AN INTERIOR-POINT METHOD FOR NONLINEAR OPTIMIZATION PROBLEMS WITH LOCATABLE AND SEPARABLE NONSMOOTHNESS

MARTIN SCHMIDT

Abstract. Many real-world optimization models comprise nonconvex and nonlinear as well as nonsmooth functions leading to very hard classes of optimization models. In this article a new interior-point method for the special but practically relevant class of optimization problems with locatable and separable nonsmooth aspects is presented. After motivating and formalizing the problems under consideration, modifications and extensions to standard interior-point methods for nonlinear programming are investigated in order to solve the introduced problem class. First theoretical results are given and a numerical study is presented that shows the applicability of the new method for real-world instances from gas network optimization.

1. Introduction

Interior-point (or barrier) methods mainly emerged in the late 1970s and 1980s. They came up due to the search for algorithms solving linear programs (LP) with a better worst-case complexity than the simplex method. The earliest proposed algorithm for solving LPs with a polynomial complexity was the ellipsoid method by Khachiyan [14]. However, this method turned out to be impractical and was not competitive with the simplex method in practice. In 1984, Karmarkar [12] published a new method that shares the complexity property of Khachiyan's algorithm but also has a good practical performance. Since then a lot of research activity came up leading to a wide range of theoretical results and implementations. One decade after Karmarkar's publication a subclass of interior-point methods arose that are known as primal-dual methods. This subclass turned out to be the most efficient instantiation of interior-point methods. Todays most powerful codes of primal-dual type include, e.g., Ipopt [30], KNITRO [31], and LOQO [26, 27].

In contrast to other approaches that were developed for linear or quadratic programming, interior-point methods have the advantage that their main algorithmic framework stays the same when applied to more difficult problem classes like nonconvex quadratic or nonlinear problems. Of course, additional techniques for globalization and handling nonconvexity etc. come into play but the main building blocks of the methods, e.g. computation of the search direction, the fraction-to-the-boundary rule or updating of the barrier parameter are almost identical for all problem classes. This motivates the main idea of this paper—namely to develop an interior-point method for a special class of nonsmooth problems by using as much as possible from the algorithmic building blocks of standard interior-point methods and modifying and extending the algorithm as much as required in order to tackle the more complex class of optimization problems.

Date: October 23, 2014.

2010 *Mathematics Subject Classification.* 90C30, 90C51, 90C90, 90C35, 90C56, 90B10.

Key words and phrases. Interior-point methods, Barrier methods, Line-Search methods, nonlinear and nonsmooth optimization, Classification of optimization models, Gas network optimization.

Interior-point methods are not the standard tool for solving nonsmooth optimization problems. The most prominent class of algorithms in this field are bundle methods which approximate nonsmooth functions by bundles of hyperplanes (see [10, 11] for a detailed description). However, standard bundle methods are only applicable for nonsmooth problems with convex objective function and constraints. The method presented in this paper does *not* possess this restriction. The reader interested in other algorithms and software for nonsmooth optimization is referred to [13].

The contribution of this paper is the following. A new class of practically relevant nonlinear and nonsmooth optimization models is formalized and modeling examples are given to demonstrate how to reformulate constraints that often appear in nonsmooth problems in order to fit into the introduced class. For this class a new interior-point method is presented that is able to solve these models for most of the tested instances. This is realized by modifying the line-search sub-procedure, the computation of the maximum step length and the usage of problem-tailored generalized gradients. For the latter it is proved that these generalized gradients belong to Clarke's generalized gradient. Finally, a stationarity test for problems with locatable and separable nonsmoothness is given. Thus, it is shown that, if converging, the algorithm returns a KKT point for nonsmooth but Lipschitz continuous problems. A complete convergence proof, however, is outstanding and topic of future research. Despite that fact, the presented numerical results on low-dimensional academic and real-world instances from gas transport are very encouraging and show the applicability of the new method.

The paper is organized as follows. In Section 2 a general interior-point framework is presented that is modified and extended in Section 4 in order to solve nonsmooth optimization problems with locatable and separable nonsmoothness. This class of nonsmooth problems is introduced in Section 3. Numerical results on low-dimensional test problems as well as on real-world instances from the field of gas transport network optimization are given in Section 5. The paper concludes with a summary and a discussion of open topics in Section 6.

2. General Interior-Point Framework

We consider interior-point methods for solving the problem

$$\min_{x\in\mathbb{R}^n} \quad f(x) \tag{1a}$$

$$\text{s.t.} \quad c(x) = 0, \tag{1b}$$

$$x \geq 0, \tag{1c}$$

where, for the moment, the objective function $f : \mathbb{R}^n \to \mathbb{R}$ and the constraints $c : \mathbb{R}^n \to \mathbb{R}^m$ are assumed to be smooth, i.e. $\mathcal{C}^2$ in this article. Later, this regularity condition is relaxed in order to consider a more general class of nonsmooth problems. In order to get rid of the inequality constraints (1c), interior-point methods replace (1) by a series of barrier problems

$$\min_{x\in\mathbb{R}^n} \quad \varphi_\mu(x) := f(x) - \mu \sum_{i=1}^{n} \ln x_i \quad \text{s.t.} \quad c(x) = 0,$$

with the so-called barrier parameter $\mu > 0$ that has to be driven to zero. In addition, the non-negativity constraint is strengthened to $x > 0$. As it is well known, this approach is equivalent to applying a homotopy method for the (perturbed) primal-dual system of (1);

$$\nabla f(x) - \nabla c(x)^T \lambda - z = 0, \tag{2a}$$

$$c(x) = 0, \tag{2b}$$

$$XZe - \mu e = 0. \tag{2c}$$

Here and in what follows, $\nabla c(x) \in \mathbb{R}^{m \times n}$ is the Jacobian of the constraint vector c and e is the vector of all ones with appropriate dimension. Capital letters denote diagonal matrices made up of the entries of the corresponding vectors (denoted by small letters). The dual variables $\lambda \in \mathbb{R}^m$ and $z \in \mathbb{R}^n$ correspond to Lagrange multipliers of the equality constraints (1b) and the variable bounds (1c), respectively.

With this at hand, primal-dual methods proceed as follows. Given a primal-dual iterate $y^{(k)} := (x^{(k)}, \lambda^{(k)}, z^{(k)})$ that fulfills the positivity constraints, i.e. $x^{(k)}, z^{(k)} > 0$, the KKT system

$$\begin{bmatrix} H(x^{(k)}) & -\nabla c(x^{(k)})^T & -I \\ \nabla c(x^{(k)}) & 0 & 0 \\ Z^{(k)} & 0 & X^{(k)} \end{bmatrix} \begin{pmatrix} \Delta x^{(k)} \\ \Delta \lambda^{(k)} \\ \Delta z^{(k)} \end{pmatrix} = - \begin{pmatrix} \nabla_x \mathcal{L}(y^{(k)}) \\ c(x^{(k)}) \\ X^{(k)} Z^{(k)} e - \mu^{(k)} e \end{pmatrix} \tag{3}$$

is solved, yielding a primal-dual search direction $(\Delta x^{(k)}, \Delta \lambda^{(k)}, \Delta z^{(k)})$. Here and in what follows, $\nabla_x \mathcal{L}$ denotes the gradient of the Lagrangian of (1), i.e.,

$$\nabla_x \mathcal{L}(y^{(k)}) = \nabla f(x^{(k)}) - \nabla c(x^{(k)})^T \lambda^{(k)} - z^{(k)}$$

and H denotes the Hessian of the Lagrangian.

Since simply applying this direction would potentially violate the positivity constraints, the so-called fraction-to-the-boundary rule is imposed,

$$\bar{\alpha}_{\text{pri}}^{(k)} := \max_{\alpha \in (0,1]} \{x^{(k)} + \alpha \Delta x^{(k)} \geq (\tau - 1) x^{(k)}\}, \tag{4a}$$

$$\bar{\alpha}_{\text{dual}}^{(k)} := \max_{\alpha \in (0,1]} \{z^{(k)} + \alpha \Delta z^{(k)} \geq (\tau - 1) z^{(k)}\}, \tag{4b}$$

where $\tau \in (0, 1)$ is the fraction-to-the-boundary parameter. By restricting all primal and dual step lengths $\alpha_{\text{pri}}^{(k)}$ and $\alpha_{\text{dual}}^{(k)}$ by these upper bounds, positivity of all iterates $x^{(k)}, z^{(k)}$ is guaranteed. For nonlinear problems it is additionally required to determine actual step lengths $\alpha_{\text{pri}}^{(k)} \in (0, \bar{\alpha}_{\text{pri}}^{(k)}], \alpha_{\text{dual}}^{(k)} \in (0, \bar{\alpha}_{\text{dual}}^{(k)}]$ by a globalization approach like trust region or line-search techniques (see, e.g., [1, 6, 24, 25, 27, 29, 30] and the references therein). In this article the focus lies on line-search methods. Here, for a given maximal primal step length $\bar{\alpha}_{\text{pri}}^{(k)}$, trial step lengths $\alpha_{\text{pri}}^{(k,l)}, l = 0, 1, 2, \ldots,$ are generated (typically by $\alpha_{\text{pri}}^{(k,l+1)} = c\alpha_{\text{pri}}^{(k,l)}$ with $c \in (0, 1)$) and the corresponding trial points $x^{(k,l)} = x^{(k)} + \alpha_{\text{pri}}^{(k,l)} \Delta x^{(k)}$ are tested for acceptance by a filter or a merit function.

Finally, the new iterate is given by

$$x^{(k+1)} = x^{(k)} + \alpha_{\text{pri}}^{(k)} \Delta x^{(k)}, \tag{5a}$$

$$\lambda^{(k+1)} = \lambda^{(k)} + \alpha_{\text{dual}}^{(k)} \Delta \lambda^{(k)}, \tag{5b}$$

$$z^{(k+1)} = z^{(k)} + \alpha_{\text{dual}}^{(k)} \Delta z^{(k)}. \tag{5c}$$

In addition, the barrier parameter $\mu^{(k)}$ can be updated, yielding a different barrier parameter $\mu^{(k+1)} \neq \mu^{(k)}$ or can stay the same, i.e. $\mu^{(k+1)} = \mu^{(k)}$. At least, the mechanism for updating has to ensure that the sequence of barrier parameters converges to zero, leading to the (un-)perturbed KKT conditions of the original problem (1). For the details, it is referred to, e.g., [17, 18, 27, 30]. Interior-point methods finally terminate if an KKT point is approximately reached, i.e., if an iterate satisfies

$$e^{(k)} := \max\{\|\nabla_x \mathcal{L}(y^{(k)})\|_\infty, \|c(x^{(k)})\|_\infty, \|X^{(k)} Z^{(k)} e\|_\infty\} \leq \varepsilon \tag{6}$$

for a user-specified tolerance $\varepsilon > 0$. Algorithm 1 states a basic interior-point framework.

Algorithm 1 : Basic Interior-Point Framework

Input : Starting point $(x^{(0)}, \lambda^{(0)}, z^{(0)})$ with $x^{(0)}, z^{(0)} > 0$, initial barrier parameter $\mu^{(0)} > 0$, convergence tolerance $\varepsilon > 0$.

Set $k = 0$.
while $e^{(k)} > \varepsilon$ **do**
  Compute the search direction $(\Delta x^{(k)}, \Delta\lambda^{(k)}, \Delta z^{(k)})$ by solving (3).
  Compute maximum step lengths $\bar{\alpha}_{\text{pri}}^{(k)}, \bar{\alpha}_{\text{dual}}^{(k)}$ using (4).
  Apply a globalization strategy to obtain actual step lengths $\alpha_{\text{pri}}^{(k)}, \alpha_{\text{dual}}^{(k)}$.
  Compute the new iterate $(x^{(k+1)}, \lambda^{(k+1)}, z^{(k+1)})$ using (5).
  Compute a new barrier parameter $\mu^{(k+1)}$ or set $\mu^{(k+1)} = \mu^{(k)}$.
  Set $k \leftarrow k + 1$.

Beside the algorithmic framework that is discussed so far, many improvements can be found in state-of-the-art implementations. To name a few, reduction techniques for the KKT system (3) are applied (see, e.g., [21]), iterative refinement techniques are used to improve the quality of the search directions (see, e.g., [9, 21, 30]), second order corrections are used to improve the globalization approach (see, e.g., [4, 21, 30]), a feasibility restoration phase is incorporated (see, e.g., [6, 28, 30]) and problem scaling is applied (see, e.g., [30]).

Finally, the reader interested in convergence results is referred to [7, 25, 29].

3. Definition of the Problem Class

In Section 2 the assumption was made that the objective function f and the constraints c are twice continuously differentiable. However, many real-world nonlinear optimization problems do not satisfy this regularity assumption. In particular, constraints violating this assumption often include min, max or absolute value functions. These functions have the property that there are only few points at which the required regularity is not given, whereas there are large regions at which they are smooth. One possible approach to tackle this situation—especially in models that include discrete decisions anyway—is to introduce binary variables to identify the (smooth) region to which the current iterate belongs and to select the smooth restriction to this region. This approach obviously increases the difficulty of the problem tremendously by introducing a (possibly large number of) integer variables. The aim of the following sections is to address this problem by a novel approach. The main idea is to start from a standard interior-point method as described in Section 2 and to modify it as little as possible but as much as required in order to be able to solve the more general problem class.

In this section a formal description of the considered class of nonsmooth problems is given, leading to the definition of the class of so-called *optimization problems with separable and locatable nonsmoothness*. The next section then discusses the modifications to the basic interior-point framework of Section 2 that enable the algorithm to tackle this problem class.

First, some basic definitions and results from the field of nonsmooth analysis are collected that are required in the following. The presentation is based in the books [2, 3]. We start with the definition of Clarke's generalized gradient:

Definition 1 (Clarke's Generalized Gradient)**.** *Let X be a real Banach space and $f : X \to \mathbb{R}$ locally Lipschitz-continuous in a neighborhood $\mathcal{N}$ of $x \in X$. Furthermore, let $d \in X$.* Clarke's generalized directional derivative *of f at x in the direction d is defined as*

$$f^\circ(x; d) := \limsup_{\substack{y \to x \\ t \downarrow 0}} \frac{f(y + td) - f(y)}{t}.$$

Clarke's generalized gradient *of f at x is given by*

$$\partial f(x) := \{y \in X^* \colon f^\circ(x; d) \geq \langle y, d\rangle \text{ for all } d \in X\},$$

where X^* denotes the dual space of X and $\langle a, b\rangle, a \in X^*, b \in X$, is the associated dual pairing.

Definition 1 is valid for finite as well as infinite dimensional Banach spaces. The aim of the following is to state a more practicable characterization of $\partial f(x)$ in finite dimension. This is done by the following theorem (see [3] for a proof).

Theorem 1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz-continuous in a neighborhood of $x \in \mathbb{R}^n$ and suppose S to be any set of Lebesgue measure 0 in $\mathbb{R}^n$. Then*

$$\partial f(x) = \operatorname{conv}\left\{\lim_{i\to\infty} \nabla f(x_i) \colon x_i \to x, x_i \notin S, x_i \notin \mathcal{K}\right\}$$

holds, where $\mathcal{K}$ is the set of points at which f fails to be differentiable. As usual, $\operatorname{conv} M$ *denotes the convex hull of the set M.*

To avoid the impractical condition "$x_i \notin S$", Rademacher's theorem is useful:

Theorem 2 (Rademacher, [2])**.** *Let $f : U \subset \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz-continuous on an open set U. Then f is differentiable almost everywhere on U (w.r.t. the Lebesgue measure).*

By applying Rademacher's theorem to Theorem 1 one obtains the following result.

Lemma 1. *For an open subset $U \subset \mathbb{R}^n$ let $f : U \to \mathbb{R}$ be locally Lipschitz-continuous and $x \in U$. Let $\mathcal{K}$ be the set of points at which f fails to be differentiable. Moreover, let $(x_i) \subset \mathbb{R}^n \setminus \mathcal{K}$ be a sequence of points converging to x. Furthermore, assume that $\lim_{i\to\infty} \nabla f(x_i)$ exists. Then $\lim_{i\to\infty} \nabla f(x_i) \in \partial f(x)$.*

Proof. Using Rademacher's theorem it follows that $\mathbb{R}^n \setminus \mathcal{K}$ is not a subset of Lebesgue measure 0. Thus, the lemma follows directly from Theorem 1. □

Next, first-order necessary conditions are stated for (1) with possibly nonsmooth but locally Lipschitz-continuous objective function f and constraints c (cf. [3]).

Theorem 3. *Let $x^* \in \mathbb{R}^n$ be a local solution of (1) with locally Lipschitz-continuous functions f and c. Then there exist Lagrange multipliers $\xi^* \in \mathbb{R}_{\geq 0}, \lambda^* \in \mathbb{R}^m, z^* \in \mathbb{R}^n$, not all zero, such that*

$$0 \in \partial_x \mathcal{L}(x^*, \xi^*, \lambda^*, z^*), \tag{7a}$$

$$0 = c(x^*), \tag{7b}$$

$$0 \leq (x^*, z^*), \tag{7c}$$

$$0 = Z^* X^* e. \tag{7d}$$

Here, $\partial_x \mathcal{L}$ denotes Clarke's generalized gradient of the Lagrangian function with an additional multiplier for the objective function, i.e.

$$\partial_x \mathcal{L}(x^*, \xi^*, \lambda^*, z^*) = \xi^* \partial f(x^*) - \sum_{i=1}^{m} \lambda_i^* \partial c_i(x^*) - z^*.$$

The only differences between the KKT conditions (2) (with $\mu = 0$) for smooth problems and the KKT conditions (7) for nonsmooth constrained problems are

- the generalization from "$=$" to "$\in$" in condition (7a),
- the generalization from standard gradients ∇c_i to Clarke's generalized gradients ∂c_i, and
- the additional Lagrange multiplier ξ of the objective function.

Finally, the definition of a subgradient is given, which is a generalization of the gradient for convex functions in the nonsmooth case.

Definition 2 (Subgradient, Subdifferential)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function. A vector $g \in \mathbb{R}^n$ is called a* subgradient *of f at $x \in \mathbb{R}^n$ if*

$$f(y) \geq f(x) + \langle g, y - x \rangle$$

holds for all $y \in \mathbb{R}^n$. The set of all subgradients of f at x is called the subdifferential *of f at x and is denoted by $\partial f(x)$.*

It can be shown that Clarke's generalized gradient coincides with the subdifferential in the convex case (cf. [3]). Thus, Definition 2 does not lead to a conflict in the notation. The following lemma about subdifferentials of univariate functions is needed later in Section 4.

Lemma 2. *Let $f : \mathbb{R} \to \mathbb{R}$ be a convex function. Then $\partial f(x)$ is a nonempty interval.*

Proof. The Lemma follows directly from the fact that a subdifferential is a nonempty, convex, closed and bounded set (see [20, Theorem 2.74]). As a subset of $\mathbb{R}$, $\partial f(x)$ is a nonempty interval. □

With these concepts at hand, the considered subclass of nonsmooth constrained optimization problems is now defined. This subclass consists of problems whose constraints c are piecewise smooth and locally Lipschitz-continuous. Furthermore, the constraints have to satisfy two additional properties. These properties are the *separable nonsmoothness property* and the existence of so-called *localization functions*. Both will be defined and illustrated in the following.

Definition 3 (Separable Nonsmoothness Property)**.** *Let $c_i(x) = 0$ be an equality constraint with piecewise smooth and locally Lipschitz-continuous c_i. The constraint $c_i(x) = 0$ satisfies the* separable nonsmoothness property *if there exist*

(1) *a single variable $x_{i_\nu}, i_\nu \in \{1, \ldots, n\}$, and a univariate, piecewise smooth and locally Lipschitz-continuous, convex function $\theta_i : \mathbb{R} \to \mathbb{R}$ depending only on $x_{i_\nu} \in \mathbb{R}$ (i.e. $\theta_i = \theta_i(x_{i_\nu})$),*
(2) *a smooth function $\tilde{c}_i : \mathbb{R}^{n+1} \to \mathbb{R}$ depending on $x \in \mathbb{R}^n$ and on an additional auxiliary variable $x_{i_a} \in \mathbb{R}$ (i.e., $i_a \notin \{1, \ldots, n\}$),*

such that $c_i(x) = 0$ can be equivalently restated as

$$\tilde{c}_i(x, x_{i_a}) = 0 \quad \text{with} \quad x_{i_a} \quad \text{subject to} \quad x_{i_a} \pm \theta_i(x_{i_\nu}) = 0. \tag{8}$$

$\tilde{c}_i$ is called a lifting *of c_i.*

The motivation of this definition is to formalize the situation in which the nonsmoothness of a constraint can be shifted into a univariate, piecewise smooth and locally Lipschitz-continuous function. This makes it possible to construct a modified stationarity test (cf. Theorem 3) that can be implemented in an efficient way. The following example illustrates the definition.

Example 1. *Consider the piecewise smooth and locally Lipschitz-continuous constraint* $c_i(x_1, x_2) = 0$ *with*

$$c_i(x_1, x_2) = x_1^2 + \min(x_2, 0) - 42.$$

With

$$x_3 + \theta_i(x_2) = 0, \quad \theta_i(x_2) := -\min(x_2, 0),$$

and

$$\tilde{c}_i(x_1, x_2, x_3) := x_1^2 + x_3 - 42.$$

One can see that c_i *satisfies the separable nonsmoothness property (with* $i_\nu = 2$ *and* $i_a = 3$*).*

The next example lists some functions that are often used for modeling of nonsmooth aspects and that can be used as the function θ_i in Definition 3.

Example 2. *The functions* $\min(x_1, 0), \max(x_1, 0)$ *and the absolute value function* $|x_1|$ *are univariate, piecewise smooth and locally Lipschitz-continuous functions.* $\max(x_1, 0)$ *and* $|x_1|$ *are convex.* $\min(x_1, 0)$ *is concave but fits into the situation of Definition 3 due to the arbitrary sign of* θ_i *in* (8) *because* $-\min(x_1, 0)$ *is convex.*

Definition 3 is not only applicable to constraints in which the nonsmooth part depends on only one variable. The next example shows reformulations for two multivariate nonsmooth constraints that often appear in practice.

Example 3.

- *Consider the bivariate absolute value constraint* $c(x_1, x_2) = |x_1 - x_2| = 0$. *By introducing an auxiliary variable* x_3 *subject to the constraint*

$$c_{\text{aux}}(x_1, x_2, x_3) = x_1 - x_2 - x_3 = 0$$

  one can rewrite $c(x_1, x_2) = 0$ *equivalently by*

$$c_{\text{aux}}(x_1, x_2, x_3) = x_1 - x_2 - x_3 = 0 \quad \text{and} \quad \hat{c}(x_3) = |x_3| = 0.$$

  $\hat{c}(x_3)$ *fits into the framework of Definition 3.*
- *Consider the bivariate constraint* $c(x_1, x_2) = \max(x_1, x_2) = 0$. *One can easily see that*

$$c(x_1, x_2) = \frac{1}{2}(x_1 + x_2 + |x_1 - x_2|)$$

  holds. Using the reformulation of the preceding example one sees that this also fits into the framework of Definition 3.

In the following, smooth and nonsmooth constraints of a problem are explicitly distinguished. For this, the set of nonsmooth constraints is denoted by the index set $\mathcal{N} \subset \{1, \ldots, m\}$. The second property that the constraints under consideration have to satisfy is that there exist so-called localization functions for all nonsmooth constraints $c_i, i \in \mathcal{N}$.

Definition 4 (Localization Functions)**.** *Let* $c_i, i \in \mathcal{N}$, *be a piecewise smooth and locally Lipschitz-continuous function. A function* $\ell_i(x) : \mathbb{R}^n \to \mathbb{R}$ *is called a* localization function *for* c_i *if and only if*

$$c_i \in \mathcal{C}^2(\mathcal{S}(\ell_i)), \tag{9}$$

where

$$\mathcal{S}(\ell_i) := \{x \in \mathbb{R}^n : \ell_i(x) \neq 0\}$$

denotes the set of points at which ℓ_i *does not vanish. The set*

$$\mathcal{K}_{c_i} = \{x \in \mathbb{R}^n : \ell_i(x) = 0\}$$

is the set of points at which c_i fails to be differentiable. The set

$$\mathcal{K} = \{x \in \mathbb{R}^n \colon \textit{it exists } i \in \mathcal{N} \textit{ with } \ell_i(x) = 0\} = \bigcup_{i \in \mathcal{N}} \mathcal{K}_{c_i}$$

is the set of points at which at least one constraint c_i fails to be differentiable.

Note that it is not possible to use the support $\operatorname{supp}(\ell_i)$ in (9), because $\operatorname{supp}(\ell_i)$ is defined to be the closure of $\mathcal{S}$. The purpose of localization functions is that one can easily check whether a constraint c_i is smooth or not at a given point $x \in \mathbb{R}^n$. The next example shows that it is easy to find localization functions for many nonsmooth functions that appear in practice.

Example 4.

- *The function $\ell(x_1) = x_1$ is a localization function for the constraints*

$$c_1(x_1) = |x_1| = 0, \quad c_2(x_1) = \min(x_1, 0) = 0, \quad c_3(x_1) = \max(x_1, 0) = 0.$$

- *The function $\ell(x_1, x_2) = x_1 - x_2$ is a localization function for the constraints*

$$\begin{aligned} c_4(x_1, x_2) &= |x_1 - x_2| = 0, \\ c_5(x_1, x_2) &= \min(x_1, x_2) = 0, \\ c_6(x_1, x_2) &= \max(x_1, x_2) = 0. \end{aligned}$$

The rest of this section treats problem (1) where $f \in \mathcal{C}^2$ and the constraints c satisfy the following conditions:

(N1): All constraints c are piecewise smooth, i.e. piecewise $\mathcal{C}^2$, and locally Lipschitz-continuous.

(N2): All nonsmooth constraints $c_i, i \in \mathcal{N}$, satisfy the separable nonsmoothness property.

(N3): There exist localization functions for all nonsmooth constraints $c_i, i \in \mathcal{N}$.

Definition 5. *A problem of type* (1) *satisfying (N1)–(N3) is called an* optimization problem with locatable and separable nonsmoothness.

For the description of the interior-point algorithm that solves optimization problems with locatable and separable nonsmoothness it is assumed that the problem is already given in the form of (8). Thus, the problem is already given with auxiliary variables x_{i_a} and auxiliary constraints $x_{i_a} \pm \theta_i(x_{i_\nu}) = 0$ for all nonsmooth constraints c_i. In particular, all nonsmooth constraints are already replaced by their liftings. This leads to the following general problem formulation:

$$\min_{(x, x_a)} \quad f(x) \tag{10a}$$

$$\text{s.t.} \quad c_i(x) = 0 \quad \text{for all } i \in \{1, \dots, m\} \setminus \mathcal{N}, \tag{10b}$$

$$\tilde{c}_i(x, x_{i_a}) = 0 \quad \text{for all } i \in \{1, \dots, m\} \cap \mathcal{N}, \tag{10c}$$

$$\vartheta_i(x_{i_a}, x_{i_\nu}) = x_{i_a} \pm \theta_i(x_{i_\nu}) = 0 \quad \text{for all } i \in \mathcal{N}, \tag{10d}$$

$$x \in \mathbb{R}^n_{\geq 0}, \quad x_a = (x_{i_a})_{i \in \mathcal{N}} \in \mathbb{R}^{|\mathcal{N}|}. \tag{10e}$$

In the following, the variable vector is abbreviated by $\hat{x} := (x, x_a) \in \mathbb{R}^{\bar{n}}$ with $\bar{n} = n + |\mathcal{N}|$. If the distinction between x and x_a is not required, the variable vector is also abbreviated by x again. Notice that the assumption of a smooth objective function is without loss of generality: A nonsmooth objective function $f(x)$ can easily be substituted by an auxiliary variable x_f that is minimized and that is subject to the additional nonsmooth equality constraint $c_f(x, x_f) = x_f - f(x) = 0$. Obviously, the introduced auxiliary constraint $c_f(x, x_f)$ has to satisfy the conditions (N1)–(N3).

4. An Interior-Point Method for Optimization Problems with Locatable and Separable Nonsmoothness

The main idea of the following is to modify the basic interior-point method discussed in Section 2 as little as possible and as much as necessary such that it can handle optimization problems with locatable and separable nonsmoothness. Thus, only those algorithmic building blocks of Algorithm 1 are replaced that are faced with nonsmooth aspects of the problem.

The main idea of the modification of the method is that the algorithm tries to classify the region in which an iterate lies. With this classification the algorithm decides if it tries to handle nonsmooth aspects of the problem explicitly or if it tries to avoid to handle them. Based on this general strategy, the state of the algorithm is distinguished into three modes:

No convergence: If an iterate $y^{(k)}$ has a stationarity measure $e^{(k)}$ (see (6)) with

$$e^{(k)} > \kappa_m \varepsilon, \quad \kappa_m > 1,$$

the algorithm is in the *no-convergence* mode. In this mode the algorithm tries to avoid points x with $\ell_i(x) = 0$ for all nonsmooth constraints $c_i, i \in \mathcal{N}$. This is realized within a modified backtracking line-search algorithm (see below). The rest of the algorithm stays the same.

Convergence in a smooth region: If an iterate $y^{(k)}$ is reached with

$$\varepsilon < e^{(k)} < \kappa_m \varepsilon \tag{11}$$

and

$$|\ell_i(x^{(k)})| > \varepsilon_{\mathcal{N}} \quad \text{for all } i \in \mathcal{N}, \quad \varepsilon_{\mathcal{N}} > 0,$$

the algorithm is in the *smooth-region-convergence* mode. In this case it is assumed that the current iterate is in a region of local convergence and that there is no point x near $x^{(k)}$ with $\ell_i(x) = 0$ for all nonsmooth constraints $c_i, i \in \mathcal{N}$. In the smooth-region-convergence mode only the backtracking line-search is modified as it is the case in the no-convergence mode.

Convergence in a nonsmooth region: If an iterate $y^{(k)}$ is reached satisfying (11) and

$$\exists i \in \mathcal{N} \quad \text{with} \quad |\ell_i(x^{(k)})| \leq \varepsilon_{\mathcal{N}},$$

the algorithm is in the *nonsmooth-region-convergence* mode. Here, it is assumed that the current iterate is in a region of local convergence and that it is likely that the limit point to which the algorithm may converge is a point $x^* \in \mathcal{K}$. The algorithmic strategy is then modified in a way such that the algorithm avoids to cross over points at which some problem data fails to be differentiable. Thus, if there is a point $x \in \mathcal{K}$ in direction $\Delta x^{(k)}$, i.e.

$$\mathcal{K} \cap \bar{R}(x^{(k)}, \Delta x^{(k)}, \bar{\alpha}_{\text{pri}}^{(k)}) \neq \emptyset,$$

with

$$\bar{R}(x^{(k)}, \Delta x^{(k)}, \bar{\alpha}_{\text{pri}}^{(k)}) := \{x^{(k)} + \alpha \Delta x^{(k)} : \alpha \in (0, \bar{\alpha}_{\text{pri}}^{(k)}]\},$$

the algorithm "visits" x and checks a modified stationarity criterion for nonsmooth problems. If the modified stationarity test passes, the algorithm stops and returns x as a local solution of the nonsmooth problem. Otherwise, the algorithm proceeds with special problem-tailored generalized gradients for those constraints that fail to be differentiable at the iterate. These generalized gradients are used in the Jacobians and the Hessian that are part of the KKT matrix of the next iteration.

4.1. **The Modified Line-Search.** If the algorithm is in the *no-convergence* mode or in the *smooth-region-convergence* mode, the overall goal is to avoid points at which some constraints are not differentiable. For this, the backtracking line-search procedure of Algorithm 1 is modified as follows. The main design of the line-search algorithm stays the same but whenever a new trial point x^+ is computed, the modified procedure first checks whether $x^+ \in \mathcal{K}$ holds or not. If not, the line-search procedure works as usual. In the other case, i.e. $x^+ \in \mathcal{K}$, Algorithm 2 is invoked.

Algorithm 2 : Line-Search Extension for Nonsmooth Problems

Input : Primal iterate $x^{(k)}$, primal search direction $\Delta x^{(k)}$, current primal step length of the "outer" line-search procedure $\alpha_{\text{pri}}^{(k,l)}$ (i.e., the line-search procedure is in backtracking iteration l), backtracking factor $\kappa_{b^-} \in (0,1)$ of the "outer" line-search procedure and "forth-tracking" factor $\kappa_{b^+} > 1$ with $\kappa_{b^+}\kappa_{b^-} < 1$, $\bar{\kappa} \in \mathbb{N}$.

Initialize `trial-found` = `false` and $\kappa = 0$.
while not `trial-found` **do**
    if $\kappa > \bar{\kappa}$ **then**
        Stop the algorithm; no trial point $x^+ \notin \mathcal{K}$ could be found.
    Compute a new trial point $x^+ = x^{(k)} + \alpha_{\text{pri}}^{(k,l)} \Delta x^{(k)}$.
    if $\exists i \in \mathcal{N}$ with $\ell_i(x^+) = 0$ **then**
        if `allow-incr` = `true` **then**
            Set $\alpha_{\text{pri}}^{(k,l)} \leftarrow \kappa_{b^+} \alpha_{\text{pri}}^{(k,l)}$.
            Set `allow-incr` = `false`.
        else
            Set $\alpha_{\text{pri}}^{(k,l)} \leftarrow \kappa_{b^-} \alpha_{\text{pri}}^{(k,l)}$.
            Set `allow-incr` = `true`.
        Set $\kappa \leftarrow \kappa + 1$.
    else
        Set `trial-found` = `true`.
return trial point x^+ and primal step length $\alpha_{\text{pri}}^{(k,l)}$.

Remark 1.

(1) *A "forth-tracking" is allowed in the modified line-search in order to avoid small step lengths that result from visited points in the line-search procedure at which some constraints fail to be differentiable. By this, the algorithm tries to avoid unnecessary small step lengths. The boolean flag* `allow-incr` *states whether a "forth-tracking" is allowed or not.*

(2) *The boolean flag* `allow-incr` *in Algorithm 2 is set to* `false` *in the beginning of the "outer" line-search procedure and set to* `true` *after every backtracking step in the "outer" line-search procedure.*

This yields the following lemma:

Lemma 3. *Consider an "outer" line-search procedure with the extension given in Algorithm 2.*

(i) *The primal step length is never increased two times consecutively.*

(ii) *The step length computed by the extended line-search is not greater than $\bar{\alpha}_{\text{pri}}^{(k)}$. In other words, the line-search procedure extended by the sub-procedure given in Algorithm 2 is still a backtracking algorithm.*

Proof. (i) The `if`-part in line 8 of Algorithm 2 can only be reached if $\kappa = 0$ and $l > 1$ (i.e. after a backtracking in the original line-search algorithm) or if $\kappa > 0$ and the `else`-block in line 11 of Algorithm 2 is reached in the last sub-iteration $\kappa - 1$. In the former case the `allow-incr` flag is set to `true` after a backtracking step of the "outer" line-search (cf. Remark 1). For the latter case, the step length is decreased in the last sub-iteration $\kappa - 1$. In both cases, the assertion holds.

(ii) First, let $l = 0$. Because the `allow-incr` flag is set to `false` in the beginning of the "outer" procedure (cf. Remark 1), the first modification of the primal step length can only be a decrease in line 11 of Algorithm 2. The rest follows directly from part (i) and $\kappa_{b^+}\kappa_{b^-} < 1$.

□

Unfortunately, it is not possible to prove that Algorithm 2 terminates after a finite number of iterations without the safeguard in line 4. There might be pathological examples of piecewise smooth constraints c with $\mathcal{T} \subset \mathcal{K}$, where $\mathcal{T}$ is an infinite set of trial points generated by Algorithm 2. In this case, the algorithm would never stop without the safeguard in line 4. On the one hand, it is not likely in practice that the algorithm is confronted with such constraints. On the other hand, it is possible that a primal search direction Δx and a maximum primal step length $\bar{\alpha}_{\text{pri}}$ with small norm $\|\bar{\alpha}_{\text{pri}}\Delta x\|$ lead to a sequence of trial points for which the nonsmoothness test in line 6 always passes *numerically*. The latter might be the case because the criterion

$$\exists i \in \mathcal{N} \quad \text{with} \quad \ell_i(x^+) = 0$$

is usually implemented as

$$\exists i \in \mathcal{N} \quad \text{with} \quad |\ell_i(x^+)| < \varepsilon_\ell$$

for a given tolerance $\varepsilon_\ell > 0$. In these cases, the safeguard in line 4 gets active and the sub-procedure returns a point at which some constraints fail to be differentiable. If the complete line-search method finds an acceptable point anyway, the main algorithm proceeds with a problem-tailored nonsmooth stationarity test (see Section 4.3).

4.2. **The Step Length Truncation Rule.** If the algorithm is in *nonsmooth-region-convergence* mode the main algorithmic strategy changes. Now, the goal is not to avoid to handle the nonsmoothness of the problem but to handle it explicitly. For this, a test that checks whether there is a nonsmooth point on the set $\bar{R} := \bar{R}(x^{(k)}, \Delta x^{(k)}, \bar{\alpha}_{\text{pri}}^{(k)})$ is included. Thus, it is distinguished whether

$$\bar{R} \cap \mathcal{K} = \emptyset \tag{12}$$

holds or not. If (12) holds, then all constraints are smooth on $\bar{R}$ and the maximum primal step length $\bar{\alpha}_{\text{pri}}^{(k)}$ can be used as the initial primal step length in the line-search procedure. Otherwise, the step length is truncated again, obtaining $\hat{\alpha}_{\text{pri}}^{(k)} < \bar{\alpha}_{\text{pri}}^{(k)}$ with

$$R(x^{(k)}, \Delta x^{(k)}, \hat{\alpha}_{\text{pri}}^{(k)}) \cap \mathcal{K} = \emptyset.$$

Here, R is defined by

$$R(x^{(k)}, \Delta x^{(k)}, \hat{\alpha}_{\text{pri}}^{(k)}) := \{x^{(k)} + \alpha \Delta x^{(k)} \colon \alpha \in (0, \hat{\alpha}_{\text{pri}}^{(k)})\}.$$

Notice that $\bar{R}$ and R only differ in the property if the interval α belongs to is closed at its right end or not. In order to determine $\hat{\alpha}_{\text{pri}}^{(k)}$, consider the one-dimensional optimization problems

$$\min \quad \alpha_i \tag{13a}$$

$$\text{s.t.} \quad \ell_i(x^{(k)} + \alpha_i \Delta x^{(k)}) = 0, \tag{13b}$$

$$\alpha_i \in [0, \bar{\alpha}_{\text{pri}}^{(k)}] \tag{13c}$$

for all $i \in \mathcal{N}$.

Lemma 4. *If* (13) *is infeasible, then $c_i \in \mathcal{C}^2(\bar{R})$. If* (13) *is feasible and has the global solution $\hat{\alpha}_i$, then $c_i \in \mathcal{C}^2(\hat{R}_i)$ with $\hat{R}_i := \{x^{(k)} + \alpha \Delta x^{(k)} \colon \alpha \in (0, \hat{\alpha}_i)\}$.*

Proof. (13) is infeasible if there is no α_i with $\ell_i(x^{(k)} + \alpha_i \Delta x^{(k)}) = 0$ and $\alpha_i \in [0, \bar{\alpha}_{\text{pri}}^{(k)}]$. Thus, the localization function has no root $x \in \bar{R}$, giving $c_i \in \mathcal{C}^2(\bar{R})$.

If (13) is feasible with optimal value $\hat{\alpha}_i$, there is no $\tilde{\alpha}_i \in [0, \hat{\alpha}_i)$ with $\ell_i(x^{(k)} + \tilde{\alpha}_i \Delta x^{(k)}) = 0$. Thus, $c_i \in \mathcal{C}^2(\hat{R}_i)$. □

Algorithm 3 : Step Length Truncation Rule for Nonsmooth Problems

---

Input : Primal iterate $x^{(k)}$, primal search direction $\Delta x^{(k)}$ and maximum primal step length $\bar{\alpha}_{\text{pri}}^{(k)}$.

1 **for** all $i \in \mathcal{N}$ **do**

2   Compute $\hat{\alpha}_i$ by solving (13) to global optimality.

3   If (13) is infeasible, set $\hat{\alpha}_i = \bar{\alpha}_{\text{pri}}^{(k)}$.

4 **return** $\hat{\alpha}_{\text{pri}}^{(k)} := \min_{i \in \mathcal{N}} \{\hat{\alpha}_i\}$.

Algorithm 3 states the complete step length truncation rule. By construction of the algorithm and Lemma 4, the following assertion holds.

Lemma 5. *Assume that the maximum step length $\hat{\alpha}_{\text{pri}}^{(k)}$ is computed by Algorithm 3 and that all optimization problems* (13) *are solved to global optimality in line 2 of Algorithm 3. Then all constraints c are smooth on*

$$\hat{R} := \{x^{(k)} + \alpha \Delta x^{(k)} \colon \alpha \in (0, \hat{\alpha}_{\text{pri}}^{(k)})\}.$$

In the following, an iteration in which the maximum primal step length is shortened by Algorithm 3, i.e. an iteration for which $\hat{\alpha}_{\text{pri}}^{(k)} < \bar{\alpha}_{\text{pri}}^{(k)}$ holds, is called a *$\mathcal{K}$-iteration.*

Lemma 6. *Assume $\Delta x_{\text{pri},i}^{(k)} \neq 0$ for all i and $\bar{\alpha}_{\text{pri}}^{(k)} > 0$. Then Algorithm 3 always returns positive step lengths $\hat{\alpha}_{\text{pri}}^{(k)} > 0$.*

Proof. The following two cases are distinguished:

(i) Consider $x^{(k)} \in \mathcal{K}$. Thus, at least one c_i is not differentiable at $x^{(k)}$. Since all c_i are piecewise smooth and $\bar{\alpha}_{\text{pri}}^{(k)} \Delta x^{(k)} \neq 0$, there exists an $\varepsilon > 0$ such that all c_i are smooth on $\{x^{(k)} + \alpha \Delta x^{(k)} \colon \alpha \in (0, \varepsilon)\}$. Hence, $\hat{\alpha}_{\text{pri}}^{(k)} > 0$.

(ii) Assume $x^{(k)} \notin \mathcal{K}$. Since all c_i are piecewise smooth, there exists an $\varepsilon > 0$ such that all c_i are smooth on the open ε-ball $B_\varepsilon(x^{(k)}) := \{x \in \mathbb{R}^n \colon \|x - x^{(k)}\|_2 < \varepsilon\}$. Thus, $\hat{\alpha}_{\text{pri}}^{(k)} \geq \min(\varepsilon, \bar{\alpha}_{\text{pri}}^{(k)}) > 0$.

□

4.2.1. *Some Remarks on Problem* (13). To ensure that Lemma 5 is valid one has to solve the one-dimensional optimization problems (13) to global optimality. Unfortunately, this is practically impossible if there are no additional requirements on the localization functions $\ell_i, i \in \mathcal{N}$. For a lot of problems that are relevant in practice (cf. Example 4), the localization functions are linear. In this case, the problems (13) for $i \in \mathcal{N}$ can be rewritten as the single linear optimization problem

$$\max \quad \alpha \tag{14a}$$

$$\text{s.t.} \quad s_i \ell_i(x^{(k)} + \alpha \Delta x^{(k)}) \geq 0 \quad \text{for all } i \in \mathcal{N}, \tag{14b}$$

$$\alpha \in [0, \bar{\alpha}_{\text{pri}}^{(k)}], \tag{14c}$$

with

$$s_i := \ell_i(x^{(k)}).$$

Problem (14) contains one variable with simple bounds and $|\mathcal{N}|$ linear inequality constraints. It can be solved to global optimality by any LP solver. In contrast to (13), (14) has the additional advantage that it is always feasible. If none of the constraints (14b) is active in the solution (except for those for which $s_i = 0$), there is no point of non-differentiability for any c_i on the set $R(x^{(k)}, \Delta x^{(k)}, \bar{\alpha}_{\text{pri}}^{(k)})$ and the optimal solution of (14) is $\bar{\alpha}_{\text{pri}}^{(k)}$.

In the more general case in which some of the localization functions are nonlinear, one typically wants to apply Newton's method for computing the zeros of the localization functions. For this case, all localization functions should only possess nondegenerate roots in order achieve fast convergence of Newton's method.

4.3. **The Nonsmooth Stationarity Test.** Interior-point algorithms try to compute a KKT point of the problem at hand. Hence, the proposed interior-point algorithm for nonsmooth constrained problems tries to compute a KKT point with respect to the KKT conditions (7) stated in Theorem 3. A termination criterion has to check whether a KKT point is approximately reached or not. To state such a termination criterion for piecewise smooth and locally Lipschitz-continuous problems one especially has to consider condition (7a) for nonsmooth problems.

In the following theorem, e_i denotes the i-th unit vector.

Theorem 4. *Consider problem* (10) *with locatable and separable nonsmoothness and let y be a primal-dual iterate. Furthermore, let $\mathcal{K}(y) \subset \{1, \ldots, m\}$ be the set of indices of constraints that fail to be differentiable at y and set*

$$\delta := \nabla f(x) - z - \sum_{i \in \{1,\ldots,m\} \setminus \mathcal{K}(y)} \lambda_i \nabla c_i(x) - \sum_{i \in \mathcal{K}(y)} \lambda_i e_{i_a}.$$

Define

$$\tilde{I}_i := \lambda_i I_i, \quad i \in \mathcal{K}(y), \tag{15}$$

where the I_i are the subdifferentials at y of the corresponding univariate, piecewise smooth and locally Lipschitz-continuous functions θ_i (cf. Definition 3). In (15)*, the multiplication of a scalar α with an interval $I := [I^-, I^+]$ is defined as*

$$\alpha I := \begin{cases} [\alpha I^-, \alpha I^+], & \alpha \geq 0, \\ [\alpha I^+, \alpha I^-], & \alpha < 0. \end{cases}$$

Moreover, define

$$\mathcal{X} := \{k \in \{1, \ldots, n\} : \exists i \in \mathcal{K}(y) \text{ with } k = i_\nu\}.$$

Finally, set $\hat{I}_j := [\hat{I}_j^-, \hat{I}_j^+]$ for $j \in \mathcal{X}$ with

$$\hat{I}_j^- = \sum_{i \in \mathcal{K}(y) : i_\nu = j} \tilde{I}_i^-, \qquad \hat{I}_j^+ = \sum_{i \in \mathcal{K}(y) : i_\nu = j} \tilde{I}_i^+.$$

Then condition (7a) *holds at* y *if there exist scalars* $\gamma_j \in \hat{I}_j$ *such that*

$$\delta_j \mp \gamma_j = 0 \quad \textit{for all } j \in \mathcal{X}$$

and

$$\delta_j = 0 \quad \textit{for all } j \notin \mathcal{X}.$$

Proof. First, it is known from Lemma 2 that subdifferentials of univariate functions (like θ_i in Definition 3) are intervals. Then condition (7a) with $\xi = 1$ is given by

$$\begin{aligned}
0 &\in \partial_x \mathcal{L}(x, \lambda, z) && \text{(16a)}\\
&= \partial f(x) - \sum_{i \in \{1,\dots,m\}} \lambda_i \partial c_i(x) - z && \text{(16b)}\\
&= \partial f(x) - \sum_{i \in \mathcal{K}(y)} \lambda_i \partial c_i(x) - \sum_{i \in \{1,\dots,m\} \setminus \mathcal{K}(y)} \lambda_i \partial c_i(x) - z && \text{(16c)}\\
&= \nabla f(x) - \sum_{i \in \{1,\dots,m\} \setminus \mathcal{K}(y)} \lambda_i \nabla c_i(x) - \sum_{i \in \mathcal{K}(y)} \lambda_i \partial c_i(x) - z && \text{(16d)}\\
&=: \tilde{\delta} - \sum_{i \in \mathcal{K}(y)} \lambda_i \partial c_i(x). && \text{(16e)}
\end{aligned}$$

In (16c), the sum is split into the constraints that are differentiable at y and constraints that fail to be differentiable at y. (16d) follows because the generalized gradient of a differentiable function is the gradient. Using the fact that all constraints satisfy the separable nonsmoothness property one can rewrite (16e) as

$$\begin{aligned}
&\tilde{\delta} - \sum_{i \in \mathcal{K}(y)} \lambda_i \partial \left[x_{i_a} \pm \theta_i(x_{i_\nu}) \right]\\
&= \tilde{\delta} - \sum_{i \in \mathcal{K}(y)} \lambda_i e_{i_a} \mp \sum_{i \in \mathcal{K}(y)} \lambda_i \partial \theta_i(x_{i_\nu})\\
&= \delta \mp \sum_{i \in \mathcal{K}(y)} \lambda_i \partial \theta_i(x_{i_\nu})\\
&= \delta \mp \sum_{i \in \mathcal{K}(y)} \lambda_i I_i e_{i_\nu}\\
&= \delta \mp \sum_{i \in \mathcal{K}(y)} \tilde{I}_i e_{i_\nu}\\
&= \delta \mp \sum_{j \in \mathcal{X}} \hat{I}_j e_j.
\end{aligned}$$

The theorem follows directly from the last equation. □

In general, it is not easy to evaluate condition (7a) in practice. However, Theorem 4 yields a useful termination criterion for problems with locatable and separable nonsmoothness. The theorem asserts that only subdifferentials, i.e. the intervals I_i, of the nonsmooth univariate functions θ_i are demanded from the user. The rest can be computed easily using the equations stated in the theorem.

4.4. **Problem-Tailored Generalized Gradients.** If the algorithm is at an iterate $x^{(k)} \in \mathcal{K}$ and the nonsmooth stationarity test established in the last section does not pass, the algorithm proceeds. Now, the main problem is that not all gradients of the problem data exist at $x^{(k)}$. Thus, one cannot build the KKT matrix

$$\begin{bmatrix} H(x^{(k)}) & -\nabla c(x^{(k)})^T & -I \\ \nabla c(x^{(k)}) & 0 & 0 \\ Z^{(k)} & 0 & X^{(k)} \end{bmatrix} \tag{17}$$

of the next iteration since it would contain gradients that are not defined. To handle this situation, problem-tailored generalized gradients for those constraints that fail to be differentiable at $x^{(k)}$ are used. The concrete choice of generalized gradients is motivated in the following.

In most of the cases in which no gradient $\nabla c_i(x^{(k)})$ exists, the last iteration was a $\mathcal{K}$-iteration in which the step length is truncated in order to avoid a step over a point at which some constraints fail to be differentiable. Thus, without the step length truncation rule, the algorithm would have taken a longer step into the last search direction $\Delta x^{(k-1)}$. In the current iteration, the algorithm "remembers" this fact and uses the first-order and second-order information belonging to the (smooth) region the last search direction $\Delta x^{(k-1)}$ points to. The idea of the problem-tailored generalized gradients is motivated in the following example and is defined formally afterwards.

Example 5 (The Absolute Value Function)**.** *Consider the absolute value function* $c(x) = |x|$. *c is smooth on* $\mathbb{R} \setminus \{0\}$ *and locally Lipschitz-continuous. A localization function is* $\ell(x) = x$. *Moreover, consider an iterate* $x^{(k-1)} < 0$, *a search direction* $\Delta x^{(k-1)} > |x^{(k-1)}|$ *and a maximum primal step length* $\bar{\alpha}_{\text{pri}}^{(k-1)} = 1$. *In this situation, the primal step length is truncated to* $\hat{\alpha}_{\text{pri}}^{(k-1)} = -x^{(k-1)}/\Delta x^{(k-1)} < 1$ *by Algorithm 3. Under the assumption that* $x^{(k-1)} + \hat{\alpha}_{\text{pri}}^{(k-1)} \Delta x^{(k-1)}$ *is accepted by the globalization strategy, it follows that* $x^{(k)} = 0$ *and* $\ell(x^{(k)}) = 0$. *Hence,* $x^{(k)} \in \mathcal{K}$ *so that* $\nabla c(x^{(k)})$ *does not exist. This situation is interpreted in a way that the algorithm "would like" to take a longer step in the direction* $\Delta x^{(k-1)}$, *i.e. towards the right half plane* $\mathbb{R}_+$, *which was prohibited by the step length truncation rule. Thus, the non-existing gradient* $\nabla c(x^{(k)})$ *is replaced by the derivative of a smooth continuation of c restricted to the positive half space* $\mathbb{R}_+$. *More precisely, the (later defined) generalized gradient* $\tilde{\nabla} c(0, \Delta x^{(k-1)}) = 1$ *is used. Figure 1 illustrates the situation.*

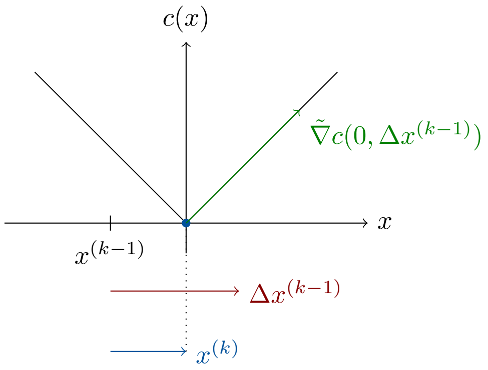

FIGURE 1. A generalized gradient for the absolute value function

Next, a formal definition for what is described in the last example is given.

Definition 6. *Consider a constraint* $\vartheta_i : \mathbb{R}^{\bar{n}} \to \mathbb{R}$ *of problem* (10), *i.e.*

$$\vartheta_i(\hat{x}) = \vartheta_i(x, x_a) = x_{i_a} \pm \theta_i(x_{i_\nu}).$$

Furthermore, let $\hat{d} \in \mathbb{R}^{\bar{n}}$ *and* $\tilde{d}_{i_\nu} \in \mathbb{R}^{\bar{n}}$ *the vector of zeros except for* $\hat{d}_{i_\nu}/|\hat{d}_{i_\nu}|$ *at the* i_ν*-th component. If* $\hat{d}_{i_\nu} = 0$, $\tilde{d}_{i_\nu}$ *is the vector of zeros. The* generalized signed one-sided partial directional derivative w.r.t. x_{i_ν} *of* ϑ_i *at* $\hat{x}$ *in the direction* $\hat{d}$ *is* *defined as*

$$\widetilde{\partial}_{i_\nu}\vartheta_i(\hat{x};\hat{d}) := \operatorname{sign}(\hat{d}_{i_\nu}) \lim_{t\downarrow 0} \frac{\vartheta_i(\hat{x}+t\tilde{d}_{i_\nu}) - \vartheta_i(\hat{x})}{t}. \tag{18}$$

Here, $\operatorname{sign}(0)$ *is defined to be* 0.

Lemma 7. *The generalized signed one-sided partial directional derivative w.r.t.* x_{i_ν} *of the constraint* ϑ_i *of problem* (10) *is well-defined.*

Proof. The difference quotient in (18) is bounded by the local Lipschitz constant $L = L(\hat{x})$ and the existence of the limit follows directly from the piecewise smoothness of ϑ_i. □

Notice that the last definition is similar to the definition of standard one-sided directional derivatives. Both coincide for the one-dimensional case. Nevertheless, Definition 6 is more general in the sense that it allows the definition of the *generalized signed one-sided directional gradient*:

Definition 7. *Let* $\vartheta_i, \hat{x}$ *and* $\hat{d}$ *be as in Definition 6. The* generalized signed one-sided directional gradient *of* ϑ_i *at* $\hat{x}$ *in the direction* $\hat{d}$ *is defined as*

$$\widetilde{\nabla}\vartheta_i(\hat{x};\hat{d}) := \left(\widetilde{\partial}_j\vartheta_i(\hat{x};\hat{d})\right)_{j=1}^{\bar{n}} \in \mathbb{R}^{\bar{n}},$$

where $\widetilde{\partial}_j\vartheta_i(\hat{x};\hat{d}) := \partial_{\hat{x}_j}\vartheta_i(\hat{x})$*, i.e. the standard partial derivative of* ϑ_i*, for all* $j \neq i_\nu$ *and* $\widetilde{\partial}_{i_\nu}$ *is the generalized signed one-sided partial directional derivative w.r.t.* x_{i_ν}.

Example 6 (The Absolute Value Function Revisited)**.** *The last two definitions formalize exactly what is described in Example 5. Consider the constraint*

$$\vartheta_i(x_1,x_2) = x_2 + \theta_i(x_1) = x_2 + |x_1| = 0.$$

Here, $i_a = 2$ *and* $i_\nu = 1$*. Let* $\hat{x} = (0,0)^T$ *and* $\hat{d} = (1,\hat{d}_2)^T$ *for arbitrary* $\hat{d}_2$*. Thus,* $\tilde{d}_{i_\nu} = (1,0)^T$*. Definition 6 implies*

$$\begin{aligned}\widetilde{\partial}_{i_\nu}\vartheta_i(\hat{x};\hat{d}) &= \operatorname{sign}(\hat{d}_{i_\nu}) \lim_{t\downarrow 0} \frac{\vartheta_i(\hat{x}+t\tilde{d}_{i_\nu}) - \vartheta_i(\hat{x})}{t}\\ &= \lim_{t\downarrow 0}\frac{\theta_i(t)}{t}\\ &= 1.\end{aligned}$$

Analogously, one obtains $\widetilde{\partial}_{i_\nu}\vartheta_i(\hat{x};\hat{d}) = -1$ *for* $\hat{d} = (-1,\hat{d}_2)^T$*. Thus,* $\widetilde{\nabla}\vartheta_i(\hat{x};\hat{d}) = (1,1)^T$ *holds for* $\hat{d} = (1,\hat{d}_2)^T$ *and* $\widetilde{\nabla}\vartheta_i(\hat{x};\hat{d}) = (-1,1)^T$ *holds for* $\hat{d} = (-1,\hat{d}_2)^T$.

The following theorem shows that the generalized signed one-sided directional gradient belongs to Clarke's generalized gradient.

Theorem 5. *Consider a constraint* $\vartheta_i : \mathbb{R}^{\bar{n}} \to \mathbb{R}$ *of problem* (10)*. Furthermore, let* $\hat{x} \in \mathbb{R}^{\bar{n}}$ *and* $\hat{d} \in \mathbb{R}^{\bar{n}}$ *with* $\hat{d}_{i_\nu} \neq 0$*. Then*

$$\widetilde{\nabla}\vartheta_i(\hat{x};\hat{d}) \in \partial\vartheta_i(\hat{x})$$

holds.

Proof. Using Lemma 1, it is sufficient to show that there exists a sequence $(\hat{x}_k)$ with $\hat{x}_k \to \hat{x}$, $\hat{x}_k \in \mathbb{R}^{\bar{n}} \setminus \mathcal{K}$ and $\widetilde{\nabla}\vartheta_i(\hat{x};\hat{d}) = \lim_{\hat{x}_k\to\hat{x}} \nabla\vartheta_i(\hat{x}_k)$. Furthermore, it is enough to show that the theorem holds component-wise, i.e. it remains to prove

$$\widetilde{\partial}_j\vartheta_i(\hat{x};\hat{d}) = \lim_{\hat{x}_k\to\hat{x}} \partial_{\hat{x}_j}\vartheta_i(\hat{x}_k). \tag{19}$$

Let $\hat{x}_k := \hat{x} + \alpha_k\hat{d}$ with positive numbers $\alpha_k \to 0$. For all variables $\hat{x}_j$ with $j \neq i_\nu$, $\widetilde{\partial}_j\vartheta_i(\hat{x};\hat{d})$ is the standard gradient and thus (19) holds because of the piecewise smoothness of ϑ_i. By this reason, it is enough to show the convergence for the i_ν-th component. Starting with the right-hand side, one has

$$\begin{aligned}\lim_{\hat{x}_k \to \hat{x}} \partial_{\hat{x}_{i_\nu}} \vartheta_i(\hat{x}_k) &= \lim_{\hat{x}_k \to \hat{x}} \lim_{t \to 0} \frac{\vartheta_i(\hat{x}_k + te_{i_\nu}) - \vartheta_i(\hat{x}_k)}{t} \\ &= \lim_{\hat{x}_k \to \hat{x}} \operatorname{sign}(\hat{d}_{i_\nu}) \lim_{s \downarrow 0} \frac{\vartheta_i(\hat{x}_k + s\tilde{d}_{i_\nu}) - \vartheta_i(\hat{x}_k)}{s} \\ &= \lim_{\hat{x}_k \to \hat{x}} \widetilde{\partial}_{i_\nu} \vartheta_i(\hat{x}_k; \hat{d}) \\ &= \widetilde{\partial}_{i_\nu} \vartheta_i(\hat{x}; \hat{d}).\end{aligned}$$

The first equality holds due to the definition of the partial derivative that exists for $\hat{x}_k$ since $\hat{x}_k \in \mathbb{R}^{\bar{n}} \setminus \mathcal{K}$ for sufficiently small α_k. Since ϑ_i is differentiable at all $\hat{x}_k$ one can switch to the directional derivative in the direction $\tilde{d}_{i_\nu}$ in the second equation. A possible alternation in the sign is addressed by the factor $\operatorname{sign}(\hat{d}_{i_\nu})$ that is independent of the limit. The penultimate equation is exactly the definition of $\widetilde{\partial}_{i_\nu} \vartheta_i(\hat{x}; \hat{d})$ and the last equation holds because ϑ_i is piecewise smooth, i.e. it is smooth on the interior of the set $\{\hat{x} + \alpha\hat{d} \colon \alpha > 0\}$ for sufficiently small $\|\alpha\hat{d}\|$. □

The assumption "$\hat{d}_{i_\nu} \neq 0$" of the last theorem is of special importance in the algorithm. The algorithm checks, if a starting point given by the user is a point at which some constraints fail to be differentiable. If there are constraints that are non-differentiable at this point, the starting point is perturbed such that $\ell_i(x^{(0)}) \neq 0$ for all $i \in \mathcal{N}$. Thereby, it is guaranteed that $\Delta x_{i_\nu}^{(k-1)} \neq 0$ holds if an iterate $x^{(k)}$ is reached with $\ell_i(x^{(k)}) = 0$. Since $\hat{d}$ is always chosen to be the last search direction in the algorithm, the assumption is not restricting in practice.

In summary, the proposed method chooses the ordinary gradients if they exist and the generalized signed one-sided directional gradients at those points where some constraints are not differentiable. In analogy, second-order information is constructed by applying the same ideas to the (generalized) gradients. This leads to a well-defined system matrix (17) for the Newton step.

5. Numerical Results

In this section numerical results are reported for the extended and modified interior-point method described so far. The numerical experiments are split up into two parts. The first part presents results on a small test library of low-dimensional optimization problems with locatable and separable nonsmoothness. The models of this library are given in Appendix A. The goal of this part is to ensure the correctness of the implementation for models for which the solution process can be comprehended manually. The second part shows the applicability of the new method on real-world instances of gas network optimization problems. More details are given in Section 5.2.

The implemented interior-point method is part of the software framework Clean (C++ Library of Efficient Algorithms in Numerics). Clean is a generic library that is being developed in the working group Algorithmic Optimization of Marc Steinbach at the Leibniz Universität Hannover and by the author. It is intended to become public domain when it is considered to be sufficiently mature.

5.1. Small Test Problems.

In this section the results of the new method for the models of the low-dimensional test problems of Appendix A are presented. All of these models are two-dimensional and have one variable $x_i, i \in \{1, 2\}$, for which one constraint fails to be differentiable at a single point. For every problem, the algorithm is applied for two different starting points. The first starting point is on the "other side" of the point at which the nonsmooth constraint fails to be differentiable than the optimal solution. The second starting point is on the same side as the optimal solution. The former case allows us to test the modified components of the algorithm (cf. Section 4) whereas the latter case may lead to a solution process that may be completely unaffected of the discussed modifications if all iterates stay in the smooth region of the optimal solution without "crossing" the point of non-differentiability.

Not all test problems in Appendix A exactly fit into the form of nonsmooth optimization problems with locatable and separable nonsmoothness (10). All problems have localization functions for all of their nonsmooth constraints but not every nonsmooth constraint fulfills the separable nonsmoothness property. However, all constraints that do not fulfill this property are more general. In these cases the nonsmooth stationarity test as proposed in Section 4.3 is not applicable and thus not executed by the algorithm. All other modifications are applied since they do not depend on the separability of the nonsmoothness.

Table 1 displays the results. The problem ID in the first column corresponds to the name of the problem in Appendix A. The acronym `sp1` stands for the starting point located on the other side of the point of non-differentiability than the optimal solution and `sp2` stands for the starting point on the same side. If the optimal solution is directly on a point of non-differentiability `sp1` denotes a starting point "left" from this point and `sp2` denotes a starting point "right" from this point.

The proposed method solves all low-dimensional test problems to optimality and computes the correct solution vectors and objective values (cf. Appendix A). All solution times are in the milliseconds. The iteration numbers are in an order of magnitude that is typical for interior-point methods for most of the problems. Larger iteration counts—like for problem `06a-sp2`—stem from the modified line-search and step length computation procedures. Figure 2 plots the x_2-coordinates of the final 25 iterates of the solution process. The x_1-coordinate is already at its optimal solution and this solution is directly located at a point ($x_1 = 1$) at which one constraint fails to be differentiable. Here, the step length truncation rule always shortens the primal step length so that only very short steps can be taken towards the optimal solution in the x_2-coordinate. A possible remedy could be to detect these situations that can be characterized as follows:

- There exists a coordinate for which consecutive search directions are zero.
- The values of this coordinate leads to one (or more) vanishing localization functions.
- The algorithm is in nonsmooth-region-convergence mode and consecutively truncates maximum step lengths.

After detection of such situations, one could project the problem formulation by fixing the corresponding coordinate. This would probably lead to faster convergence in this case.

5.2. **Nonsmooth Problems in Gas Transport.** In this section numerical results for the problem of *validation of nominations* in gas networks are presented. In gas transportation, transmission system operators (TSOs) are faced with the problem of transporting gas from *entry points* of the network to *exit points* through a system of pipelines and other network elements. Beside gravitational aspects gas flows from higher to lower pressure. In pipes, the gas pressure decreases due to friction at the inner pipe walls. This effect is encountered in so-called compressor stations that are able to increase the gas pressure in order to transport the gas over large distances. This compression process requires energy that is taken from the gas of the network itself by using, e.g., gas turbines or is delivered by electric motors. The fuel and

TABLE 1. Results for the low-dimensional test problems of Appendix A

Problem ID	Objective value	Solution vector	# Iterations
01a-sp1	-1.24	$(0.6180, 0.6180)^T$	250
01a-sp2	-1.24	$(0.6180, 0.6180)^T$	6
01b-sp1	-1.00	$(0.0000, 1.0000)^T$	5
01b-sp2	-1.00	$(0.0000, 1.0000)^T$	5
02a-sp1	-0.62	$(-0.6180, 0.3820)^T$	8
02a-sp2	-0.62	$(-0.6180, 0.3820)^T$	6
02b-sp1	-2.00	$(-2.0000, -1.0000)^T$	17
02b-sp2	-2.00	$(-2.0000, -1.0000)^T$	5
03a-sp1	-1.25	$(0.5000, 0.7500)^T$	5
03a-sp2	-1.25	$(0.5000, 0.7500)^T$	5
03b-sp1	-1.24	$(0.6180, 0.6180)^T$	17
03b-sp2	-1.24	$(0.6180, 0.6180)^T$	8
04a-sp1	-1.62	$(1.6180, -1.6180)^T$	16
04a-sp2	-1.62	$(1.6180, -1.6180)^T$	5
04b-sp1	-1.00	$(1.0000, -1.0000)^T$	6
04b-sp2	-1.00	$(1.0000, -1.0000)^T$	5
05a-sp1	-0.73	$(-0.7321, 2.0000)^T$	6
05a-sp2	-0.73	$(-0.7321, 2.0000)^T$	5
05b-sp1	-2.00	$(-2.0000, 0.3979)^T$	5
05b-sp2	-2.00	$(-2.0000, 1.1938)^T$	5
06a-sp1	1.00	$(1.0000, -1.0000)^T$	70
06a-sp2	1.00	$(1.0000, -1.0000)^T$	159
06b-sp1	0.00	$(1.3126, -1.3126)^T$	5
06b-sp2	-2.00	$(0.0000, -2.0000)^T$	6
07a-sp1	1.00	$(1.0000, 1.0000)^T$	10
07a-sp2	1.00	$(1.0000, 1.0000)^T$	270
07b-sp1	-4.00	$(2.0000, 2.0000)^T$	7
07b-sp2	-4.00	$(2.0000, 2.0000)^T$	5
08a-sp1	-4.00	$(2.0000, 2.0000)^T$	5
08a-sp2	-4.00	$(2.0000, 2.0000)^T$	5
08b-sp1	-4.00	$(-2.0000, -2.0000)^T$	5
08b-sp2	-4.00	$(-2.0000, -2.0000)^T$	5

electricity consumption are the main costs in gas transportation. Thus, the goal of the TSOs is to transport the gas in way such that all contracts with entry and exit customers are satisfied and that the operation of the compressor stations is as cheap as possible. The reader interested in more details about gas network optimization is referred to, e.g., [8, 15, 19, 21, 22, 23].

Due to the combination of highly nonlinear and nonconvex phenomena of gas dynamics in pipes and other network elements and the switching of controllable elements, e.g. the (de-)activation of compressors, the problem of validation of nominations leads to nonlinear and nonconvex mixed-integer models that are hard to solve for real-world instances. In addition, most models of the network element types comprise certain nonsmooth aspects that can be formulated as models with locatable and separable nonsmoothness. For instance, so-called resistors lead to linear pressure losses in the direction of flow, yielding the nonsmooth model

$$p_{\text{in}} - p_{\text{out}} = \operatorname{sign}(q)\Delta p,$$

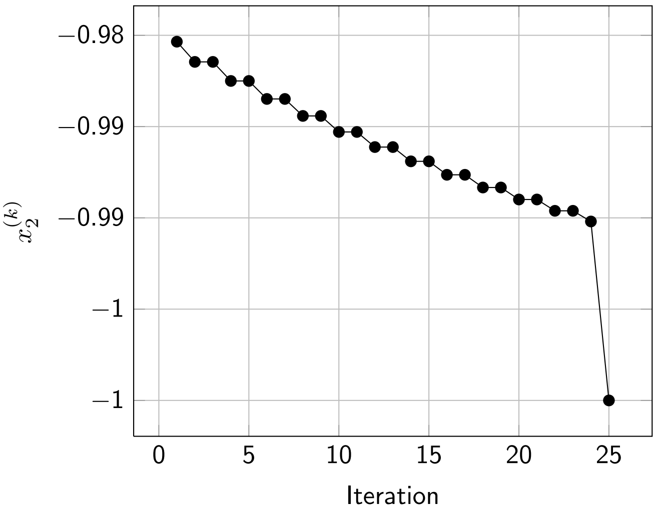

FIGURE 2. The last 25 iterates for problem 06a-sp2

TABLE 2. Size of the reformulated MPCC model (for the northern high-calorific gas network)

Variables	1963
Equality constraints	1872
Inequality constraints	7
Complementarity constraints	64

where $p_{\text{in}}, p_{\text{out}}$ denote the in- and outflow pressure of the resistor, q is the flow through the element and Δp is the constant pressure loss. This model can replaced by

$$q(p_{\text{in}} - p_{\text{out}}) = x_a \Delta p, \quad x_a - \theta(q) = 0, \quad \theta(q) = |q|$$

and thus fits into the framework of Def. 3 and 4. For more details about the details of the model, especially the nonsmooth aspects, see [21, 23].

In order to tackle the mixed-integer aspects of the model (like opening or closing of valves or (de-)activating of compressor stations) problem-specific continuous reformulations of the discrete aspects in gas transportation models are developed in [21, 22]. These reformulations lead to mathematical programs with complementarity conditions (MPCCs) that are regularized by standard methods from the literature. Thus, one ends up with nonlinear and nonconvex problems with locatable and separable nonsmooth aspects.

The instances for which the interior-point method described in Section 4 is applied stem from the industrial project ForNe. The aim of ForNe was to solve practically relevant problems of TSOs by mathematical optimization. The data used are real-world data from the industrial partner Open Grid Europe GmbH (OGE)[1]. Figure 3 shows a schematic plot of the northern high-calorific network of OGE for which the problem of validation of nominations is solved in the following. The network consists of more then 1200 km of pipes, compressor stations and other network elements. The corresponding directed graph comprises 624 edges and 592 vertices. The size of the nonsmooth and complementarity constrained model for this network is given in Table 2.

[1] https://www.open-grid-europe.com

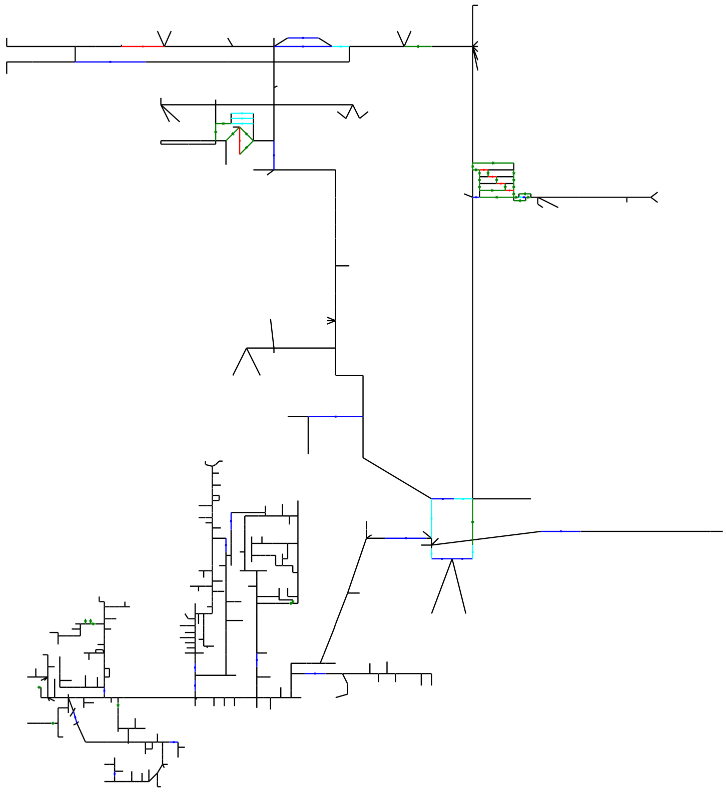

FIGURE 3. Schematic plot of the northern high-calorific gas transport network of Open Grid Europe GmbH

As it is well-known, convergence for highly nonlinear and nonconvex problems often depends on the chosen parameterization of the algorithm. This is the reason why the algorithm of Section 4 is tested for different update strategies for the barrier parameter μ. To be concrete, the algorithm is tested with the rule implemented in LOQO [26, 27], the strategy implemented in Ipopt [30] and Mehrotra's predictor-corrector method [16].

In what follows, results for a set of 1000 randomly chosen nominations are presented. These nominations are generated within the ForNe project based on the current set of entry and exit contracts of OGE and historical data about nominated entry and exit capacities (see [15] for the details). All computations are done with an Intel Core i7 CPU 920 with 2.67 GHz and 12 GB RAM. The operating system is openSUSE 12.1. All executables are generated without any code optimization with the GCC compiler version 4.7.1.

Figure 4 shows the performance profile (as proposed by Dolan and Moré in [5]) for the mentioned 1000 nominations for the northern high-calorific gas network of OGE. The algorithm is parameterized using three different barrier parameter update strategies and three different initial values for the barrier parameter, respectively. It can be seen that all parameterizations of the algorithm lead to comparable results but the larger initial values for the barrier parameter lead to a slightly more robust algorithm on these instances. In addition to the six performance profiles of the different algorithmic choices, Figure 4 also shows a *best-of* profile, i.e. a profile that is achieved if one takes the best of all algorithmic parameterizations for every instance. Obviously, this line is constant and its value ($\approx 90\,\%$) corresponds to the percentage of instances that is solved at least with one parameterization of the

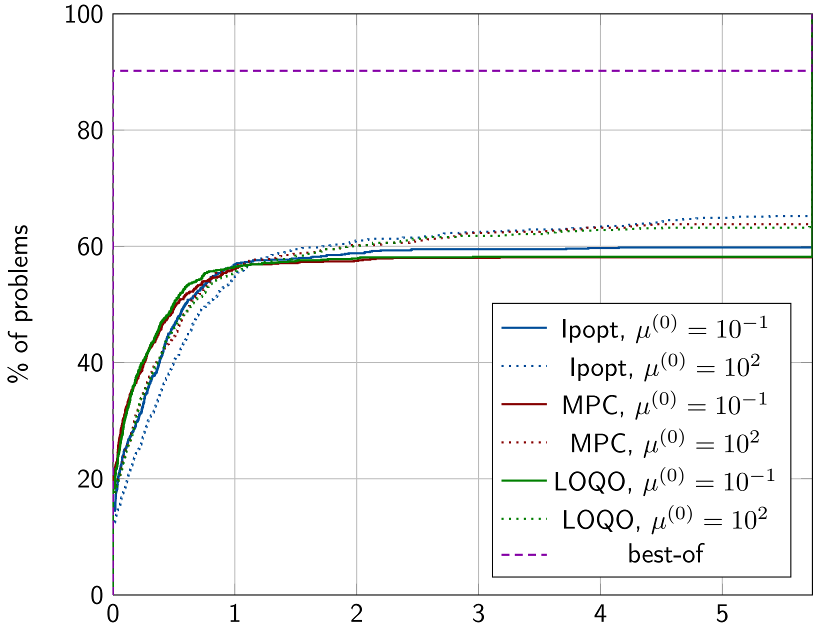

FIGURE 4. Performance profiles for 1000 nominations on the northern high-calorific value network of OGE

TABLE 3. Iteration counts (k) for different barrier parameter update strategies and different initial barrier parameters

Solver option	min. k	max. k	avg. k
Ipopt rule, $\mu^{(0)} = 10^{-1}$	41	1866	112
Ipopt rule, $\mu^{(0)} = 10^{2}$	40	2807	213
MPC rule, $\mu^{(0)} = 10^{-1}$	47	703	95
MPC rule, $\mu^{(0)} = 10^{2}$	43	2086	168
LOQO rule, $\mu^{(0)} = 10^{-1}$	47	1479	95
LOQO rule, $\mu^{(0)} = 10^{2}$	45	5241	167

algorithm. By this, it can be seen that the solution process for these instances significantly depends on the chosen algorithmic options. This is a well-known fact for nonlinear optimization in general and seems to be particularly true for the considered class of nonsmooth MPCCs.

Table 3 shows a statistic of the minimum, maximum and average iteration numbers of all successfully finished runs. Again, there are quite large iteration numbers which arise due to the same reasons as described in Section 5.1. Figure 5 plots the primal and dual step lengths for an exemplary instance. It can be seen that the primal step lengths are often shortened up to a size of 10^{-4} leading to slow convergence. Solution times can be found in Table 4. It should be remarked, that approximately 2/3 of the solution time is used for the evaluation of the problem data.

To sum up, the results show that the approach of tackling the nonsmooth aspects of the models directly within the interior-point method can be applied to real-world instances of hard optimization problems. Approximately 90 % of the instances can be solved which is a quite high number for this challenging class of problems.

TABLE 4. Solution times (t, in s) for different barrier parameter update strategies and different initial barrier parameters

Solver option	min. t	max. t	avg. t
Ipopt rule, $\mu^{(0)} = 10^{-1}$	6	442	17
Ipopt rule, $\mu^{(0)} = 10^{2}$	6	425	31
MPC rule, $\mu^{(0)} = 10^{-1}$	7	130	15
MPC rule, $\mu^{(0)} = 10^{2}$	7	325	27
LOQO rule, $\mu^{(0)} = 10^{-1}$	5	217	14
LOQO rule, $\mu^{(0)} = 10^{2}$	6	300	25

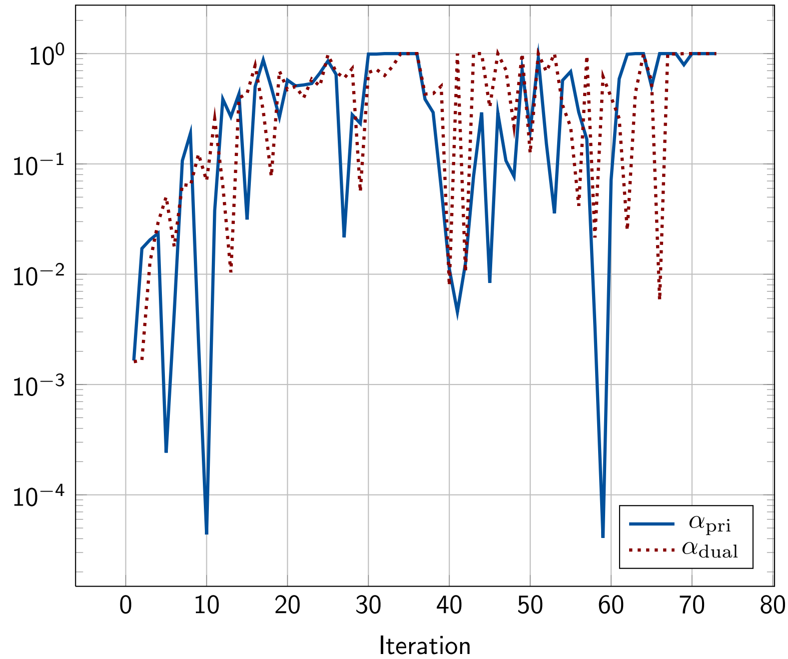

FIGURE 5. Primal and dual step lengths during the solution of an exemplary instance

6. Conclusions and Outlook

In this article a special subclass of nonsmooth optimization problems is introduced and a new interior-point method for this class is proposed. This method extends a standard interior-point method for nonlinear and nonconvex problems by additional line-search and step length computation features and a problem-tailored stationarity test for the introduced problem class. Moreover, a special type of generalized gradients is developed that is used at points of the nonsmooth problems that fail to be differentiable. First theoretical results are given for the algorithmic extensions like the belonging of the generalized gradients to Clarke's generalized gradient. However, a convergence result is outstanding and thus topic of future research. In addition, several algorithmic issues could be improved in order to increase the robustness of the method. Here, future topics include the fixation of variables that lead to non-differentiability of some constraints and projection of the problem to increase the regularity of the model. Despite this open task and other possible improvements, the presented numerical results show the desired behavior of the algorithm and the applicability of the method to real-world problems from industry.

Acknowledgement

This work has been supported by the German Federal Ministry of Economics and Technology owing to a decision of the German Bundestag. The responsibility for the content of this publication lies with the author. The author would also like to thank Open Grid Europe GmbH and the project partners in the ForNe consortium. This research was performed as part of the Energie Campus Nürnberg and supported by funding through the "Aufbruch Bayern (Bavaria on the move)" initiative of the state of Bavaria. Moreover, the author thanks Marc Steinbach, Jan Thiedau and Andreas Wächter for several comments on the algorithm.

References

[1] Richard H. Byrd, Jean Charles Gilbert, and Jorge Nocedal. "A trust region method based on interior point techniques for nonlinear programming." In: *Math. Program., Ser. B* 89.1 (2000), pp. 149–185. ISSN: 0025-5610. DOI: 10.1007/PL00011391.

[2] Francis H. Clarke, Yuri S. Ledyaev, Ronald J. Stern, and Peter R. Wolenski. *Nonsmooth Analysis and Control Theory.* Vol. 178. Graduate Texts in Mathematics. Springer, 1998.

[3] Frank H. Clarke. *Optimization and Nonsmooth Analysis.* Society for Industrial and Applied Mathematics, 1990. DOI: 10.1137/1.9781611971309.

[4] Andrew R. Conn, Nicholas I. M. Gould, and Philippe Toint. *Trust-Region Methods.* MPS-SIAM series on optimization. SIAM, 2000.

[5] Elizabeth D. Dolan and Jorge J. Moré. "Benchmarking Optimization Software with Performance Profiles." In: *Math. Program.* 91 (2 2002), pp. 201–213. ISSN: 0025-5610. DOI: 10.1007/s101070100263.

[6] Roger Fletcher and Sven Leyffer. "Nonlinear Programming without a penalty function." In: *Math. Program.* 91 (2000), pp. 239–269.

[7] Anders Forsgren, Philip E. Gill, and Margaret H. Wright. "Interior Methods for Nonlinear Optimization." In: *SIAM Review* 44.4 (2002), pp. 525–597.

[8] Armin Fügenschuh, Björn Geißler, Ralf Gollmer, Christine Hayn, René Henrion, Benjamin Hiller, Jesco Humpola, Thorsten Koch, Thomas Lehmann, Alexander Martin, Radoslava Mirkov, Antonio Morsi, Jessica Rövekamp, Lars Schewe, Martin Schmidt, Rüdiger Schultz, Robert Schwarz, Jonas Schweiger, Claudia Stangl, Marc C. Steinbach, and Bernhard M. Willert. "Mathematical optimization for challenging network planning problems in unbundled liberalized gas markets." In: *Energy Systems* 5.3 (2014), pp. 449–473. ISSN: 1868-3967. DOI: 10.1007/s12667-013-0099-8.

[9] Gene H. Golub and Charles F. van Loan. *Matrix Computations.* 2nd. Baltimore, MD, USA: Johns Hopkins University Press, 1989.

[10] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex Analysis and Minimization Algorithms I: Fundamentals.* Vol. 305. Grundlehren der mathematischen Wissenschaften. Springer, 1993.

[11] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods.* Vol. 306. Grundlehren der mathematischen Wissenschaften. Springer, 1993.

[12] Narendra Karmarkar. "A New Polynomial-Time Algorithm for Linear Programming." In: *Combinatorica* 4.4 (1984), pp. 373–395. ISSN: 0209-9683. DOI: 10.1007/BF02579150.

[13] Napsu Karmitsa. *NonSmooth Optimization (NSO) Software*. http://napsu.karmitsa.fi/nsosoftware/.

[14] Leonid Genrikhovich Khachiyan. "A polynomial algorithm in linear programming." In: *Soviet Mathematics Doklady* 20 (1979), pp. 191–194.

[15] Thorsten Koch, Benjamin Hiller, Marc E. Pfetsch, and Lars Schewe, eds. *Evaluating Gas Network Capacities*. SIAM-MOS series on Optimization. SIAM, Dec. 2014. approx.x + 375. ISBN: 978-1-611973-68-6. Forthcoming.

[16] Sanjay Mehrotra. "On the Implementation of a Primal-Dual Interior Point Method." In: *SIAM J. Optim.* 2.4 (1992), pp. 575–601. ISSN: 1052-6234. DOI: 10.1137/0802028.

[17] Jorge Nocedal, Andreas Wächter, and Richard A. Waltz. "Adaptive Barrier Update Strategies for Nonlinear Interior Methods." In: *SIAM J. Optim.* 19.4 (2009), pp. 1674–1693. DOI: 10.1137/060649513.

[18] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. 2nd. Berlin: Springer, 2006. DOI: 10.1007/978-0-387-40065-5.

[19] Marc E. Pfetsch, Armin Fügenschuh, Björn Geißler, Nina Geißler, Ralf Gollmer, Benjamin Hiller, Jesco Humpola, Thorsten Koch, Thomas Lehmann, Alexander Martin, Antonio Morsi, Jessica Rövekamp, Lars Schewe, Martin Schmidt, Rüdiger Schultz, Robert Schwarz, Jonas Schweiger, Claudia Stangl, Marc C. Steinbach, Stefan Vigerske, and Bernhard M. Willert. "Validation of nominations in gas network optimization: models, methods, and solutions." In: *Optimization Methods and Software* (2014). DOI: 10.1080/10556788.2014.888426. Pre-published.

[20] Andrzej Ruszczyński. *Nonlinear Optimization*. Princeton University Press, 2006.

[21] Martin Schmidt. "A Generic Interior-Point Framework for Nonsmooth and Complementarity Constrained Nonlinear Optimization." PhD thesis. Gottfried Wilhelm Leibniz Universität Hannover, 2013.

[22] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. "A Primal Heuristic for Nonsmooth Mixed Integer Nonlinear Optimization." In: *Facets of Combinatorial Optimization*. Ed. by Michael Jünger and Gerhard Reinelt. Berlin, Heidelberg: Springer, 2013, pp. 295–320. ISBN: 978-3-642-38188-1. DOI: 10.1007/978-3-642-38189-8_13.

[23] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. "High detail stationary optimization models for gas networks." English. In: *Optimization and Engineering* (2014), pp. 1–34. ISSN: 1389-4420. DOI: 10.1007/s11081-014-9246-x.

[24] André L. Tits, Andreas Wächter, Sasan Bakhtiari, Thoma J. Urban, and Craig T. Lawrence. "A Primal-Dual Interior-Point Method for Nonlinear Programmng with Strong Global and Local Convergence Properties." In: *SIAM J. Optim.* 14.1 (2003), pp. 173–199.

[25] Michael Ulbrich, Stefan Ulbrich, and Luís N. Vicente. "A globally convergent primal-dual interior-point filter method for nonlinear programming." English. In: *Mathematical Programming* 100.2 (2004), pp. 379–410. ISSN: 0025-5610. DOI: 10.1007/s10107-003-0477-4. URL: http://dx.doi.org/10.1007/s10107-003-0477-4.

[26] Robert J. Vanderbei. *LOQO User's Manual – Version 4.05*. Princeton University, School of Engineering, Applied Science, Department of Operations Research, and Financial Engineering. Princeton, New Jersey, Sept. 2006.

[27] Robert J. Vanderbei and David F. Shanno. "An Interior-Point Algorithm For Nonconvex Nonlinear Programming." In: *Comput. Optim. Appl.* 13 (1997), pp. 231–252.

[28] Andreas Wächter and Lorenz T. Biegler. "Failure of global convergence for a class of interior point methods for nonlinear programming." English. In: *Mathematical Programming* 88.3 (2000), pp. 565–574. ISSN: 0025-5610. DOI: 10.1007/PL00011386.

[29] Andreas Wächter and Lorenz T. Biegler. "Line Search Filter Methods for Nonlinear Programming: Motivation and Global Convergence." In: *SIAM J. on Optimization* 16.1 (May 2005), pp. 1–31. ISSN: 1052-6234. DOI: 10.1137/S1052623403426556. URL: http://dx.doi.org/10.1137/S1052623403426556.

[30] Andreas Wächter and Lorenz T. Biegler. "On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming." In: *Math. Program.* 106.1 (2006), pp. 25–57. ISSN: 0025-5610. DOI: 10.1007/s10107-004-0559-y.

[31] Richard A. Waltz, José Luis Morales, Jorge Nocedal, and Dominique Orban. "An interior algorithm for nonlinear optimization that combines line search and trust region steps." In: *Math. Program.* 107.3 (July 2006), pp. 391–408. ISSN: 0025-5610. DOI: 10.1007/s10107-004-0560-5.

Appendix A. Low-Dimensional Test Problems

A.1. **Problem 1a.**

$$\begin{aligned} \min \quad & f(x) = -x_1 - x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - |x_1| \leq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0, \\ & x_2 \geq -1 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (0.61803, 0.61803)$
Objective value: $f(x^*) = -1.23606$

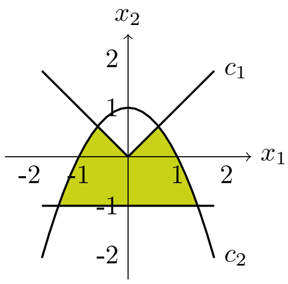

Figure 6. Illustration of test problem 1a

A.2. **Problem 1b.**

$$\begin{aligned} \min \quad & f(x) = -x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - |x_1| \geq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (0, 1)$
Objective value: $f(x^*) = -1$

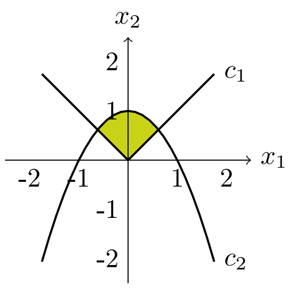

Figure 7. Illustration of test problem 1b

A.3. **Problem 2a.**

$$\begin{aligned} \min \quad & f(x) = x_1 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - x_1^2 \geq 0, \\ & c_2(x_1, x_2) := x_2 + |x_1| - 1 \leq 0 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (-0.61804, 0.38197)$
Objective value: $f(x^*) = -0.61804$

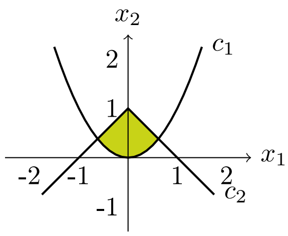

FIGURE 8. Illustration of test problem 2a

A.4. **Problem 2b.**

$$\begin{aligned} \min \quad & f(x) = x_1 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - x_1^2 \leq 0, \\ & c_2(x_1, x_2) := x_2 + |x_1| - 1 \leq 0, \\ & x_2 \geq -1 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (-2, -1)$
Objective value: $f(x^*) = -2$

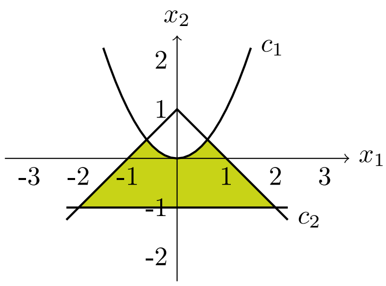

FIGURE 9. Illustration of test problem 2b

A.5. **Problem 3a.**

$$\begin{aligned} \min \quad & f(x) = -x_1 - x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - \max\{0, x_1\} \geq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (0.5, 0.75)$
Objective value: $f(x^*) = -1.25$

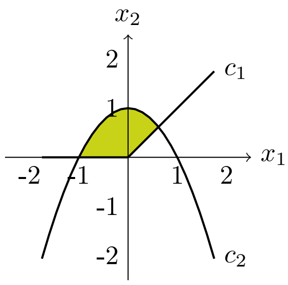

FIGURE 10. Illustration of test problem 3a

A.6. **Problem 3b.**

$$\begin{aligned} \min \quad & f(x) = -x_1 - x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - \max\{0, x_1\} \leq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0, \\ & x_2 \geq -1 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (0.61803, 0.61803)$
Objective value: $f(x^*) = -1.23606$

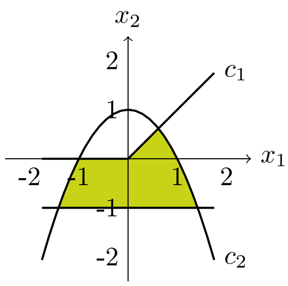

FIGURE 11. Illustration of test problem 3b

A.7. **Problem 4a.**

$$\begin{aligned} \min \quad & f(x) = -x_1 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - \min\{0, -x_1\} \geq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (1.6180, -1.6180)$
Objective value: $f(x^*) = 1.6180$

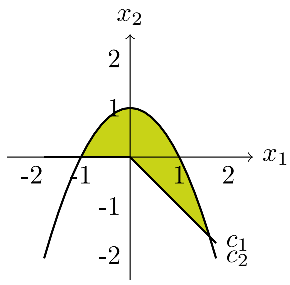

Figure 12. Illustration of test problem 4a

A.8. **Problem 4b.**

$$\begin{aligned} \min \quad & f(x) = -x_1 \\ \text{s.t.} \quad & c_1(x_1, x_2) := x_2 - \min\{0, -x_1\} \leq 0, \\ & c_2(x_1, x_2) := x_2 + x_1^2 - 1 \leq 0, \\ & x_2 \geq -1 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (1, -1)$
Objective value: $f(x^*) = -1$

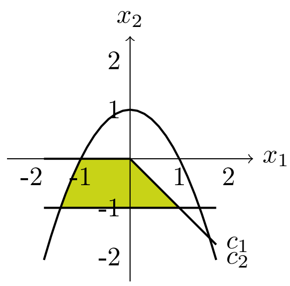

Figure 13. Illustration of test problem 4b

A.9. **Problem 5a.**

$$\begin{aligned}
\min \quad & f(x) = x_1 \\
\text{s.t.} \quad & c_1(x_1,x_2) \geq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 - (x_1-1)^2 + 1, & x_1 < 1, \\ x_2 + x_1, & x_1 \geq 1, \end{cases} \\
& x_1 \leq 2, \\
& x_2 \leq 2
\end{aligned}$$

Localization functions: $\ell_1(x_1,x_2) = x_1 - 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (-0.73205, 2)$
Objective value: $f(x^*) = -0.73205$

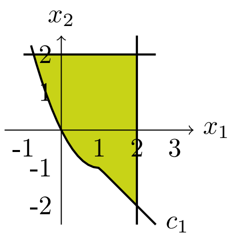

FIGURE 14. Illustration of test problem 5a

A.10. **Problem 5b.**

$$\begin{aligned}
\min \quad & f(x) = x_1 \\
\text{s.t.} \quad & c_1(x_1,x_2) \leq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 - (x_1-1)^2 + 1, & x_1 < 1, \\ x_2 + x_1, & x_1 \geq 1, \end{cases}, \\
& x_1 \geq -2, \\
& -2 \leq x_2 \leq 2
\end{aligned}$$

Localization functions: $\ell_1(x_1,x_2) = x_1 - 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (-2, x_2^*)$ with $-2 \leq x_2^2 \leq 2$
Objective value: $f(x^*) = -2$

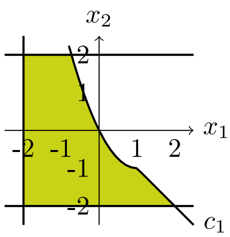

FIGURE 15. Illustration of test problem 5b

A.11. **Problem 6a.**

$$\begin{aligned} \min \quad & f(x) = -x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) \leq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 + (x_1-1)^2 + 1, & x_1 \leq 1, \\ x_2 + x_1, & x_1 > 1, \end{cases} \\ & x_2 \geq -2 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1 - 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (1, -1)$
Objective value: $f(x^*) = 1$

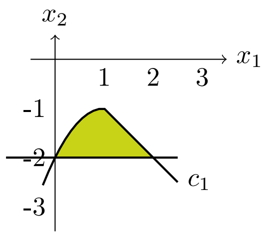

FIGURE 16. Illustration of test problem 6a

A.12. **Problem 6b.**

$$\begin{aligned} \min \quad & f(x) = x_1 + x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) \geq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 + (x_1-1)^2 + 1, & x_1 \leq 1, \\ x_2 + x_1, & x_1 > 1, \end{cases} \\ & 0 \leq x_1 \leq 2, \\ & x_2 \leq 0 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1 - 1$
Global optimal solution: $x^* = (x_1^*, x_2^*) = (0, -2)$
Global optimal objective value: $f(x^*) = -2$
Local optimal solution: $x^* = (x_1^*, x_2^*)$ with $x_1^* \in (1, 2]$ and $x_2^* = -x_1^*$.
Local optimal objective value: $f(x^*) = 0$.

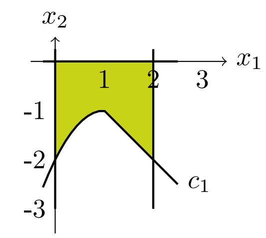

FIGURE 17. Illustration of test problem 6b

A.13. **Problem 7a.**

$$\begin{aligned}
\min \quad & f(x) = x_2 \\
\text{s.t.} \quad & c_1(x_1,x_2) \geq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 - (x_1-1)^2 - 1, & x_1 \leq 1, \\ x_2 - x_1, & x_1 > 1, \end{cases} \\
& x_2 \leq 2
\end{aligned}$$

Localization functions: $\ell_1(x_1,x_2) = x_1 - 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (1,1)$
Objective value: $f(x^*) = 1$

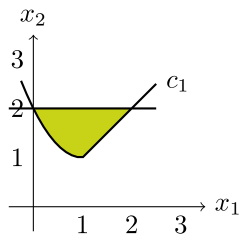

FIGURE 18. Illustration of test problem 7a

A.14. **Problem 7b.**

$$\begin{aligned}
\min \quad & f(x) = -x_1 - x_2 \\
\text{s.t.} \quad & c_1(x_1,x_2) \leq 0, \quad c_1(x_1,x_2) := \begin{cases} x_2 - (x_1-1)^2 - 1, & x_1 \leq 1, \\ x_2 - x_1, & x_1 > 1, \end{cases} \\
& 0 \leq x_1 \leq 2, \\
& x_2 \geq 0
\end{aligned}$$

Localization functions: $\ell_1(x_1,x_2) = x_1 - 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (2,2)$
Objective value: $f(x^*) = -4$

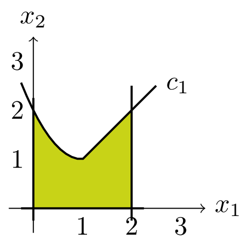

FIGURE 19. Illustration of test problem 7b

A.15. **Problem 8a.**

$$\begin{aligned} \min \quad & f(x) = -x_1 - x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) \geq 0, \quad c_1(x_1, x_2) := \begin{cases} x_2 + (x_1+1)^2 + 1, & x_1 \leq -1, \\ x_2 - x_1, & x_1 > -1, \end{cases} \\ & x_1 \geq -2, \\ & x_2 \leq 2 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1 + 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (2, 2)$
Objective value: $f(x^*) = -4$

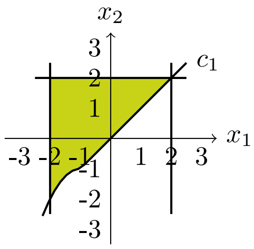

FIGURE 20. Illustration of test problem 8a

A.16. **Problem 8b.**

$$\begin{aligned} \min \quad & f(x) = x_1 + x_2 \\ \text{s.t.} \quad & c_1(x_1, x_2) \leq 0, \quad c_1(x_1, x_2) := \begin{cases} x_2 + (x_1+1)^2 + 1, & x_1 \leq -1, \\ x_2 - x_1, & x_1 > -1, \end{cases} \\ & x_1 \leq 2, \\ & x_2 \geq -2 \end{aligned}$$

Localization functions: $\ell_1(x_1, x_2) = x_1 + 1$
Optimal solution: $x^* = (x_1^*, x_2^*) = (-2, -2)$
Objective value: $f(x^*) = -4$

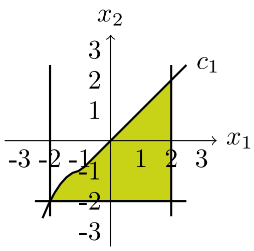

FIGURE 21. Illustration of test problem 8b

Martin Schmidt, (a) Friedrich-Alexander-Universität Erlangen-Nürnberg, Department Mathematik, Cauerstrasse 11, 91058 Erlangen, Germany; (b) Energie Campus Nürnberg, Fürther Strasse 250, 90429 Nürnberg, Germany
E-mail address: mar.schmidt@fau.de
URL: http://www.mso.math.fau.de/edom/team/schmidt-martin