An Algorithm for Piecewise Linear Optimization of Objective Functions in Abs-normal Form

Andreas Griewank[1] and Andrea Walther[2]

Draft of December 22, 2017

Abstract

In the paper [11] we derived first order (KKT) and second order (SSC) optimality conditions for functions defined by evaluation programs involving smooth elementals and absolute values. For this class of problems we showed in [12] that the natural algorithm of successive piecewise linear optimization with a proximal term (SPLOP) achieves a linear or even quadratic rate of convergence under suitable assumptions. A version of SPLOP called LiPsMin has already been implemented and tested in [13, 3]. In this paper, we develop a more efficient method for the inner loop, i.e., the minimization of the local piecewise linear model with a quadratic regularization term. Rather than completely solving each Quadratic Optimization Problem (QOP) on one of the signature domains, we may switch between them as soon as we have reached a merely stationary point. The resulting active set and signature strategy very much resembles the classical method for convex QOPs and utilizes the very same numerical linear algebra techniques. Preliminary numerical results document an order of magnitude improvement compared to the original proof of concept implementation, which was already competitive with alternative nonsmooth optimization methods.

Keywords: Successive Piecewise Linear Optimization (SPLOP), Quadratic Regularization, Abs-Normal Form, Linear Independence Kink Qualification (LIKQ), Tangential Stationarity, Karush Kuhn Tucker (KKT), Normal Growth, Active Set and Signature.

[1]School of Mathematical Sciences and Information Technology, Yachay Tech, Urcuquí, Imbabura, Ecuador

[2]Institut für Mathematik, Universität Paderborn, Paderborn, Germany

1 Introduction and Motivation

All continuous piecewise linear functions $y = \varphi(x) : \mathbb{R}^n \to \mathbb{R}$ can be represented in the abs-normal form

$$Mz = \mathring{x} + Z(x - \mathring{x}) + L(|z| - |\mathring{z}|) \quad = \quad \underbrace{(\mathring{x} - Z\mathring{x} - L|\mathring{z}|)}_{=c} + Zx + L|z| \;, \quad (1)$$

$$y = \mathring{y} + a^\top(x - \mathring{x}) + b^\top(|z| - |\mathring{z}|) \quad = \quad \underbrace{(\mathring{y} - a^\top\mathring{x} - b^\top|\mathring{z}|)}_{=0} + a^\top x + b^\top|z| \quad (2)$$

Here $x \in \mathbb{R}^n$ is the vector of independent variables, $z \in \mathbb{R}^s$ the vector of switching variables, and $Z \in \mathbb{R}^{s\times n}, L, M \in \mathbb{R}^{s\times s}$ are matrices, where L is strictly lower triangular and M unitary lower triangular. We will prefer to use the second, nonincremental form on the right hand sides of Eqs. (1) and (2) where the constant term in the objective can be assumed zero without loss of generality.

In the original form introduced in [7] and analyzed in several subsequent papers, see, e.g., [11], the matrix M was simply the identity. This assumption represents theoretically no loss of generality because we can multiply Eq. (1) from the left by M^{-1} yielding

$$z = M^{-1}c + M^{-1}Zx + M^{-1}L|z|$$

as defining equation for the switching variables. However, as is shown in the appendix, repeated application of $\max(\cdot,\cdot)$, $\min(\cdot,\cdot)$ or the positive part function $\max(0,\cdot)$ may conveniently be expressed using a bidiagonal matrix M, whose inverse is dense though diagonally dominant. Hence, it is for computational purposes advisable to allow for a general unitary lower triangular matrix M. Due to the lower triangularity assumption on $M^{-1}L$ there exists for all x a unique $z = z(x) \in \mathbb{R}^s$, each component of which is a piecewise linear, continuous function of x on the whole of $\mathbb{R}^n$.

The form (1) – (2) arises naturally if one approximates a given piecewise smooth function of the class first introduced in [6] by a piecewise linearization at the reference point $\mathring{x}$. Then, the matrices M, Z, L and the vectors a, b, c can be obtained by a minor extension of standard automatic differentiation using tools like ADOL-C [26] and Tapenade [15]. Given a coercive quadratic objective as a function of x and $|z|$ one may then successively solve these essentially piecewise linear model problems. As shown in [12], this successive piecewise linear optimization (SPLOP) approach yields linear convergence if the second order sufficiency condition (SSC) with strict complementarity holds, provided, the Linear Independence Kink Qualification (LIKQ) introduced in [11] is satisfied. Moreover, it was shown that in the case of sharp local minimizers, where the function exhibits linear growth, SPLOP converges quadratically. In this paper we will develop and test an efficient solver of the inner loop problem.

The paper is organized as follows. In the following Subsec. 1.1 we discuss the polyhedral domain decomposition implicitly defined by Eq. (1) in some detail and fix the notation appropriately. In Subsec. 1.2 we introduce the regularized nonsmooth objective and reformulate the task of minimizing it as a family of QOPs. For these we adapt in the subsequent Subsec. 1.3 the local first order optimality condition from [11]. In Sec. 2 we develop the actual algorithm, starting in Subsec. 2.1 with the computation of stationary points by successively activating kinks, continuing with the procedure for dropping active kinks in Subsec. 2.2, and specifying the method using an informal language representation in Subsec. 2.3. In Sec. 3, we discuss efficient ways of performing the numerical linear algebra tasks arising in the step computation. We verify the method numerically on four test problems in Sec. 4. Sec. 5 contains summary, conclusion, and outlook. Finally, we study in Appendix 6 the generalization in the computation of the switching variables as described above.

1.1 Walking the polyhedral decomposition of $\mathbb{R}^n$

For each $x \in \mathbb{R}^n$ and the corresponding $z = z(x)$ we may define the signature vector and matrix as

$$\sigma = \sigma(x) = \operatorname{sgn}(z(x)) \in \{-1, 0, 1\}^s \quad \text{and} \quad \Sigma = \Sigma(x) = \operatorname{diag}(\sigma(x)) \ .$$

Sometimes we will consider σ and Σ as constant and discuss the corresponding arguments x that form the inverse images

$$P_\sigma \ = \ \{x \in \mathbb{R}^n : \operatorname{sgn}(z(x)) = \sigma\} \ \subset \ \bar{P}_\sigma \{x \in \mathbb{R}^n : \Sigma z(x) = |z(x)|\} \ ,$$

called *signature domains*, which represent a disjoint decomposition of $\mathbb{R}^n$ into relatively open polyhedra. Their *essential closures* $\bar{P}_\sigma$ are by continuity of the functional dependence $z(x)$ closed and may be nontrivial even when P_σ is empty. As shown in [13] their partial ordering by inclusion is reflected in the partial signature ordering

$$\bar{P}_\sigma \subset \bar{P}_{\tilde{\sigma}} \iff \sigma \prec \tilde{\sigma} \iff \sigma_i^2 \leqslant \tilde{\sigma}_i \sigma_i \ \text{ for } \ 1 \leqslant i \leqslant s \ .$$

It is also easy to check that $\sigma \prec \tilde{\sigma}$ implies $P_\sigma \subset P_{\tilde{\sigma}}$ although the converse might not hold if $P_\sigma = \varnothing$. The signatures $\sigma \in \{-1, 1\}^s$ are called definite and the associated P_σ are by continuity open, which may however also happen if σ contains zeros and is thus indefinite. Then we may extend σ to a definite $\tilde{\sigma} \succ \sigma$ and will obtain the definite representation $\bar{P}_\sigma = \bar{P}_{\tilde{\sigma}}$. In any case we have

$$\bigcup_{\sigma \in \{-1,1\}^s} \bar{P}_\sigma \ = \ \mathbb{R}^n \ = \ \bigcup_{\sigma \in \{-1,0,1\}^s} P_\sigma \ ,$$

where the first decomposition is overlapping, the second is not. Any algorithm for solving piecewise smooth problems will walk the polyhedral decomposition in a suitable fashion, hopefully touching not too many P_σ or

$\bar{P}_\sigma$. However, a Klee Minty type exhaustive search, see [18], may theoretically occur on very specific problems. Of course, it is neither necessary nor advisable to ever compute the complete polyhedral decomposition or even a significant part of it explicitly.

Any uniformly convex continuous objective must attain a unique minimizer x_σ on each one of the closed sets $\bar{P}_\sigma$. If the minimizer x_σ lies in the corresponding relative interior P_σ itself, we label it and the corresponding signature as *stationary*. That property is sometimes called subspace minimality in the quadratic optimization literature [27]. Generally speaking, such x_σ satisfy the KKT conditions, except that some Lagrange multipliers may have the wrong sign.

Our strategy will be a simple generalization of the usual active set QOP method to first find a stationary point x_σ, and then release one of the kinks with the wrong multiplier sign, thus making the corresponding switching variable basic to obtain a new signature vector $\tilde{\sigma}$. Subsequently, we compute a step through $P_{\tilde{\sigma}}$ to either reach $x_{\tilde{\sigma}}$ if it is stationary, or pick up a new kink, which will become active, so that the corresponding switching variable becomes nonbasic. The somewhat surprising aspect is that while P_σ has generally $2^{n-\|\sigma\|_1}$ neighbor polyhedra, and there are thus just as many neighboring restricted optimization problems, there exists a common vector of Lagrange multipliers, which is easy to compute, provided LIKQ holds. Since there are only finitely many stationary polyhedra and we consistently reduce the common objective function value such an algorithm must converge in finitely may steps.

1.2 The regularized objective function

Without loss of generality we may shift x such that $x_0 = 0$ and set $y_0 = 0$. Since $\varphi(x)$ may be unbounded below, we add to the objective y a regularization term $\frac{1}{2}x^\top Q x$ with a positive definite matrix $Q = Q^\top \in \mathbb{R}^{n\times n}$. This term may be primarily viewed as a bound on the discrepancy between the piecewise linear model and the underlying piecewise smooth function. The matrix Q is often chosen simply equal to qI with $0 < q \in \mathbb{R}$ and we may later allow it to be a symmetric matrix, whose projection on certain null spaces is positive definite. In this way, we expect to achieve superlinear convergence rates by secant updating of Q, a technique well known from smooth constrained optimization [22]. Using the polyhedral decomposition one may write our nonsmooth optimization task as minimum over a set of smooth optimization problems

$$\min_{\sigma,x,z} a^\top x + b^\top \Sigma z + \tfrac{1}{2}x^\top Q x \quad \text{s.t.} \quad x \in \bar{P}_\sigma \;, \tag{3}$$

where the σ may be restricted to be definite ones or not. Due to the assumed positive definiteness of Q there exists for each branch problem defined by

$\sigma \in \{-1, 0, 1\}^s$ a global minimizer the smallest one of whom is the global solution of the original problem of minimizing $\varphi(x) + \frac{1}{2}x^\top Q\, x$. Notice that the feasible sets for indefinite signatures σ are contained in those for corresponding definite signatures $\tilde{\sigma} \succ \sigma$, where all zeros are replaced by 1 or -1, which also leaves the restriction of the objective function to P_σ unchanged. It is a key advantage of the abs-normal representation that the 3^s branch problems including the 2^s definite ones need not be considered separately but that their connectivity is exploited both for the optimality conditions and by the algorithm proposed here.

1.3 First Order Minimality Conditions (FOM)

For each signature vector $\sigma \in \{-1, 0, 1\}^s$ we have the quadratic optimization problem

$$\min_{x,z} a^\top x + b^\top \Sigma z + \tfrac{1}{2}x^\top Q x \qquad \text{s.t.}$$
$$M|\Sigma|z = c + Zx + L\Sigma z, \qquad |\bar{\Sigma}|z = 0\ , \qquad \Sigma z \geqslant 0\ ,$$

where $|\bar{\Sigma}| = I - |\Sigma|$ is the complementary orthogonal projection to $|\Sigma|$. Here, the feasible set is the closed version $\bar{P}_\sigma$, which will of course frequently be empty when the implicit condition $\Sigma z = |z|$ cannot be satisfied at any $x \in \mathbb{R}^n$. For all definite Σ we have $|\Sigma| = I$ and the equations can be simplified somewhat, but in contrast to earlier presentations, see, e.g., [11], we wish to explicitly deal with the indefinite ones as well and force the corresponding components of z to zero.

Applying standard KKT theory, we get the optimality conditions

$$\begin{aligned}
0 &= c + Zx + (L\Sigma z - M|\Sigma|)z, && (4)\\
0 &= |\bar{\Sigma}|z && (5)\\
0 &\leqslant \Sigma z && (6)\\
0 &= a^\top + x^\top Q + \lambda^\top Z && (7)\\
0 &= b^\top \Sigma + \lambda^\top L\Sigma - \lambda^\top M|\Sigma| + \gamma^\top|\bar{\Sigma}| - \mu^\top \Sigma && (8)\\
0 &\leqslant \mu && (9)\\
0 &= \mu^\top \Sigma z\ . && (10)
\end{aligned}$$

Obviously, Eq. (4) with Eq. (5) and Eq. (6) represents primal feasibility so that x must belong to the closure $\bar{P}_\sigma$. In [11], Eq. (7) was called *tangential stationarity* and Eq. (8) with Eq. (9) *normal growth*. In other words, tangential stationarity represents KKT without sign conditions on the Lagrange multipliers, which are then enforced additionally as normal growth.

When Q is not a multiple of the identity it may be advantageous to perform its Cholesky factorization $Q = C^\top C$ and to use $\tilde{Z} = ZC^{-1}$, $\tilde{c} = C^{-1}c$

and $\tilde{x} = Cx$ in the formulation of the optimality conditions. This yields $\tilde{Q} = I$ and all other quantities including z and λ remain unchanged. This a priori transformation may be computationally efficient if the algorithms proposed in the next Sec. 2 can be expected to take quite a few steps and the transformation from Z to $\tilde{Z}$ does not create too much fill-in.

2 The generalized quadratic optimization algorithm

In contrast to standard linear or quadratic problems we have no problem finding a feasible starting point. Starting with literally any $x_0 \in \mathbb{R}$ we can compute $z_0 = z(x_0)$ by Eq. (1) and set initially $\sigma = \mathrm{sgn}(z_0)$. For a general x_0 the resulting σ is likely to be definite, which means that all kinks are flagged as inactive initially. At least if the curvature term is quite small that usually will mean that the algorithm takes close to n steps in order to build up a significant set of kinks before finding a stationary point. Especially, in the context of Successive Piecewise Linear Optimization it is advisable to begin with a warm start as far as the active kinks are concerned. However, one should check whether the old σ satisfies $\sigma \prec \sigma(x_0)$, otherwise the initial point may be on the wrong side of some kinks or P_σ may be empty altogether.

2.1 Finding a Stationary Point

Multiplying Eq. (8) from the right by Σ, we get

$$0 \leqslant \mu^\top |\Sigma| \;\equiv\; b^\top |\Sigma| + \lambda^\top L|\Sigma| - \lambda^\top M\Sigma \,, \tag{11}$$

which, if $\sigma = \mathrm{sgn}(z)$ so that the underlying x is stationary, reduces because of the required complementarity $\mu^\top \Sigma z = 0$ to

$$0 = b^\top |\Sigma| + \lambda^\top L|\Sigma| - \lambda^\top M\Sigma \,.$$

Hence, if the stationary point x_σ exists it must satisfy the system of equations

$$\begin{bmatrix} a + Qx + Z^\top \lambda \\ \Sigma b + (\Sigma L^\top - |\Sigma| M^\top)\lambda + |\bar{\Sigma}| z \\ c + Zx + (L\Sigma - M|\Sigma|)z \end{bmatrix} \;=\; 0 \,. \tag{12}$$

Here, we have scaled the middle block once more by Σ to make the resulting Jacobian fully symmetric. Moreover we have added $|\bar{\Sigma}|z$ to force the nonbasic components of z to zero and to hopefully make the whole system nonsingular. Thus we obtain the saddle point system

$$\begin{bmatrix} Q & 0 & Z^\top \\ 0 & |\bar{\Sigma}| & \Sigma L^\top - |\Sigma| M^\top \\ Z & L\Sigma - M|\Sigma| & 0 \end{bmatrix} \begin{bmatrix} x \\ z \\ \lambda \end{bmatrix} = - \begin{bmatrix} a \\ \Sigma\, b \\ c \end{bmatrix} . \tag{13}$$

It can be solved quite effectively using the special structure of the triangular matrices, as will be shown in the next section.

In many cases the resulting Σz may have negative components so that there is a blocking constraint. That is, with z part of the solution of the system given in Eq. (13) and z^{now} the current iterate, which is maintained primally feasible, i.e., belongs to P_σ for the current σ, one has $\beta_j < 1$ where

$$\beta_j = \inf_{1 \leqslant i \leqslant s} \{\beta_i \equiv -z_i^{now}/(z_i - z_i^{now}) \mid z_i^{now}(z_i - z_i^{now}) < 0\} \in (0, \infty] \,. \quad (14)$$

If the minimum is obtained first for some index j, we have to reduce $\sigma = \sigma - \sigma_j e_j$, which means setting $\sigma_j = 0$. The primal variables are simply updated according to $z^{now} = (1-\beta_j)z^{now} + \beta z$ and possibly $x^{now} = (1-\beta_j)x^{now} + \beta x$ if desired. Then we repeat the step calculation with the new active set until a stationary point x_σ is found.

2.2 Releasing a kink

Now the question arises whether the stationary point x_σ is already a minimizer of $\varphi(x) + \frac{1}{2}x^\top Qx$ and if not, how we should continue the search in one of the neighboring polyhedra $P_{\tilde{\sigma}}$ with $\tilde{\sigma} > \sigma$. Any such $\tilde{\sigma}$ can be decomposed into $\sigma + \bar{\sigma}$ where $|\sigma|^\top |\bar{\sigma}| = 0$. Replacing Σ in the above optimality conditions by the corresponding $\Sigma + \bar{\Sigma}$ we see that the primal feasibility conditions and the dual equality constraint are still justified by the current values x, z and λ. The only thing that changes is that Eq. (11) has as many new nontrivial components as $\bar{\sigma}$ which can be written as

$$0 \leqslant \mu^\top |\bar{\sigma}| \;\equiv\; b^\top |\bar{\sigma}| + \lambda^\top L|\bar{\sigma}| - \lambda^\top M\bar{\sigma} \,. \quad (15)$$

This optimality condition is violated if and only if there is at least one index k such that $\bar{\sigma} = e_k \operatorname{sgn}(\lambda^\top M e_k)$ satisfies

$$0 \;>\; b_k + \lambda^\top L e_k - |\lambda^\top M| e_k \quad \text{and} \quad \sigma_k = 0 \,, \quad (16)$$

which represents a violation of the normal growth condition. It is a natural strategy to choose the k for which the right-hand side is minimal, but possibly other considerations might be applied as well. Then we can update $\sigma = \sigma + \bar{\sigma}$ and reenter the equality constrained phase from the old point $x = x_\sigma$.

2.3 The overall algorithm

What we have sketched above can be viewed as an active signature strategy, where the zeros of σ denote the active kinks and the nonzeros determine the sign of the basic variables. An informal language representation of this algorithm is given by:

Algorithm 1 Active signature method for piecewise linear minimization

Require: $n \in \mathbb{N}, s \in \mathbb{N},\ a \in \mathbb{R}^n,\ b, c \in \mathbb{R}^s,\ Z \in \mathbb{R}^{s\times n}, L \in \mathbb{R}^{s\times s}$ strictly lower triangular, $M \in \mathbb{R}^{s\times s}$ unitary lower triangular, $Q = Q^\top \in \mathbb{R}^{n\times n}$ positive definite.

Ensure: $z^{now} = z(x_0)$ via Eq. (1) and check $\sigma \prec \sigma(x_0)$.

```
while Eq. (13) numerically solvable do
    Compute β_j via Eq. (14)
    if β_j ⩾ 1 then
        z^now = z
        if Eq. (16) holds with numerical certainty then
            Determine k with σ_k = sgn(λ^⊤ M e_k)        ▷ Drop Kink k
        else
            Output x^now = −Q^{−1}(a + Z^⊤ λ)
            Terminate                                    ▷ Problem solved
        end if
    else
        z^now = (1 − β_j) z^now + β_j z
        σ_j = 0                                          ▷ Add Kink j
    end if
end while
Irregular termination due to near singularity.           ▷ Numerical Failure
```

Note that theoretically, our problem has always a global solution and the algorithm should at least find a local minimizer. However, due to degeneracies and numerical round-off the search may sometimes be unsuccessful. There remains some freedom in choosing the kink to be dropped or added, and we glossed over some of the numerical issues, one of which is the actual step calculation which is discussed in the following section.

3 Solving the saddle point system

There is an extensive numerical literature, see, e.g., [27], on solving saddle point systems of the form (13), where we have the additional structure that the block $(M - L\Sigma)$ is unitary triangular and hopefully rather sparse. On KKT systems of smooth NLOPs and other equations with switching depth 1, see, e.g., [8], the triangular matrix is just the identity. In any case we may apply the following method.

Lemma 3.1 (Partitioned solution)**.** *The solution of Eq.* (13) *can be reduced to solving the symmetric semi definite linear system*

$$\left[|\bar{\Sigma}|(M-L\Sigma)^{-1}ZQ^{-1}Z^\top(M^\top-\Sigma L^\top)^{-1}|\bar{\Sigma}|\right]\tilde{\lambda} \tag{17}$$

$$= |\bar{\Sigma}|(M-L\Sigma)^{-1}\left[c - ZQ^{-1}(a + Z^\top(M^\top-\Sigma L^\top)^{-1}\Sigma b)\right] \tag{18}$$

for the nontrivial entries of $|\bar{\Sigma}|\tilde{\lambda}$*. The system is uniquely solvable exactly when the rows of the* $s \times n$ *matrix* $(M - L\Sigma)^{-1}Z$ *that correspond to active kinks are linearly independent. The dual and primal variables are then easily obtained as*

$$\begin{aligned} \lambda &= (M^\top - \Sigma L^\top)^{-1}(\Sigma\, b + |\bar{\Sigma}|\tilde{\lambda})\,, \\ x &= -Q^{-1}(a + Z^\top\lambda)\,, \\ z &= (M - L\Sigma)^{-1}(Zx + c)\,, \end{aligned}$$

which must yield $|\bar{\Sigma}|z = 0$ *up to round off. After that has been checked the nonbasic components of* z *can be set exactly to zero, while* λ *is generally dense without any sign constraints.*

Proof. Firstly we modify the middle block in the last row of Eq. (13) by dropping the post factor $|\Sigma|$ from M to obtain

$$\begin{bmatrix} Q & 0 & Z^\top \\ 0 & |\bar{\Sigma}| & \Sigma L^\top - |\Sigma| M^\top \\ Z & L\Sigma - M & 0 \end{bmatrix} \begin{bmatrix} x \\ z \\ \lambda \end{bmatrix} = - \begin{bmatrix} a \\ \Sigma\, b \\ c \end{bmatrix}. \tag{19}$$

The solution is unchanged since the middle row enforces that $z = |\Sigma|z$ anyhow. Using Q to eliminate the block Z on the bottom left we obtain

$$\begin{bmatrix} Q & 0 & Z^\top \\ 0 & |\bar{\Sigma}| & \Sigma L^\top - |\Sigma| M^\top \\ 0 & L\Sigma - M & -ZQ^{-1}Z^\top \end{bmatrix} \begin{bmatrix} x \\ z \\ \lambda \end{bmatrix} = - \begin{bmatrix} a \\ \Sigma\, b \\ \tilde{c} \end{bmatrix}. \tag{20}$$

where $\tilde{c} = c - ZQ^{-1}a$. Now we can use the triangular block in the middle of the last row to eliminate the second diagonal block yielding the permuted block triangular system

$$\begin{bmatrix} Q & 0 & Z^\top \\ 0 & 0 & \Sigma L^\top - |\Sigma| M^\top + |\bar{\Sigma}|(L\Sigma - M)^{-1} Z Q^{-1} Z^\top \\ 0 & L\Sigma - M & -ZQ^{-1}Z^\top \end{bmatrix} \begin{bmatrix} x \\ z \\ \lambda \end{bmatrix} = - \begin{bmatrix} a \\ \tilde{b} \\ \tilde{c} \end{bmatrix}. \tag{21}$$

where $\tilde{b} = \Sigma\, b - |\bar{\Sigma}|(L\Sigma - M)^{-1}\tilde{c}$. Hence, the main task is to solve the system

$$\left[\Sigma L^\top - |\Sigma| M^\top + |\bar{\Sigma}|(L\Sigma - M)^{-1} Z Q^{-1} Z^\top\right] \lambda \;=\; -\tilde{b}$$

for $\lambda \in \mathbb{R}^s$. Using $\Sigma L^\top - |\Sigma| M^\top = |\Sigma|(\Sigma L^\top - M^\top)$ we obtain by factoring out

$$\left[|\Sigma| + |\bar{\Sigma}|(L\Sigma - M)^{-1} Z Q^{-1} Z^\top (\Sigma L^\top - M^\top)^{-1}\right] (\Sigma L^\top - M^\top)\lambda \;=\; -\tilde{b}\,,$$

which yields after switching the order of $L\Sigma$ and M so that the unitary M is leading

$$\left[|\Sigma| + |\bar{\Sigma}|(M - L\Sigma)^{-1} Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}\right] \tilde{\lambda} \;=\; \tilde{b}\,, \tag{22}$$

where $\tilde{\lambda} \equiv (M^\top - \Sigma L^\top)\lambda$. This $s \times s$ system is permuted block triangular with the two diagonal blocks being symmetric and hopefully positive definite. Projecting by $|\Sigma|$ yields $|\Sigma|\tilde{\lambda} = |\Sigma|\tilde{b}$ and projecting by $|\bar{\Sigma}|$ yields

$$|\bar{\Sigma}|(M - L\Sigma)^{-1} Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}(|\bar{\Sigma}| + |\Sigma|)\tilde{\lambda} = |\bar{\Sigma}|\tilde{b}\,,$$

so that finally after moving the expression in $|\Sigma|\tilde{\lambda}$ to the right hand side

$$|\bar{\Sigma}|(M - L\Sigma)^{-1} Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}|\bar{\Sigma}|\tilde{\lambda} = \tilde{\tilde{b}}\,,$$

where using the definitions of $\tilde{b}$ and $\tilde{c}$ we get in terms of the original vectors $b, c \in \mathbb{R}^s$ and $a \in \mathbb{R}^n$

$$\begin{aligned}
\tilde{\tilde{b}} &= (I - |\Sigma|)\left[\tilde{b} - (M - L\Sigma)^{-1} Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}|\Sigma|\tilde{b}\right] \\
&= |\bar{\Sigma}|\left[(L\Sigma - M)^{-1}\tilde{c} - (M - L\Sigma)^{-1} Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}|\Sigma|\tilde{b}\right] \\
&= |\bar{\Sigma}|(M - L\Sigma)^{-1}\left[c - Z Q^{-1} a - Z Q^{-1} Z^\top (M^\top - \Sigma L^\top)^{-1}\Sigma\, b\right] \\
&= |\bar{\Sigma}|(M - L\Sigma)^{-1}\left[c - Z Q^{-1}(a + Z^\top (M^\top - \Sigma L^\top)^{-1})\Sigma\, b\right]\,.
\end{aligned}$$

This completes the proof since the expressions for x and z can be read off directly from the original system Eq. (13). □

Thus we see that ultimately we have to solve a symmetric linear system in as many variables as there are active kinks. It must be positive definite wherever LIKQ is satisfied. All other calculations just require forward or backward substitutions on the unitary triangular systems with the hopefully quite sparse matrix $(M–L\Sigma)$ or its transposed. When one has done the initial transformation of Q to $\tilde{Q} = I$ as discussed at the end of Subsec. 1.3, we get $\tilde{x} = -\tilde{a} - \tilde{Z}^\top \lambda$ and $z = (M - L\Sigma)^{-1}(\tilde{Z}\tilde{x} + c)$. The most challenging task for a direct solution is no doubt the Cholesky factorization of the nonzero parts of the matrix

$$(I - |\Sigma|)(M - L\Sigma)^{-1}\tilde{Z}\tilde{Z}^\top (M^\top - \Sigma L^\top)^{-1}(I - |\Sigma|) \quad \text{with} \quad \tilde{Z}\tilde{Z}^\top = Z Q^{-1} Z^\top\,.$$

This matrix is subject to a rank two update whenever the signature vector σ changes. The same holds if Q^{-1} is updated for example by the inverse BFGS formula, but that will happen more rarely, in fact probably at the same time when Z and L change completely. So unless we use the total secant updating strategy employed in [1] for smooth NLOP we can assume Q to be constant. As in the total quasi-Newton approach [1] all this can be done with a *quadratic* effort of order $\mathcal{O}(s^2)$, even if Z and M or L are dense. During the piecewise linear optimization we would usually expect that Q^{-1} is constant so that only the changes in $(I - |\Sigma|)(M - L\Sigma)^{-1}$ must be incorporated. Since the number of nontrivial rows and columns will also change there will be a significant amount of indirect addressing and other overhead.

3.1 Compressed Representation

For any σ, we have the Jacobian

$$Z_\sigma = (M - L\Sigma)^{-1} Z \quad \text{with} \quad Z_0 = Z \ .$$

The sparsity pattern of all Z_σ will be contained in the generic pattern of Z_e which is obtained by ignore any cancellation that may arise in its calculation. Now suppose as described in [10] we use a coloring

$$c : \{1, 2, \ldots, n\} \to \{1, 2 \ldots \rho\} \quad \text{with} \quad \rho \ll n \ ,$$

such that $c(j) = c(k)$ implies that the j-th and k-th column of Z_e are structurally orthogonal, i.e., have no nonzero entries in any one of the s rows. In other words c is a coloring of the corresponding column incidence graph of Z_e. Then we can define the seed matrix

$$S = (e_{c(j)}^\top)_{j=1\ldots n} \quad \in \quad \mathbb{R}^{n \times \rho} \tag{23}$$

such that all Z_σ can be reconstructed from the compressed Jacobian $Z_\sigma S$. Moreover, we have

$$Z_\sigma S = (M - L\Sigma)^{-1} ZS \ = \ (M - L\Sigma)^{-1}(ZS) \quad \text{with} \quad ZS \in \mathbb{R}^{s \times \rho} \ .$$

The row-compressed matrix ZS can of course be evaluated directly by the AD tool in use once and for all as we set up the normal form, before any active signatures are considered. Due to the compression the rows of ZS are hopefully reasonably dense and should be accessed and manipulated as vectors. To compute the $Z_\sigma S$ we factorize the unitary lower triangular matrix $M - L\Sigma$ using $\tilde{M} = M - I - L\Sigma$ as

$$M - L\Sigma = (I + e_2 e_2^\top \tilde{M})(I + e_3 e_3^\top \tilde{M}) \cdot \cdots \cdot (I + e_{s-1} e_{s-1} \tilde{M})(I + e_s e_s^\top \tilde{M})$$

By the Sherman Morrison Woodbury formula [22] we have $(I + e_i e_i^\top \tilde{M})^{-1} = I - e_i e_i^\top \tilde{M}$ so that we get

$$Z_\sigma S = (I - e_s e_s^\top \tilde{M})(I - e_{s-1} e_{s-1} \tilde{M}) \cdot \cdots \cdot (I - e_3 e_3^\top \tilde{M})(I - e_2 e_2^\top \tilde{M}) ZS$$

This is just the forward vector mode of algorithmic differentiation with the signature σ fixed. The multiplication of a $s \times \rho$ matrix $\tilde{Z}$ by any one of the $(I - e_i \tilde{m}_i^\top)$ with $\tilde{m}_i = e_i^\top \tilde{M}$ means that for any one of the $\tilde{m}_{i,j} \neq 0$ this number times the j-th row $e_j^\top \tilde{Z}$ is subtracted from the i-th row $e_i^\top Z$. Such a SAXPY operation is efficient on the continuous row vectors and warrants to look up the next j with nonzero $\tilde{m}_{i,j}$ based on a compressed row storage version of M, L and thus $\tilde{M}$. Note that we have to make a difference between *row compression* in the sense of AD seeding, applied to Z_σ and the classical row compressed storage format used in sparse linear algebra.

3.2 Low rank updating

Now suppose we have $Z_\sigma S$ and wish to change σ by adding or subtracting e_i, which is the only operation required by our active signature algorithm. Then we have the rank one update

$$M - L(\Sigma \pm e_i e_i^\top) = M - L\Sigma \mp \ell_i e_i^\top \quad \text{with} \quad \ell_i = Le_i \; .$$

That implies for the inverse again by the Sherman Morrison Woodbury formula that

$$(M - L\Sigma \pm \ell_i e_i^\top)^{-1} \;=\; (M - L\Sigma)^{-1} \mp (M - L\Sigma)^{-1} \ell_i e_i^\top (M - L\Sigma)^{-1} \; .$$

Thus we get that for $\tilde{\sigma} = \sigma \pm e_i$

$$Z_{\tilde{\sigma}} S \;=\; Z_\sigma S \mp \tilde{\ell}_i e_i^\top Z_\sigma S \quad \text{for} \quad \tilde{\ell}_i = (M - L\Sigma)^{-1} \ell_i \tag{24}$$

This means that $\tilde{l}_{ij}$ times the i-th row of $Z_\sigma S$ is subtracted from its j-th row, naturally only if the multiplier is nonzero. So we can originally compute $Z_\sigma S$ for a given σ from Z and subsequently update it always in the compressed row format fully exploiting the sparsity of M and L. All these operations are applied to an $s \times \rho$ matrix where ρ is maximally equal to n.

For each row of Z_e we have an index set $\mathcal{J}_i \subset \{1 \ldots n\}$ containing the column indices of the nonzero entries occurring in row i. Ordering the j in increasing order we may think J_i as a vector and gain some efficiency in various procedures. In particular we can reconstruct the complete i-th row of Z_σ monotonically as

$$e_i^\top Z_\sigma \;=\; (e_i^\top Z_\sigma S e_{c(j)} e_j^\top)_{j \in \mathcal{J}_i} \; . \tag{25}$$

In other words the $(i, c(j))$-th entry of $Z_\sigma S$ becomes the (i, j)-th entry of Z_σ. Of course Z_σ is a generally much larger $s \times n$ matrix that we want to avoid dealing with as a whole. Given σ our main task is to solve a a linear system in the normal matrix

$$H_\sigma \;\equiv\; \left(e_i^\top Z_\sigma Q^{-1} Z_\sigma^\top e_j\right)_{\sigma_j \neq 0}^{\sigma_i \neq 0} \;\in\; \mathbb{R}^{\|\sigma\|_1 \times \|\sigma\|_1} \quad .$$

which will usually done via a Cholesky factorization $H_\sigma = C_\sigma C_\sigma^\top$. There are well established procedures for updating the Cholesky factorization to incorporate the low rank changes caused in the transition from σ to $\tilde{\sigma} = \sigma \pm e_i$, which also means reducing H_σ by one row and column, if $\tilde{\sigma}_i = 0$, and adding one of each otherwise. However, notice that by Eq. (24) these are not the only changes but generally all rows and columns of Z_σ and thus H_σ are modified. Of course all this can be done, although the coding may be a little tedious, especially if H_σ is not repacked but handled by indirect addressing.

3.3 Compressed Approximations of the Normal Hessian

Assuming for the time being that $Q = qI$ and that $q = 1$ without any further loss of generality, we could use either one of the approximations

$$\left(e_i^\top Z_\sigma S S^\top Z_\sigma^\top e_j\right)_{\sigma_j \neq 0}^{\sigma_i \neq 0} \;\approx\; H_\sigma \;\geq\; \left(e_i^\top Z_\sigma S D^{-1} S^\top Z_\sigma^\top e_j\right)_{\sigma_j \neq 0}^{\sigma_i \neq 0} \tag{26}$$

where $D = \mathrm{diag}(S^\top S) \geqslant I$. Notice that these matrices are defined directly in terms of $Z_\sigma S$ and thus do not need any unpacking of the rows according to Eq. (25). Since the rows of $Z_\sigma S$ and Z_σ contain in their corresponding rows exactly the same elements, the diagonal entries in the matrix on the right hand side are equal to those of H_σ. And of course we still have a Haar matrix so the approximation should not be too bad. With respect to the approximation on the right we note that the columns of S are by definition orthogonal and the scaling by $D^{-1/2}$ from the left makes them orthonormal so that $SD^{-1}S^\top$ is an orthogonal projection and what is missing is the complement $Z_\sigma(I - SD^{-1}S^\top)Z_\sigma^\top$, which is of course also positive semidefinite. Updating the Cholesky factorization of either of them is potentially much simpler that doing that for H_σ itself. It is not clear whether a similar approximate Hessian compression could be applied in combination with Newsam Ramsdell seeding as described in [9, 23].

3.4 Unaccumulated approach and cost estimate per pivot

In some case it may be more efficient to never even accumulate the matrices representing the abs-normal form. Calculating $e_i^\top Z_\sigma$ requires just a partial reverse sweep through the linearized graph of the underlying nonlinear function $z = F(x, |z|)$ and computing $\tilde{\ell}_i$ as defined in Eq. (24) corresponds to a partial forward sweep on the same graph with the signatures σ_i fixed accordingly. The number of operations needed for the couple is at most equal to the number of edges in the graph and thus the cost of a single reverse sweep for a conventional gradient calculation. Similarly the right hand side (18) of the reduced equation (17) can be calculated using just a couple of reverse sweeps. Hence, the total number of signature changes can be compared with the number of generalized gradient evaluations in other methods. Sparsity is then exploited implicitly and we can assume that the vectors that the actual algorithm is dealing with are dense.

Rather than using an extended AD tool the user may also slightly modify his oracle code `oracle(x, y, g)`, whose availability or easy derivability is assumed in much of the nonsmooth optimization literature. Let us assume that the forward part evaluating $y \in \mathbb{R}$ as a function φ of $x \in \mathbb{R}^n$ is written without branches in the control flow such that all nonsmoothness arises in the form of $\mathrm{abs}(\cdot)$, $\max(\cdot\,,\cdot)$, $\min(\cdot\,,\cdot)$ and the positive part function $\mathrm{pos}(\cdot) = \max(0\,,\cdot)$. Let us furthermore be optimistic and assume that only on a measure zero set of points $x \in \mathcal{C} \subset \mathbb{R}$ any one of the four kink

functions is critical, i.e., has a zero argument in the case of $\mathrm{abs}(\cdot)$ or $\mathrm{pos}(\cdot)$ and a tie between its arguments in the case of $\max(\cdot\,,\cdot)$ or $\min(\cdot\,,\cdot)$. Then it follows immediately that on the open complement $\mathcal{D} = \mathbb{R}^n \backslash \mathcal{C}$ the code for the function φ is continuously differentiable and we will assume that the programmer or possibly the AD tool that generated the oracle got that right. Now let us redefine all scalar variables to so called `doublets` consisting of a normal value and a directional derivative, which can be propagated simultaneously forward in the usual control flow. Then during the evaluation at a point $x \in \mathbb{R}^n$ we can let the kink functions sequentially record the signs of their arguments and write them into the vector σ. For simplicity let us only consider an absolute value $v = \mathrm{abs}(u)$. To compute the gradient g the oracle function will apply the chain rule in one way or another substituting $\mathrm{sgn}(u)$ as local partial derivative $\partial v / \partial u$. In order to evaluate the vectors $e_i^\top Z_\sigma$, which are adjoint vectors and the $\tilde{\ell}_i$ defined in Eq. (24), which are second order adjoints, we must be able to perform the gradient calculation at the same point x with the same smooth partial derivatives, but as partial of the $\mathrm{abs}(\cdot)$ we must be able to substitute an arbitrary signature $\{-1, 0, 1\}$ as specified by the corresponding component of the vector σ, which is now prescribed by the optimization algorithm. The other kinks can be handled in a similar fashion, or expressed from the beginning in terms of $\mathrm{abs}(\cdot)$. This instrumentation of the oracle code is rather basic and will usually not effect, whatever smart things the programmer has done to minimize storage and complexity in the oracle code. Hence, the application of the optimization approach here to a piecewise linear problem should be rather easy when the application of oracle-based methods is already feasible.

4 Numerical results

To illustrate the algorithm proposed in this paper, we implemented Alg. 1 in Matlab and considered four piecewise linear test problems. To analyse the performance of the new algorithm, we provide comparisons with the nonsmooth optimization routines MPBNGC, i.e., a proximal bundle method described in [20], and the quasi-Newton type method HANSO described in [19] for three of the test problems.

Example 4.1 (HUL)**.** *Hiriart-Urruty and Lemaréchal highlighted in [16] the piecewise linear and convex function* $\varphi : \mathbb{R}^2 \to \mathbb{R}$*,*

$$\begin{aligned} \varphi(x) &= \max\{\varphi_0(x), \varphi_{-5}(x), \varphi_5(x), \varphi_{-2}(x), \varphi_2(x)\} \qquad \textit{with} \\ \varphi_0(x) &= -100, \quad \varphi_{-5}(x) = 2x_1 - 5x_2, \quad \varphi_5(x) = 2x_1 + 5x_2 \\ \varphi_{-2}(x) &= 3x_1 - 2x_2, \quad \varphi_2(x) = 3x_1 + 2x_2\ . \end{aligned}$$

For the formulation in abs-normal form, we use the mathematically equiva-

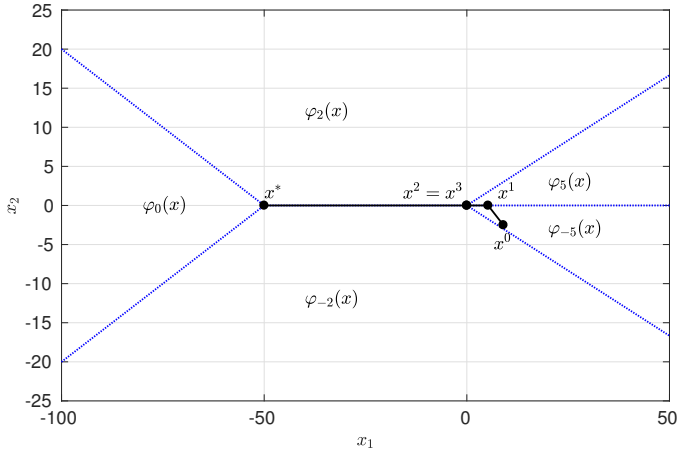

Figure 1: Signature pivots generated by Alg. 1 for HUL example

lent description

$$\varphi(x) = \max\{\max\{-100, 2x_1 + 5|x_2|\}, 3x_1 + 2|x_2|\} \tag{27}$$

that requires only three switching variables, i.e., one has $n = 2$ *and* $s = 3$. *Starting from* $x_0 = (9, -2.5)$, *it was shown in [16] that a steepest descent approach with exact line search leads to a Zenon-like zigzagging with convergence to the even nonstationary point* $\bar{x} = (0, 0)$. *Using Alg. 1, four signature changes are needed as stated Tab. 1, that also illustrates the deactivation and activation of switches during the optimization. Fig. 1 illustrates the convergence history, where the dotted lines represent the points where the target function is not differentiable. As can be seen, the structure of these kinks is exploited by the iterates. It was found in [2] that MPBNGC needed*

Signature Pivot	x_i	$\sigma(x_i)$
0	(9.00, −2.50)	(−1 1 1)
1	(5.25, 0.00)	(0 1 1)
2	(0.00, 0.00)	(0 1 0)
3	(0.00, 0.00)	(0 1 −1)
4	(−50.00, 0.00)	(0 0 −1)

Table 1: Signature pivots and iterates for Rosenbrock-Nesterov II example

6 iterations and HANSO 3 iterations to compute the solution for this test problem.

Example 4.2 (Goffin)**.** *In [5], Goffin proposed the piecewise linear and convex test function given by*

$$\varphi : \mathbb{R}^{50} \to \mathbb{R}, \quad \varphi(x) = \max_{1 \leqslant i \leqslant 50} x_i + \sum_{i=1}^{50} x_i, \qquad \text{with} \qquad x_{0,i} = i - 25.5 \; . \tag{28}$$

For the abs-normal form of this function, one obtains $s = 49$ when $n = 50$. Using Alg. 1, the optimal point $x_* = 0 \in \mathbb{R}^{50}$ is reached in 50 signature pivots. It was found in [2] that the bundle method MPBNGC took 65 iterations and the adapted BFGS version HANSO needed 745 iterations to determine the minimizer. Our previous implementation took six outer iterations using 2030 inner iterations the majority of which was needed to determine a descent direction in a bundle like fashion.

Example 4.3 (Rosenbrock-Nesterov II)**.** *According to [14] Nesterov suggested three Rosenbrock-like test functions, where the first one is smooth, the second one is piecewise smooth and the third one defined as*

$$\varphi : \mathbb{R}^n \to \mathbb{R}, \quad \varphi(x) = \tfrac{1}{4} \, |x_1 - 1| \; + \sum_{i=1}^{n-1} |x_{i+1} - 2|x_i| + 1| \tag{29}$$

is piecewise linear. All three target functions are nonconvex. The two nonsmooth variants were carefully investigated in [14]. All three functions have the unique global minimizer $x = (1, 1, \ldots, 1) \in \mathbb{R}^n$ but the piecewise linear one has $2^{n-1} - 1$ other stationary points, at which nonsmooth optimization algorithms may get stuck even though none of them is a local minimizer. Numerical test showed that with the initial point

$$x_{0,1} = -1, \qquad x_{0,i} = 1 \quad 2 \leqslant i \leqslant n \; ,$$

it is very likely to encounter the numerous stationary points, see again [14]. This is indeed the case when applying Alg. 1, since exactly 2^n signature pivots are needed for $n = 1, \ldots, 10$ to determine the unique minimizer $x_* = (1 \ldots 1) \in \mathbb{R}^n$ as illustrated the row marked with S in Table 2. It was shown in [3] that as somewhat ad hoc defined "reflection" algorithm took exactly the same number of iteration, but each of them required the solution of a separate QP. Table 2 also lists the iterations counts obtained for HANSO (H) and MPBNGC (M) where the star as superscript of the iteration count marks that the iteration terminates not at the minimizer but at a stationary point.

Example 4.4 (Lasso)**.** *Given a matrix $A \in \mathbb{R}^{m \times n}$, a vector $d \in \mathbb{R}^m$ and a weight $\rho \geqslant 0$ consider the problem*

$$\min \tfrac{1}{m} \|Ax - d\|_2^2 + \rho \|x\|_1 \iff \min -\tfrac{2}{m} d^\top A x + \rho e^\top |x| + \tfrac{1}{m} x^\top A^\top A x \; ,$$

n	1	2	3	4	5	6	7	8	9	10
S	2	4	8	16	32	64	128	256	512	1024
H	3	61	494*	1341*	2521*	329*	357*	326*	307*	515*
M	3	52	9859	9978*	3561*	4166*	2547*	1959*	9420*	9807*

Table 2: Signature pivots and iterates for Rosenbrock-Nesterov II example

where $e = (1, \ldots, 1) \in \mathbb{R}^n$. *It represents the so-called Least Absolute Shrinkage and Selection Operator (LASSO) introduced in [24] as a regression approach to perform both variable selection and regularization to enhance the prediction accuracy and interpretability of the statistical model it produces. Since clearly* $s = n$, *the corresponding abs-normal form equals*

$$Z = I, \quad L = 0, \quad c = 0, \quad a^\top = -d^\top A \in \mathbb{R}^n, \quad b^\top = \rho e \in \mathbb{R}^n, \; Q = \tfrac{2}{m} A^\top A \; .$$

Hence, given any $\sigma \in \{-1, 0, 1\}^s$, *we have to solve the linear system*

$$|\bar{\Sigma}| \left(\tfrac{2}{m} A^\top A\right)^{-1} |\bar{\Sigma}| \tilde{\lambda} \;=\; -|\bar{\Sigma}| \left(\tfrac{2}{m} A^\top A\right)^{-1} (A^\top d - \rho \Sigma e)$$

yielding immediately

$$\lambda = \rho |\Sigma| e + |\bar{\Sigma}| \tilde{\lambda} \quad \textit{and} \quad x = \left(\tfrac{2}{m} A^\top A\right)^{-1} (A^\top d - \lambda) \; .$$

Now the feasibility test on $x = z$, *namely whether* $\Sigma x \geqslant 0$ *is easy and cutting back the step to pick up a new kink also. Having found a stationary* x_σ *we can then test whether for some* k

$$0 \;>\; \rho - |\lambda_k| \quad \textit{and} \quad \sigma_k = 0$$

in which case the k*-th kink can be released with the sign of* λ_k. *Naturally we choose the* λ_k *with the largest absolute value to maximize the violation of the normal growth condition. The natural starting point would be the least squares solution* $x_0 = (A^\top A)^{-1} A^\top d$ *with the corresponding* $\sigma = \sigma(z_0) = \sigma(x_0)$.

To test Alg. 1 also for this class of problems, we implemented an adapted version of Alg. 1 as described above. As example problem, we considered the prostate cancer data set already proposed in [24] with $m = 68$, $n = 8$, *and* $\rho = 1.2$. *The optimal solution is computed using eight signature pivots as illustrated also in Fig. 2. Here, the goal is to generate a sparse solution* x_*. *For the given setting, this is obtained since only the components* $x_{*,1}$, $x_{*,2}$, *and* $x_{*,8}$ *are nonzero. With* $A = qI$ *for* $q \geqslant \|A^\top A\|$ *the method is equivalent to a conventional proximal point method and takes almost a thousand steps to reach a solution accuracy of* 10^{-3}.

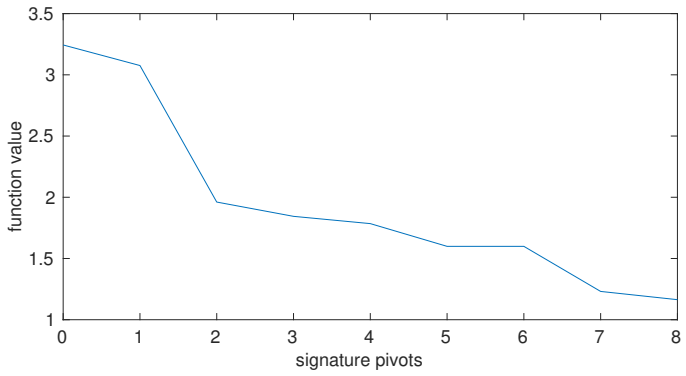

Figure 2: Function values of iterates generated by Alg. 1 for the lasso example

5 Summary, Conclusion, and Outlook

In this paper we have developed a new method for solving piecewise linear optimization problems with a quadratic regularization term. It is remarkably similar to active set strategies for convex quadratic optimization problems, except that the basic switching variables have a variable sign constraint. The activity is encoded in a signature vector $\sigma \in \{-1, 0, 1\}^s$ whose zeros fix the nonbasic switching variables and whose nonzeros constrain the sign of the basic ones. The linear algebra can be organized such that the computational effort and the data traffic per iteration is at worst quadratic in the dimensions and can exploit sparsity of the matrices defining the absnormal form.

The algorithm alternates between a phase where it selects more and more nonbasics until a stationary point is reached and a phase where it releases one of them with a sign that promises the greatest reduction and leads to a neighboring polyhedron. Typically, there is a one step alternation between the two phases. In any case, the algorithm terminates after a finite number of iterations at a local minimizer of the quadratically regularized piecewise linear system. At least for small and moderately sized problems the proposed scheme appears to be fairly optimal.

As an inner loop of Successive Piecewise Linear Optimization (SPLOP) the new algorithm guarantees that all cluster points x_* of the outer iteration are first order minimal in that their local piecewise linearization at each x_* is minimal at this point. Moreover, as shown on [12] the rate or convergence is linear under LIKQ and SSC with strict complementarity, and it is even quadratic in the case of sharp minimizers without any kink qualifications. In this nonlinear context the new algorithm has significant warm start capabilities, especially with respect to the signature vector σ.

The theoretical results are verified on four test problems, one of which involves a nontrivial M matrix, which was restricted to be simply the identity in the original version of the abs-normal form in [7]. The utility of this generalization is illustrated on two classes of problems described in the appendix. On the very limited test set the new active signature strategy yields more numerical stability and a drastic reduction in the number of gradient evaluations or their equivalents, compared to our previous implementation, which was in some sense a bundle method [17]. Throughout the actual calculations reported in Sec. 4 for the first three examples the quadratic regularization term was of the form $Q = qI$ and in the algorithmic design we still assume that Q is positive definite.

The natural next steps are first to implement the structure exploiting linear algebra sketched in Section 3 as compilable code. Then we should allow iterative secant updates of Q and thus to achieve superlinear convergence for SPLOP under suitable regularity conditions. It should be noted that the Hessian to be approximated, which arises in the second order optimality conditions of [11] and [12], is completely smooth and thus bounded on compact level sets. Hence the approximating Hessians need not blow up, as they must when BFGS is directly applied to the discontinuous gradients as in [19]. In a second step one may require Q to be positive definite only on the nullspace of the active constraint Jacobian and to generalize the inertia controlling step selection [27] familiar from successive quadratic optimization (SQOP). Of course, with and without this generalization, the efficient implementation of the numerical linear algebra aspect warrants further attention, especially with regards to the possibility of indeterminacy and cycling in certain degenerate situations. Also, ADOL-C [26] and other AD tools must be taught to evaluate not only Z, L and the vectors a, b, c but also the lower triangular part of M. Finally, in view of really large problems, e.g., from machine learning, a sparse data implementation, possibly applying limited memory BFGS to Q, must be developed. In contrast to partial and even stochastic gradient methods, SPLOP with the proposed inner loop algorithm cannot get stuck at merely (Clarke) stationary points, as demonstrated in [3] on the piecewise linear version of chained Rosenbrock analyzed in [14]. This possibility appears to be a significant concern in machine learning [25].

References

[1] M. Korzec A. Griewank, A. Walther. Maintaining factorized kkt systems subject to rank-one updates of hessians and jacobians. *Optimization Methods and Software*, 22(2):279–295, 2007.

[2] S. Fiege. *Minimization of Lipschitzian Piecewise Smooth Objective Functions*. PhD thesis, Universität Paderborn, 2017.

[3] S. Fiege, A. Walther, and A. Griewank. An algorithm for nonsmooth optimization by successive piecewise linearization. Technical Report SPP1962-007, Universität Paderborn, 2016. available at optimization-online.

[4] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. *Journal of Machine Learning Research*, 15, 2011.

[5] J.-L. Goffin. Subgradient optimization in nonsmooth optimization (including the soviet revolution). *Documenta Mathematica*, Extra Vol.:277–290, 2012.

[6] A. Griewank. Automatic directional differentiation of nonsmooth composite functions. In *Recent developments in optimization. 7th French-German conference on optimization, Dijon, France, June 27-July 2, 1994*, pages 155–169. Springer, 1995.

[7] A. Griewank. On stable piecewise linearization and generalized algorithmic differentiation. *Optimization Methods and Software*, 28(6):1139–1178, 2013.

[8] A. Griewank, J.-U. Bernt, M. Randons, and T. Streubel. Solving piecewise linear equations in abs-normal form. *Lin. Algebra and its Appl.*, 471:500–530, 2015.

[9] A. Griewank and N. Strogies. Efficient evaluation of sparse jacobians by matrix compression, Part I: Motivation and theory. In *Proceeding of the 2nd International Conference Onmathematics, Computational and Statistical Sciences*, pages 33–40, 2014.

[10] A. Griewank and A. Walther. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. SIAM, 2008.

[11] A. Griewank and A. Walther. First and second order optimality conditions for piecewise smooth objective functions. *Optimization Methods and Software*, 31(5):904–930, 2016.

[12] A. Griewank and A. Walther. Relaxing kink qualifications and proving convergence rates in piecewise smooth optimization. Technical report, Yachay TEch, 2017. available at optimization-online.

[13] A. Griewank, A. Walther, S. Fiege, and T. Bosse. On Lipschitz optimization based on gray-box piecewise linearization. *Mathematical Programming Series A*, 158(1-2):383–415, 2016.

[14] M. Gürbüzbalaban and M.L. Overton. On Nesterov's nonsmooth Chebyshev-Rosenbrock functions. *Nonlinear Anal: Theory, Methods & Appl.*, 75(3):1282–1289, 2012.

[15] L. Hascoët and V. Pascual. The Tapenade automatic differentiation tool: Principles, model, and specification. *ACM Transactions on Mathematical Software*, 39(3):20:1–20:43, 2013.

[16] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I.* Springer, 1993.

[17] N. Karmitsa and M. Mäkelä. Limited memory bundle method for large bound constrained nonsmooth optimization: convergence analysis. *Optimization Methods and Software*, 25(6):895–916, 2010.

[18] V. Klee and G.J. Minty. How good is the simplex algorithm? Inequalities III, Proc. 3rd Symp., Los Angeles 1969, 159-175., 1972.

[19] A. Lewis and M. Overton. Nonsmooth optimization via quasi-Newton methods. *Mathematical Programming Series A*, 141(1-2):135–163, 2013.

[20] M.M. Mäkelä and P. Neittaanmäki. *Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control.* World Scientific Publishing Co., 1992.

[21] B. Mordukhovich and M. Sarabi. Critical multipliers in variational systems via second-order generalized differentiation. *Mathematical Programming*, 4(2), 2017.

[22] J. Nocedal and S. Wright. *Numerical Optimization.* Springer Series in Operations Research. Springer, 2006.

[23] N. Strogies and A. Griewank. Efficient evaluation of sparse jacobians by matrix compression, Part II: Implementation and experiments. *WSEAS Transactions on Mathematics*, 13:646–653, 2014.

[24] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288, 1996.

[25] N. Tripuraneni, M. Stern, C. Jin, J. Regier, and M. Jordan. Stochastic cubic regularization for fast nonconvex optimization. Technical Report 1711.02838, arXiv, 2017.

[26] A. Walther and A. Griewank. *Combinatorial Scientific Computing*, chapter Getting Started with ADOL-C, pages 181–202. Chapman-Hall CRC Computational Science, 2012.

[27] E. Wong. *Active-Set Methods for Quadratic Programming.* PhD thesis, University of California, San Diego, 2011.

6 Appendix

Here we consider two classes of examples in order to motivate the generalization from I to a general unitary lower triangular matrix M.

6.1 The max function, an instructive example

In parts of the nonsmooth analysis literature, particular attention is devoted to the case of convex piecewise linear functions, see, e.g., [21]. It is well understood that any globally defined convex piecewise linear function is continuous and can be represented in the simple max form

$$\varphi(x) \;=\; \max_{0 \leqslant i \leqslant s} (a_i^\top x - \beta_i) \quad \text{with} \quad a_i \in \mathbb{R}^n, \beta_i \in \mathbb{R} \quad \text{for} \quad 0 \leqslant i \leqslant s \; .$$

Obviously, its evaluation in one way or another must involve s comparisons. We will consider the simple loop

$$\gamma_0 = a_0^\top x - \beta_0 \; , \qquad \gamma_i = \max(\gamma_{i-1}, a_i^\top x - \beta_i) \quad \text{for} \quad 1 \leqslant i \leqslant s \; , \qquad y = \gamma_s \; .$$

In terms of absolute values the loop can be rewritten as

$$\gamma_i = a_i^\top x - \beta_i + \tfrac{1}{2} z_i + \tfrac{1}{2}|z_i| \quad \text{with} \quad z_i = \gamma_{i-1} - a_i^\top x + \beta_i \quad \text{for} \quad 1 \leqslant i \leqslant s \; .$$

Expressing γ_i in terms of z_{i+1} we obtain for $i = 0, \ldots, s-1$

$$z_{i+1} = -\tilde{a}_{i+1}^\top x + \tilde{\beta}_{i+1} + \tfrac{1}{2} z_i + \tfrac{1}{2}|z_i| \quad \text{with} \quad \tilde{a}_{i+1} = a_{i+1} - a_i \text{ and } \tilde{\beta}_{i+1} = \beta_{i+1} - \beta_i \; .$$

With $L \in \mathbb{R}^{s \times s}$ the square matrix, whose only nonzero entries are in the $s-1$ subdiagonal positions with constant value $\frac{1}{2}$, we obtain in matrix vector form

$$z = Lz + \tilde{b} - \tilde{A}x + L|z| \qquad \text{with} \qquad \tilde{b} = (\tilde{\beta}_i)_{1 \leqslant i \leqslant s} \in \mathbb{R}^s \text{ and } \tilde{A} = (\tilde{a}_i^\top)_{1 \leqslant i \leqslant s} \in \mathbb{R}^{s \times n} \; .$$

To arrive at the abs-normal form as introduced in [7] we have to solve for z using the lower triangular inverse

$$I + \tilde{L} \equiv (I - L)^{-1} = (c_{ij})_{1 \leqslant j \leqslant s}^{1 \leqslant i \leqslant s} \quad \text{with} \quad c_{ij} = \begin{cases} (\frac{1}{2})^{i-j} & \text{if} \quad i \geqslant j \\ 0 & \text{else} \end{cases} \; .$$

Using $(I - L)^{-1} L = (I - L)^{-1}(L - I + I) = (I - L)^{-1} - I = \tilde{L}$ we obtain finally the standard triangular system

$$z = (I + \tilde{L})(\tilde{b} - \tilde{A}x) + \tilde{L}|z|$$

for the switching variable vector z. Now suppose the problem is localized at x in that all z_i vanish so that $Z = -(I + \tilde{L})\tilde{A}$. Then LIKQ requires exactly that $\tilde{A}$ has full row rank $s \leqslant n$ which turns out to be equivalent to the *affine independence qualification condition* (AIQC) given in [21] for points

in the interior of the domain. The reason is that $-(I + \tilde{L})$ is a nonsingular transformation and that $\tilde{A} = (a_i^\top - a_{i-1}^\top)_{1 \leqslant i \leqslant s} \in \mathbb{R}^{s \times n}$ has full row rank $s \leqslant n$ exactly when the $s + 1$ extended vectors $(a_i^\top, 1) \in \mathbb{R}^{n+1}$ for $0 \leqslant i \leqslant s$ are linearly independent as required by AIQC.

The objective function takes the somewhat complicated form

$$\varphi(x) = a_s^\top x - \beta_s + \tfrac{1}{2} e_s^\top \left[(I + \tilde{L})(\tilde{b} - \tilde{A}x) + \tilde{L}|z| \right] + \tfrac{1}{2}|z_s| \;=\; \alpha + a^\top x + b^\top |z|$$

with

$$\alpha = \tfrac{1}{2} e_s^\top (I + \tilde{L})\tilde{b} - \beta_s, \quad a^\top = a_s^\top - \tfrac{1}{2} e_s^\top (I + \tilde{L})\tilde{A}, \quad b^\top = \tfrac{1}{2} e_s^\top (I + \tilde{L})$$

because we have to include $\frac{1}{2} z_s$ as a contribution to $\gamma_s = y$.

Given the simplicity of the original function definition, the abs-normal form we derived is surprisingly complicated. More specifically, since $\tilde{L}$ is triangular but otherwise dense the naive evaluation in the final form requires an effort of order s^2, whereas the original problem can be quite sparse depending on the structure of A of course. It may also suffer a lot of fill-in in the transition to $\tilde{A}$. One aspect that is similarly troubling is that from the final procedure it is no longer clear that all the z_i like the γ_i are convex functions of the input vector x. This is really a serious objection. One ad hoc possibility to improve things is to generalize the abs-normal representation of the switching variables to

$$Mz = c + Zx + L|z|$$

with M a unitary lower triangular matrix. For monotonicity and convexity arguments it would be nice if M was an M-matrix whose inverse has by definition only positive entries as is the case here. Of course we would also like M to be rather sparse as we do expect from L. The calculation of M could also be effected by ADOL-C and other tools, we just would do less accumulation and thus hopefully less fill-in in calculating the partitioned Jacobian. Now there is still a problem in that in contrast to the positive part function $u_+ = \max(0, u)$ the absolute value $|u|$ maintains convexity only where the argument is positive, which is not guaranteed here. Obviously, this issue needs some (or even) a lot more thoughts and some judgment to what extent we really wish to accommodate (possibly partially) convex problems effectively. All the existing theory can be easily rewritten simply by premultiplying the right hand side components by M^{-1}.

6.2 Modeling a neural network

Finally we consider a neural net with the positive part function taking the place of the usual sigmoidal function as recommended for example in [4]. The transition of one layer to the next can then be written as

$$z_{i+1} \;=\; W_i \, \max(z_i, 0) \;=\; W_i(z_i + |z_i|)/2 \quad \text{where} \quad z_i \in \mathbb{R}^{n_i} \text{ and } W_i \in \mathbb{R}^{n_{i+1} \times n_i} \, .$$

Separating the switching variables and their absolute values we get the recursion

$$z_{i+1} - \tfrac{1}{2} W_i z_i \;=\; \tfrac{1}{2} W_i |z_i| \quad \text{for} \quad i = 1 \ldots s \quad \text{from} \quad z_1 = x \quad \text{with} \quad x \in \mathbb{R}^{n_0} \,.$$

Here the initial layer $z_1 = x$ is the input vector that given the weight matrices W_i yields the output vector $y = z_{s+1} \in \mathbb{R}^{n_s}$. Thus the complete abs-normal is given by the block bidiagonal form

$$\underbrace{\begin{bmatrix} I & 0 & 0 & \ldots\ldots\ldots & 0 & 0 \\ -\frac{1}{2}W_1 & I & 0 & \ldots\ldots\ldots & 0 & 0 \\ 0 & -\frac{1}{2}W_2 & I & \ldots\ldots\ldots & 0 & 0 \\ . & . & . & \ldots\ldots\ldots & 0 & 0 \\ . & . & . & \ldots\ldots\ldots & 0 & 0 \\ 0 & 0 & 0 & \ldots\ldots\ldots & I & 0 \\ 0 & 0 & 0 & \ldots\ldots\ldots & -\frac{1}{2}W_s & I \end{bmatrix}}_{=M} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \ldots \\ \ldots \\ z_s \\ z_{s+1} \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} I \\ 0 \\ 0 \\ \ldots \\ \ldots \\ 0 \\ 0 \end{bmatrix}}_{=Z} x + \underbrace{\begin{bmatrix} 0 & 0 & 0 & \ldots\ldots\ldots & 0 & 0 \\ \frac{1}{2}W_1 & 0 & 0 & \ldots\ldots\ldots & 0 & 0 \\ 0 & \frac{1}{2}W_2 & 0 & \ldots\ldots\ldots & 0 & 0 \\ . & . & . & \ldots\ldots\ldots & 0 & 0 \\ . & . & . & \ldots\ldots\ldots & 0 & 0 \\ 0 & 0 & 0 & \ldots\ldots\ldots & 0 & 0 \\ 0 & 0 & 0 & \ldots\ldots\ldots & \frac{1}{2}W_s & 0 \end{bmatrix}}_{=L} \begin{bmatrix} |z_1| \\ |z_2| \\ |z_3| \\ \ldots \\ \ldots \\ |z_s| \\ |z_{s+1}| \end{bmatrix}$$

Here the constant shift c is zero and obviously multiplying through by the block dense matrix M^{-1} would be a very bad idea in terms of sparsity. Hence we definitely need the generality of having a non-identity as M. Notice that we have not specified a scalar objective function, which typically might be some norm of the discrepancy between the output signal z_{s+1} and given training data. It is also important to note that the weights W_i which are adjusted during training are considered as constant parameters rather than input variables. If x is constant and the W_i are variable we have again a piecewise linear vector function, but the equations above are formally not its abs-normal form. We are currently investigating how piecewise linearity can be exploited for the training of such neural networks.