02	12.22	1.55	32	2006	741	10.50

Finally, Figure 4c shows descent of our objective function value against objective function iteration. `LMTR` again performs the best, whereas `LM` and `TR` were similar in this metric. This again enunciates the tradeoff between objective, gradient, and proximal operator expense. Expensive proximal evaluations would be the limiting factor in `TR` and `R2`; one can think of Total Variation regularization as a test case, since the proximal operator is itself a minimization problem.



(a) Simulation: `R2`

(b) Simulation: `TR`

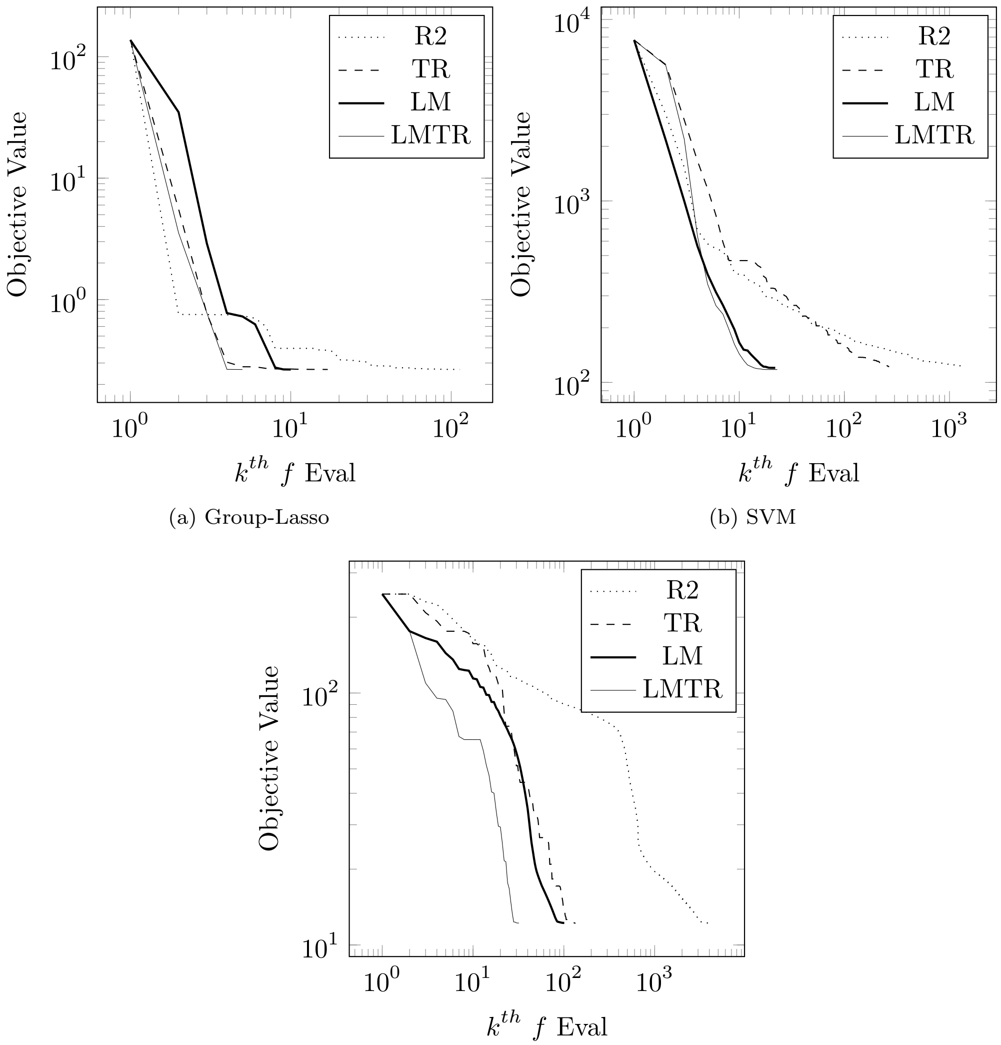
(c) Simulation: `LM`

(d) Simulation: `LMTR`

Figure 3: Simulation of the FH problem (48) solutions found by `R2`, `TR`, `LM`, `LMTR`.

7 Discussion

Similarly to smooth optimization, exploiting the least-squares structure of f can decrease significantly the number of outer iterations. The challenge highlighted by our numerical results, which is the subject of ongoing research, is to either identify a closed-form minimizer of (6c) for relevant choices of ψ, or to devise methods that can produce a higher-quality step than `R2` with fewer transposed-Jacobian-vector products. As long as the subproblem solver yields a step satisfying Step Assumption 4.1, our convergence properties and worst-case complexity bounds are guaranteed to hold. Thus, any improvement in the step computation mechanism will immediately translate into a more efficient solver overall. In ongoing research, we are exploring other improvements, including inexact evaluations of f and ∇f, nonmonotone methods, and inexact evaluation of proximal operators.



(a) Group-Lasso

(b) SVM

(c) FH

Figure 4: Objective decrease per objective or residual evaluation.

References

[1] A. Aravkin, R. Baraldi, and D. Orban. A proximal quasi-Newton trust-region method for nonsmooth regularized optimization. SIAM J. Optim., (2):900–929, 2022. DOI: 10.1137/21M1409536.

[2] Francis Bach, Rodolph Jenatton, Julien Mairal, and Guillaume Obozinski. Optimization with Sparsity-Inducing Penalties, volume 4 of Foundations and Trends in Machine Learning. now publishers, 2012. DOI: 10.1561/2200000015.

[3] R. Baraldi and D. Orban. RegularizedOptimization.jl: Algorithms for regularized optimization. `https://github.com/JuliaSmoothOptimizers/RegularizedOptimization.jl`, February 2022.

[4] R. Baraldi and D. Orban. RegularizedProblems.jl: Test cases for regularized optimization. `https://github.com/JuliaSmoothOptimizers/RegularizedProblems.jl`, February 2022.

[5] R. Baraldi and D. Orban. ShiftedProximalOperators.jl: Proximal operators for regularized optimization. `https://github.com/JuliaSmoothOptimizers/ShiftedProximalOperators.jl`, February 2022.

[6] Amir Beck. First Order Methods in Optimization. SIAM, Philadelphia, USA, 2017. DOI: 10.1137/1.9781611974997.

[7] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. SIAM Rev., 59(1):65–98, 2017. URL `https://doi.org/10.1137/141000671`.

[8] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Program., (146):459—-494, 2014. DOI: 10.1007/s10107-013-0701-9.

[9] R. I. Boţ, E. R. Csetnek, and S.C. László. An inertial forward–backward algorithm for the minimization of the sum of two nonconvex functions. EURO J. Comput. Optim., (4):3–25, 2016. DOI: 10.1007/s13675-015-0045-8.

[10] Wenfei Cao, Jian Sun, and Zongben Xu. Fast image deconvolution using closed-form thresholding formulas of lq (q = 12, 23) regularization. Journal on visual communication and image representation, 24(1), 2013.

[11] Coralia Cartis, Nicholas I. M. Gould, and Ph. L. Toint. On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming. SIAM J. Optim., 21(4):1721–1739, 2011. DOI: 10.1137/11082381X.

[12] Patrick L Combettes and Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In Fixed-point algorithms for inverse problems in science and engineering, pages 185–212. Springer, 2011. DOI: 10.1007/978-1-4419-9569-8˙10.

[13] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. Trust-Region Methods. Number 1 in MOS-SIAM Series on Optimization. SIAM, Philadelphia, USA, 2000. DOI: 10.1137/1.9780898719857.

[14] David L Donoho. Compressed sensing. IEEE T. Inform. Theory, 52(4):1289–1306, 2006. DOI: 10.1109/TIT.2006.871582.

[15] Richard FitzHugh. Mathematical models of threshold phenomena in the nerve membrane. B. Math. Biophys., 17(4):257–278, 1955. DOI: 10.1007/BF02477753.

[16] Masao Fukushima and Hisashi Mine. A generalized proximal point algorithm for certain non-convex minimization problems. Int. J. Syst. Sci., 12(8):989–1000, 1981. DOI: 10.1080/00207728108963798.

[17] G.N. Grapiglia, J. Yuan, and Yx. Yuan. Nonlinear stepsize control algorithms: Complexity bounds for first- and second-order optimality. J. Optim. Theory and Applics., (171):980—997, 2016. DOI: 10.1007/s10957-016-1007-x.

[18] Jason D. Lee, Yuekai Sun, and Michael A. Saunders. Proximal Newton-type methods for minimizing composite functions. SIAM J. Optim., 24(3):1420–1443, 2014. DOI: 10.1137/130921428.

[19] K. Levenberg. A method for the solution of certain problems in least squares. Q. Appl. Math., (2):164–168, 1944. DOI: 10.1090/qam/10666.

[20] Huan Li and Zhouchen Lin. Accelerated proximal gradient methods for nonconvex programming. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS'15, pages 379–387, Cambridge, MA, USA, 2015. MIT Press. URL http://irc.cs.sdu.edu.cn/973project/result/download/2015/28.AcceleratedProximal.pdf.

[21] Donald W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. Journal of the Society for Industrial and Applied Mathematics, 11(2):431–441, 1963. DOI: 10.1137/0111030.

[22] Jinichi Nagumo, Suguru Arimoto, and Shuji Yoshizawa. An active pulse transmission line simulating nerve axon. Proceedings of the IRE, 50(10):2061–2070, 1962. DOI: 10.1109/JRPROC.1962.288235.

[23] Jarrett Revels. Reverse mode automatic differentiation for Julia. https://github.com/JuliaDiff/ReverseDiff.jl, 2022.

[24] Jarrett Revels, Miles Lubin, and Theodore Papamarkou. Forward-mode automatic differentiation in Julia, 2016. URL https://arxiv.org/abs/1607.07892.

[25] R.T. Rockafellar and R.J.B. Wets. Variational Analysis, volume 317. Springer Verlag, 1998. DOI: 10.1007/978-3-642-02431-3.

[26] L. Stella, A. Themelis, P. Sopasakis, and P. Patrinos. A simple and efficient algorithm for nonlinear model predictive control. In 2017 IEEE 56th Annual Conference on Decision and Control (CDC), pages 1939–1944, 2017. DOI: 10.1109/CDC.2017.8263933.

[27] Andreas Themelis, Lorenzo Stella, and Panagiotis Patrinos. Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone linesearch algorithms. SIAM J. Optim., 28 (3):2274–2303, 2018. DOI: 10.1137/16M1080240.

[28] Robert Tibshirani. Regression shrinkage and selection via the lasso. J. Roy. Statist. Soc. Ser. B, 58(1): 267–288, 1996. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

[29] Balth Van der Pol. Lxxxviii. On "relaxation-oscillations". The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2(11):978–992, 1926. DOI: 10.1080/14786442608564127.

[30] Hao Zhu, Geert Leus, and Georgios B. Giannakis. Sparsity-cognizant total least-squares for perturbed compressive sampling. IEEE T. Signal Proces., 59(5):2002–2016, 2011. DOI: 10.1109/TSP.2011.2109956.