Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution

Ibtihel Ben Gharbia, Joëlle Ferzly, Martin Vohralík, Soleiman yousef

Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution*

Ibtihel Ben Gharbia[†] Joëlle Ferzly[†‡§] Martin Vohralík[‡§] Soleiman Yousef[†]

June 17, 2022

Abstract

We consider nonlinear algebraic systems with complementarity constraints stemming from numerical discretizations of nonlinear complementarity problems. The particularity is that they are non-differentiable, so that classical linearization schemes like the Newton method cannot be applied directly. To approximate the solution of such nonlinear systems, an iterative linearization algorithm like the semismooth Newton-min or an interior-point algorithm can be used. Alternatively, the non-differentiable nonlinearity can be smoothed, which allows a direct application of the Newton method. Corresponding linear systems can be solved only approximately using an iterative linear algebraic solver, leading to inexact approaches. In this work, we design a general framework to systematically steer these different ingredients. We first derive an a posteriori error estimate given by the norm of the considered system's residual. We then, relying on smoothing, design a simple strategy of tightening the smoothing parameter. We finally distinguish the smoothing, linearization, and algebraic error components, which enables us to formulate an adaptive algorithm where the linear and nonlinear solvers are stopped when the corresponding error components do not affect significantly the overall error. Numerical experiments indicate that the proposed algorithm, possibly in combination with the GMRES algebraic solver, ensures important savings in terms of the number of iterations and execution time. It appears rather promising in comparison with the other methods, namely since its performance seems remarkably stable over a range of academic and industrial problems.

Key words: nonlinear complementarity constraints, semismooth smoothing Newton methods, interior-point method, a posteriori error estimate, adaptivity, stopping criteria

1 Introduction

Consider a system of algebraic equations with complementarity constraints written in the following form: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{1.1a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \boldsymbol{0},\ \boldsymbol{G}(\boldsymbol{X}) \geq \boldsymbol{0},\ \ \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0, \tag{1.1b}$$

where for two integers $n > 1$ and $0 < m < n$, $\mathbb{E} \in \mathbb{R}^{n-m,n}$ is a matrix, $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$ are (linear) operators, and $\boldsymbol{F} \in \mathbb{R}^{n-m}$ is a given vector. The first line (1.1a) typically represents the discretization of a linear partial differential equation. The second line (1.1b) then represents the complementarity constraints. It states that the vectors $\boldsymbol{K}(\boldsymbol{X})$ and $\boldsymbol{G}(\boldsymbol{X})$ have nonnegative components and are orthogonal.

Complementarity problems have important applications in many fields: economics, engineering, operations research, nonlinear analysis... In the literature, many theoretical results and numerical methods have been proposed to solve problem (1.1), see for example the books of Facchinei and Pang [23, 24], Ito and Kunisch [30], Ulbrich [43], Bonnans et al. [14], and the study of Aganagić [1].

*This project has partly received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 647134 GATIPOR).

[†]IFP Énergies nouvelles, 1 et 4 avenue du Bois Préau, 92852 Rueil-Malmaison, France (ibtihel.ben-gharbia@ifpen.fr, joelle.ferzly@ifpen.fr, soleiman.yousef@ifpen.fr).

[‡]Inria, 2 Rue Simone Iff, 75589 Paris, France (martin.vohralik@inria.fr).

[§]CERMICS, École des Ponts, 77455 Marne-la-Vallée, France

By means of so-called C-functions (C for complementarity), see [23, 24], the complementarity constraints (1.1b) can be rewritten as a system of equations $\boldsymbol{C}(\boldsymbol{X}) = \mathbf{0}$, where $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^n$ is nonlinear and semismooth. We then obtain the following equivalent formulation of problem (1.1): Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \boldsymbol{C}(\boldsymbol{X}) &= \mathbf{0}. \end{aligned} \tag{1.2}$$

A direct application of the standard Newton method to (1.2) is, however, impeded by the fact that $\boldsymbol{C}(\boldsymbol{X})$ is not differentiable. An introduction of the Clarke differential [16] allows to give a weaker differentiability meaning and leads to the class of semismooth Newton methods, with reputedly good convergence properties [33, 24, 10, 11, 12, 19, 20]. These methods are in certain cases equivalent to primal–dual active set strategies, see Hintermüller et al. [27]. Moreover, in [47], a regularized semismooth Newton method combined with a hyperplane projection technique was proposed.

Augmented Lagrangian method is one of the commonly used algorithms for constrained optimization, see, e.g., [29] and the references therein. It seeks a solution by replacing the original constrained problem by a series of unconstrained problems and add to the objective function a penalty term, and another term designed to mimic a Lagrange multiplier.

An additional technique, often used in a function space setting, consists in introducing a proper regularization, motivated by the augmented Lagrangian method. It allows to apply an infinite-dimensional semismooth Newton method for the solution of the regularized problem, see, e.g., [43]. In the present context, this leads to replacing the complementarity conditions (1.1b) by

$$\boldsymbol{K}(\boldsymbol{X}) + \min\{\mathbf{0}, -\boldsymbol{K}(\boldsymbol{X}) + \gamma \boldsymbol{G}(\boldsymbol{X})\} = \mathbf{0},$$

for a parameter $\gamma > 0$. This method can be combined with a path-following strategy to update the regularization parameter γ, see for instance [41, 28, 40].

Another important class of methods for constrained optimization problems of the form (1.1) is formed by interior-point methods. These methods consist in generating a sequence in the feasible region $\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}$, under the assumption of knowing a feasible initial point. We refer to the work of Wright [46], Bellavia et al. [4], and the references therein for a review.

Lastly, an additional notable method is the smoothing Newton method. The main idea of this approach is to approximate the semismooth (non-differentiable) function $\boldsymbol{C}$ from (1.2) by a smooth (differentiable) function that depends on a smoothing parameter. The problem is reformulated as a sequence of regularized smooth equations that can be solved by applying the standard Newton method, and where one drives the smoothing parameter down to zero, cf. [39, 36, 35] and the references therein.

In this work, we design a general framework to systematically steer the above different ingredients. Our main philosophy is adaptive smoothing (regularization). For $\mu^j > 0$, let a smoothed function $\boldsymbol{C}_{\mu^j}(\cdot)$, satisfy $\|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}) - \boldsymbol{C}(\boldsymbol{X})\| \to 0$ as $\mu^j \to 0$, for $\boldsymbol{X} \in \mathbb{R}^n$. The smoothing parameter μ^j is reduced at each smoothing iteration $j \geq 1$. Thus, problem (1.1), or equivalently (1.2), can be reformulated as a system of smooth (differentiable) equations written in the form: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X}^j &= \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) &= \mathbf{0}. \end{aligned} \tag{1.3}$$

Hence, Newton-type methods can be applied to solve system (1.3), yielding, at each linearization step $k \geq 1$, a linear system

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{1.4}$$

where $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ is a matrix and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ is a vector.

Solving (1.4) with a direct method may be very expensive. A popular approach is to solve it approximately by applying only a few steps of an iterative algebraic solver. Such inexact approaches can be found in [22, 32] for semismooth Newton methods, in [39, 26] for smoothing Newton methods, in [49] for augmented Lagrangian methods, and in [3] for interior-point methods. In the algorithms introduced therein, the iterations of different solvers are stopped according to a fixed maximal number of iterations, the Euclidean norm of the residual vector, or other parameters-dependant stopping criteria. In this work, the a posteriori estimate constitute a distinctive element at the heart of the proposed smoothing method. Importantly, it ensures the desired balance between each source of error at any resolution step, unlike existing approaches based on classical stopping criteria.

At each linear algebraic step $i \geq 1$ for (1.4), one in particular obtains $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\text{alg}}^{j,k,i},$$

where $\boldsymbol{R}_{\rm alg}^{j,k,i} \in \mathbb{R}^n$ is the algebraic residual vector of (1.4).

Our principal aim is to reduce the computational cost of the numerical resolution of (1.1) by employing an adaptive strategy based on a posteriori error estimates. There is a well-developed literature on a posteriori error estimates and *mesh adaptivity* for partial differential equations, see for instance the books of Ainsworth and Oden [2], Repin [37], and Nochetto et al. [34]. For variational inequalities, we can mention the contributions of Repin [38], Ben Belgacem et al. [5], Bürg and Schröder [15], and Dabaghi et al. [17]. Although smoothing Newton approaches have been widely studied, to the best of our knowledge, almost no work has been done to this day on a posteriori error estimates and adaptivity for *solvers* applied to discrete problems of the form (1.1).

We first derive an upper bound on the norm of the residual of system (1.2), given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}.$$

Then, decomposing $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$, we distinguish the different error components. This leads to an a posteriori control of the form

$$\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\rm r} \le \eta^{j,k,i} = \eta_{\rm sm}^{j,k,i} + \eta_{\rm lin}^{j,k,i} + \eta_{\rm alg}^{j,k,i}. \tag{1.5}$$

Here, $\eta^{j,k,i}$ is a fully computable upper bound that holds true at any smoothing (regularization) step j, linearization step k, and algebraic solver step i, whereas the role of the estimators $\eta_{\rm sm}^{j,k,i}, \eta_{\rm lin}^{j,k,i}$, and $\eta_{\rm alg}^{j,k,i}$ is to identify the smoothing, linearization, and algebraic components of the error. This error bound allows to define adaptive stopping criteria for the nonlinear and linear algebraic solvers, in the spirit of [21, 17], and the references therein. These criteria, as well as a simple way to tighten the smoothing parameter μ^j, are incorporated in a three-level adaptive algorithm. In contrast to common approaches, where the termination requires reaching a fixed threshold, the particularity of this adaptive algorithm is that the iterations are stopped when the error component of the concerned solver is smaller than the total error, up to a desired fraction. The efficiency of the proposed adaptive algorithm for (inexact) smoothing Newton methods and (inexact) interior-point methods is showcased numerically on practical problems.

It is relevant to mention that this work is extented in [8], where the present approach is applied to a system of PDEs with complementarity constraints in infinite-dimensional space. In particular, taking into account the discretization error allows to adaptively steer the smoothing in system (1.3). Although we do not address mesh adaptivity in our work, we underline that a posteriori estimators are an important tool for adaptive mesh refinement strategies, see, e.g., [18] and the references therein. Consequently, algorithms based on the previous criteria ensure significant computational gains in terms of total number of iterations and mesh cells.

Our manuscript is organized as follows. In Section 2, we recall a semismooth Newton method based on an equivalent reformulation of the complementarity constraints in the form (1.2). Section 3 is devoted to introduce our adaptive inexact smoothing Newton method based on the reformulation as a system of smooth equations as in (1.3). We establish here the a posteriori error estimates (1.5) and propose an adaptive algorithm with a posteriori stopping criteria. We survey a nonparametric interior-point method in Section 4, and introduce its adaptive version in Section 5. Finally, a detailed numerical study is presented in Sections 6 and 7.

2 Semismooth Newton method

The purpose of this section is to briefly recall the semismooth Newton method to approximate the solution of the nonlinear system of equations (1.1), see, e.g., [33, 23, 17]. The complementarity constraints represented by (1.1b) as algebraic inequalities are here rewritten as non-differentiable algebraic equalities, using a complementarity function (C-function). A function $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m,\ m \ge 1$, is called a C-function if

$$\tilde{\boldsymbol{C}}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0} \iff \boldsymbol{x} \ge \boldsymbol{0},\ \boldsymbol{y} \ge \boldsymbol{0},\ \boldsymbol{x} \cdot \boldsymbol{y} = 0 \qquad \forall (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m.$$

A variety of C-functions can be found in the literature, see, e.g., [42, 25]. We give as examples the minimum (min) function and the Fischer–Burmeister (F–B) function: for $l = 1, \ldots, m$,

$$\left(\tilde{\boldsymbol{C}}_{\rm min}(\boldsymbol{x}, \boldsymbol{y})\right)_l := (\min(\boldsymbol{x}, \boldsymbol{y}))_l = (\boldsymbol{x}_l + \boldsymbol{y}_l)/2 - |\boldsymbol{x}_l - \boldsymbol{y}_l|/2, \tag{2.1}$$

$$\left(\tilde{\boldsymbol{C}}_{\rm FB}(\boldsymbol{x}, \boldsymbol{y})\right)_l := \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l). \tag{2.2}$$

In general, the C-functions are not Fréchet differentiable. The min and the Fischer–Burmeister functions are, for example, differentiable everywhere except in $\boldsymbol{x} = \boldsymbol{y}$ and $(\boldsymbol{0}, \boldsymbol{0})$, respectively. Let us introduce a function

$\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined as $\boldsymbol{C}(\boldsymbol{X}) := \tilde{\boldsymbol{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, where $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any C-function. By using this reformulation in (1.1b), it is obvious that problem (1.1) can be equivalently rewritten as: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$, such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{2.3a}$$

$$\boldsymbol{C}(\boldsymbol{X}) = \boldsymbol{0}. \tag{2.3b}$$

Next, we detail the semismooth Newton linearization. Let an initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$ be given. At the step $k \geq 1$, one looks for $\boldsymbol{X}^k \in \mathbb{R}^n$ such that

$$\mathbb{A}^{k-1}\boldsymbol{X}^k = \boldsymbol{B}^{k-1}, \tag{2.4}$$

where the square matrix $\mathbb{A}^{k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}^{k-1} \in \mathbb{R}^n$ are given by

$$\mathbb{A}^{k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1}) \end{bmatrix}, \quad \boldsymbol{B}^{k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1})\boldsymbol{X}^{k-1} - \boldsymbol{C}(\boldsymbol{X}^{k-1}) \end{bmatrix}. \tag{2.5}$$

Note that the Jacobian corresponding to (2.3a) is constant and equal to $\mathbb{E}$ since it is linear. The semismooth nonlinearity occurs in the second line (2.3b): the notation $\mathbf{J}_{\boldsymbol{C}}$ in (2.5) stands for the Jacobian matrix in the sense of Clarke of the function $\boldsymbol{C}$, cf. [23, 24]. To give an example, consider the semismooth min function (2.1) and define the matrices $\mathbb{K}$ and $\mathbb{G} \in \mathbb{R}^{m,n}$ respectively by $\mathbb{K} := [\nabla \boldsymbol{K}(\boldsymbol{X})]$ and $\mathbb{G} := [\nabla \boldsymbol{G}(\boldsymbol{X})]$. Then the l^{th} row of the Jacobian matrix in the sense of Clarke $\mathbf{J}_{\boldsymbol{C}}$ is either given by the l^{th} row of $\mathbb{K}$, if $(\boldsymbol{K}(\boldsymbol{X}^{k-1}))_l \leq (\boldsymbol{G}(\boldsymbol{X}^{k-1}))_l$, or by the l^{th} row of $\mathbb{G}$, if $(\boldsymbol{G}(\boldsymbol{X}^{k-1}))_l < (\boldsymbol{K}(\boldsymbol{X}^{k-1}))_l$.

We will need below the total residual vector of problem (2.3), defined by

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}(\boldsymbol{V}) \end{bmatrix}, \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{2.6}$$

In this context, the relative norm of a vector $\boldsymbol{V} \in \mathbb{R}^n$ is given by $||\boldsymbol{V}||_{\mathrm{r}} := ||\boldsymbol{V}|| \,/\, \left|\left|\boldsymbol{R}(\boldsymbol{X}^0)\right|\right|$, where $\|\cdot\|$ is the L_2-norm.

3 Adaptive inexact smoothing Newton method

In this section we introduce our adaptive inexact smoothing Newton method. Based on a posteriori error estimators, adaptive stopping criteria are formulated to conceive an adaptive iterative algorithm.

3.1 Smoothing of the C-functions

The key of our developments is to smooth the non-differentiable equation formulation (2.3b) of the complementarity constraints (1.1b) with the help of a smooth (i.e. continuously differentiable) function. This smoothing allows us to approximately transform the nonsmooth nonlinear system (2.3) to a smooth system of nonlinear equations to be solved by using the standard Newton method.

Let $\mu > 0$ be a (small) smoothing parameter. We construct an approximation function $\tilde{\boldsymbol{C}}_\mu : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ of a C-function $\tilde{\boldsymbol{C}}$ such that $\tilde{\boldsymbol{C}}_\mu(\cdot,\cdot)$ is of class $\mathcal{C}^1$ on $\mathbb{R}^m \times \mathbb{R}^m$ and satisfies

$$\|\tilde{\boldsymbol{C}}(\boldsymbol{x},\boldsymbol{y}) - \tilde{\boldsymbol{C}}_\mu(\boldsymbol{x},\boldsymbol{y})\| \to 0 \text{ as } \mu \to 0 \quad \text{for all } (x,y) \in \mathbb{R}^m \times \mathbb{R}^m.$$

For example, for $l = 1, \ldots, m$, a possible smoothing of the min and the Fischer–Burmeister functions (2.1) and (2.2) can be

$$\left(\tilde{\boldsymbol{C}}_{\min_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{\left(|\boldsymbol{x}-\boldsymbol{y}|_\mu\right)_l}{2}, \quad \text{with } \left(|\boldsymbol{z}|_\mu\right)_l = \sqrt{\boldsymbol{z}_l^2 + \mu^2}, \tag{3.1}$$

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l), \tag{3.2}$$

where the μ-smoothed absolute value function $|\cdot|_\mu : \mathbb{R}^m \to \mathbb{R}^m_+,\ m \geq 0$, replaces the absolute value function (not differentiable at $\boldsymbol{0}$), see Figure 1. Note that both functions $|\cdot|_\mu$ and $\tilde{\boldsymbol{C}}_{\mathrm{FB},\mu}$ are of class $\mathcal{C}^\infty$.

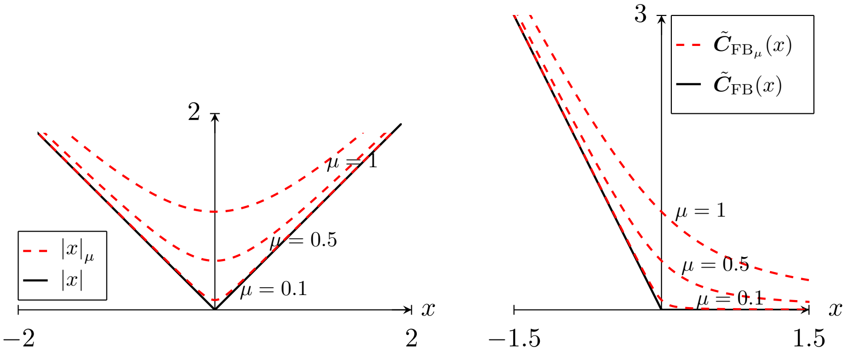

Figure 1: Left: Absolute value function $|\cdot|$ and smoothed absolute value function $|\cdot|_\mu$. Right: Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\cdot)$ and smoothed Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\cdot)$, for different values of the smoothing parameter μ.

We define the function $\boldsymbol{C}_\mu : \mathbb{R}^n \to \mathbb{R}^m$ as $\boldsymbol{C}_\mu(\boldsymbol{X}) := \tilde{\boldsymbol{C}}_\mu\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right),$ where $\tilde{\boldsymbol{C}}_\mu : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any smoothed C-function of at least class $\mathcal{C}^1$. This allows to approximate problem (1.1) or (1.2) by a system of smooth equations: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$, such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \boldsymbol{C}_\mu(\boldsymbol{X}) &= \boldsymbol{0}. \end{aligned} \tag{3.3}$$

Thus, Newton-type methods can be applied to solve the system of nonlinear algebraic equations (3.3).

Fixing $\mu^1 > 0$, we now describe an iterative method for solving problem (2.3). At the beginning of each smoothing iteration (outer iteration) denoted hereafter by $j \geq 1$, an initial guess $\boldsymbol{X}^j \in \mathbb{R}^n$ is given, and a smoothing parameter μ^j is determined; μ^j will be driven down to zero. Then some iterative nonlinear solver like the Newton method is employed to solve the smoothed problem written in the form: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X}^j &= \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) &= \boldsymbol{0}. \end{aligned} \tag{3.4}$$

3.2 Newton linearization of the nonlinear algebraic system

In what follows, we detail the Newton method employed to solve problem (3.4) at a fixed outer smoothing step $j \geq 1$. Given an initial vector $\boldsymbol{X}^{j,0}$ (typically $\boldsymbol{X}^{j,0} = \boldsymbol{X}^{j-1}$), Newton's algorithm generates a sequence $(\boldsymbol{X}^{j,k})_{k\geq 1}$ with $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ given by the following system of linear algebraic equations

$$\mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{3.5}$$

where the Jacobian matrix $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_{\mu^j}^{j,k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \qquad \boldsymbol{B}_{\mu^j}^{j,k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})\boldsymbol{X}^{j,k-1} - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \tag{3.6}$$

with $\mathbf{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})$ the Jacobian matrix of the smooth function $\boldsymbol{C}_{\mu^j}$ at $\boldsymbol{X}^{j,k-1}$.

3.3 Inexact solution of the linear algebraic system

The linearized system (3.5) may not be solved exactly, since the use of a direct method may be expensive. For this reason, we consider in this work also an inexact resolution. For a fixed smoothing step $j \geq 1$, a fixed Newton step $k \geq 1$, and an initial guess $\boldsymbol{X}^{j,k,0}$ (typically $\boldsymbol{X}^{j,k,0} = \boldsymbol{X}^{j,k-1}$), only a few steps of an iterative linear algebraic solver can be applied to find an approximate solution to (3.5), yielding, on step $i \geq 1$, an approximate $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$. This satisfies (3.5) up to the residual vector given by

$$\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,i}. \tag{3.7}$$

Define now the linearization function $\boldsymbol{C}^{j,k-1}_{\mu^j} : \mathbb{R}^n \to \mathbb{R}^m$ of $\boldsymbol{C}_{\mu^j}$ at smoothing step j and Newton step k as

$$\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{V}) := \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}) \qquad \forall\, \boldsymbol{V} \in \mathbb{R}^n. \tag{3.8}$$

This allows us to write the algebraic residual vector for $\boldsymbol{V} \in \mathbb{R}^n$ as

$$\boldsymbol{R}^{\mathrm{AISN}}_{\mathrm{alg}}(\boldsymbol{V}) := \boldsymbol{B}^{j,k-1}_{\mu^j} - \mathbb{A}^{j,k-1}_{\mu^j}\boldsymbol{V} = \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{V}) \end{bmatrix}. \tag{3.9}$$

3.4 An upper bound for the norm of the residual

We consider the total residual vector of problem (2.3) given in (2.6). By adding and subtracting $\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ given by (3.8), the total residual vector can be decomposed as follows:

$$\begin{aligned}
\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) &= \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \pm \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix} \\
&= \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothing}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}} + \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}.
\end{aligned}$$

It is reasonable to get these three terms. Indeed, the first one reflects the error due to the approximation of the semismooth function $\boldsymbol{C}$ by the smoothed function $\boldsymbol{C}_{\mu^j}$. The second term is related to the linearization of the nonlinear smooth problem (3.4). Taking into account that the resolution of the smooth linearized problem (3.5) is possibly done "inexactly", the remaining term represents the error of the inexact algebraic resolution. By the triangle inequality, the relative norm of $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$ is thus bounded by the smoothing, linearization, and algebraic estimators respectively defined as

$$\eta^{j,k,i}_{\mathrm{sm,AISN}} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}(\boldsymbol{X}^{j,k,i})\right|\right|_{\mathrm{r}}, \tag{3.10a}$$

$$\eta^{j,k,i}_{\mathrm{lin,AISN}} := \left|\left|\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right|\right|_{\mathrm{r}}, \tag{3.10b}$$

$$\eta^{j,k,i}_{\mathrm{alg,AISN}} := \left(\left|\left|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\right|\right|^2_{\mathrm{r}} + \left|\left|\boldsymbol{C}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right|\right|^2_{\mathrm{r}}\right)^{\frac{1}{2}}. \tag{3.10c}$$

Note that $\eta^{j,k,i}_{\mathrm{alg,AISN}}$ is exactly equal to the relative norm of $\boldsymbol{R}^{\mathrm{AISN}}_{\mathrm{alg}}(\boldsymbol{X}^{j,k,i})$ given by (3.9). From these developments we conclude:

Theorem 3.1 *Let $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ arise from an inexact solve of (3.5). We have*

$$\left|\left|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\right|\right|_{\mathrm{r}} \leq \eta^{j,k,i}_{\mathrm{AISN}} := \eta^{j,k,i}_{\mathrm{sm,AISN}} + \eta^{j,k,i}_{\mathrm{lin,AISN}} + \eta^{j,k,i}_{\mathrm{alg,AISN}}.$$

3.5 Adaptive inexact smoothing Newton algorithm

Theorem 3.1 motivates the following. Let two real parameters α_{lin} and α_{alg} be given in $]0,1[$, representing the desired relative size of the algebraic and linearization errors, and let $\varepsilon > 0$ be a given desired tolerance for the total error. The stopping criteria for the linearization, algebraic, and smoothing steps, with the bars denoting the stopping indices, are respectively set as

$$\eta^{j,k,\bar{i}}_{\mathrm{alg,AISN}} < \alpha_{\mathrm{alg}} \eta^{j,k,\bar{i}}_{\mathrm{lin,AISN}}, \tag{3.11a}$$

$$\eta^{j,\bar{k},\bar{i}}_{\mathrm{lin,AISN}} < \alpha_{\mathrm{lin}} \eta^{j,\bar{k},\bar{i}}_{\mathrm{sm,AISN}}, \tag{3.11b}$$

$$\|\boldsymbol{R}(\boldsymbol{X}^{\bar{j},\bar{k},\bar{i}})\|_{\mathrm{r}} < \varepsilon. \tag{3.11c}$$

The first criterion (3.11a) for the algebraic iterative solver expresses that there is no need to continue with the algebraic steps when the linearization error becomes dominant. Similarly, the second one (3.11b) aims at stopping the linearization iterations when the linearization error does not substantially contribute to the smoothing error. Finally, the termination criterion for the smoothing steps (3.11c) is of the standard type, that is when we stop the entire procedure, when the relative norm of the total residual vector lies below the desired tolerance ε.

The entire method is described by the following adaptive algorithm, which drives the smoothing parameter μ^j to zero as $\mu^j := \alpha\mu^{j-1}$ at each smoothing iteration. Other common empirical ways to progressively reduce μ^j can be found, e.g., in [45]. The adaptive inexact smoothing Newton algorithm is the following:

Algorithm 1 Adaptive inexact smoothing Newton algorithm

1. Initialization
Choose a tolerance $\varepsilon > 0$ and parameters $\alpha \in]0,1[$ and $\alpha_{\rm lin}, \alpha_{\rm alg} \in]0,1]$.
Fix $\mu^1 > 0$ and an initial approximation $\boldsymbol{X}^0 \in \mathbb{R}^n$. Set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^0$ as an initial guess for the nonlinear solver. Set $k := 1$.

2.2 Newton linearization loop

2.2.1 From $\boldsymbol{X}^{j,k-1}$ define $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ by (3.6).

2.2.2 Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{3.12}$$

2.2.3 Set $\boldsymbol{X}^{j,k,0} := \boldsymbol{X}^{j,k-1}$ as initial guess for the iterative algebraic solver. Set $i := 1$.

2.2.4 Algebraic solver loop

i) Starting from $\boldsymbol{X}^{j,k-1}$, perform a step of the iterative algebraic solver for the solution of (3.12), yielding, on step i an approximation $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$ satisfying

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\rm alg}^{\rm AISN}(\boldsymbol{X}^{j,k,i}).$$

ii) Compute the estimators given in (3.10).

iii) If $\eta_{\rm alg,AISN}^{j,k,i} < \alpha_{\rm alg} \eta_{\rm lin,AISN}^{j,k,i}$, set $\bar{i} := i$ and stop. If not, set $i := i+1$ and go to **i)**.

2.2.5 If $\eta_{\rm lin,AISN}^{j,k,\bar{i}} < \alpha_{\rm lin} \eta_{\rm sm,AISN}^{j,k,\bar{i}}$, set $\bar{k} := k$ and stop. If not, set $k := k+1$ and go to **2.2.1**.

2.3 If $\|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\|_{\rm r} < \varepsilon$, set $\bar{j} := j$ and stop.
If not, set $j := j+1$, $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$, and $\mu^j := \alpha \mu^{j-1}$. Then set $k := 1$ and go to **2.2.1**.

4 Nonparametric interior-point method

Now we employ a nonparametric interior-point method to problem (1.1). More precisely, we consider the method introduced in [44] where a systematic strategy is used to steer the sequence of smoothing parameters towards zero.

We introduce a vector $\boldsymbol{\mu} = \mu \mathbf{1} \in \mathbb{R}^m$, where $\mu > 0$ is the smoothing parameter and $\mathbf{1} \in \mathbb{R}^m$ is the vector with all components equal to 1. The original nonsmooth problem (1.1) is replaced by a smoothed problem written in the form: Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{4.1a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \;\; \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \;\;\; \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \boldsymbol{\mu}, \tag{4.1b}$$

where $[(\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X})]_m = [\boldsymbol{K}(\boldsymbol{X})]_m[\boldsymbol{G}(\boldsymbol{X})]_m$. In order to properly adjust the sequence of smoothing parameters, the smoothing parameter μ is treated as an unknown, by introducing the following new equation into system (4.1)

$$\theta\mu + \mu^2 = 0, \tag{4.2}$$

where θ is a small positive real parameter, chosen once and for all. This equation prevents μ from rushing to zero in just one iteration, and ensures quadratic convergence, see [44]. The unknown of system (4.1) is now the enlarged vector $\boldsymbol{\mathcal{X}} = (\boldsymbol{X}, \mu)^T \in \mathbb{R}^{n+1}$. We are thus brought back to applying the standard Newton method to a smooth problem.

Let $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \mathbf{0}$ be given. To update the iterate $\boldsymbol{\mathcal{X}}^{k-1}$, we compute a search direction denoted by $\boldsymbol{d}^k = [\boldsymbol{d}^k_{\boldsymbol{X}}, d^k_\mu] \in \mathbb{R}^{n+1}$, where $\boldsymbol{d}^k_{\boldsymbol{X}} \in \mathbb{R}^n$ and $d^k_\mu \in \mathbb{R}$. Then, to preserve positivity of $\boldsymbol{K}(\boldsymbol{X}^k)$ and $\boldsymbol{G}(\boldsymbol{X}^k)$ at each step of the nonlinear solver, a truncation of the Newton direction $\boldsymbol{d}^k$ is performed so that the corresponding update satisfies $\boldsymbol{K}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}^k_{\boldsymbol{X}}) \geq 0$ and $\boldsymbol{G}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}^k_{\boldsymbol{X}}) \geq 0$ for some $\kappa^k \in]0, 1]$, as close to 1 as possible. After this, we can set

$$\boldsymbol{\mathcal{X}}^k := \boldsymbol{\mathcal{X}}^{k-1} + \kappa^k \boldsymbol{d}^k.$$

Recall that our goal is to make μ equal to 0 in the limit while ensuring the positivity of the updated iterate. Another choice for the additional equation (4.2) added to system (4.1) was developed and introduced in a recent work, see [45, Section 3]. The proposed equation does not require to truncate the Newton direction, and couples μ and $\boldsymbol{X}$ in a tighter way.

We rewrite system (4.1) as an enlarged system of $n + 1$ equations

$$\begin{aligned} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \boldsymbol{K}(X)\boldsymbol{G}(X) - \boldsymbol{\mu} &= \mathbf{0}, \\ \theta\mu + \mu^2 &= 0. \end{aligned} \tag{4.3}$$

5 Adaptive inexact interior-point method

We present in this section our adaptive inexact version of the nonparametric interior point method of Section 4. In contrast to Section 4, we consider, however, $\mu > 0$ as a parameter, and not as an unknown. At each smoothing step $j \geq 1$, we may solve the system of smoothing equations written as: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^j) \geq \mathbf{0}, \; \boldsymbol{G}(\boldsymbol{X}^j) \geq \mathbf{0}$, and

$$\mathbb{E}\boldsymbol{X}^j = \boldsymbol{F}, \tag{5.1a}$$

$$\boldsymbol{H}_{\mu^j}(\boldsymbol{X}^j) := \boldsymbol{K}(\boldsymbol{X}^j)\boldsymbol{G}(\boldsymbol{X}^j) - \boldsymbol{\mu}^j = \mathbf{0}. \tag{5.1b}$$

The values of μ^j are gradually decreased at each smoothing iteration, creating a sequence of suitable μ^j converging to zero.

5.1 Newton linearization of the nonlinear algebraic system

Let $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \mathbf{0}$ be given. At each smoothing iteration $j \geq 1$ and each linearization step $k \geq 1$, starting with an initial approximation $\boldsymbol{X}^{j,0}$ such that $\boldsymbol{K}(\boldsymbol{X}^{j,0}) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^{j,0}) \geq \mathbf{0}$ (typically $\boldsymbol{X}^{j,0} = \boldsymbol{X}^{j-1}$), we try to approach the solution of problem (5.1) by finding $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}^{j,k-1}_{\mu^j} \boldsymbol{X}^{j,k} = \boldsymbol{B}^{j,k-1}_{\mu^j}, \tag{5.2}$$

where the Jacobian matrix $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_{\mu^j}^{j,k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \qquad \boldsymbol{B}_{\mu^j}^{j,k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1})\boldsymbol{X}^{j,k-1} - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \tag{5.3}$$

with $\mathbf{J}_{\boldsymbol{H}_{\mu^j}}$ the Jacobian matrix of $\boldsymbol{H}_{\mu^j}$. To ensure the positivity of the complementarity constraints, we then define the direction $\boldsymbol{d}^{j,k} := \boldsymbol{X}^{j,k} - \boldsymbol{X}^{j,k-1} \in \mathbb{R}^n$ and find $\kappa^{j,k} \in]0,1]$ such that

$$\boldsymbol{K}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k}\boldsymbol{d}^{j,k}) \geq \mathbf{0} \quad \text{and} \quad \boldsymbol{G}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k}\boldsymbol{d}^{j,k}) \geq \mathbf{0}.$$

5.2 Inexact solution of the linear algebraic system

For a fixed smoothing iteration $j \geq 1$, a fixed Newton step $k \geq 1$, and an initial guess $\boldsymbol{X}^{j,k,0}$ (typically $\boldsymbol{X}^{j,k,0} = \boldsymbol{X}^{j,k-1}$), an iterative algebraic solver can be applied to approach the solution of (5.2), yielding, on step $i \geq 1$, an approximation $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$. This satisfies (5.2) up to a residual vector defined by

$$\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,i}. \tag{5.4}$$

Introduce the linearization $\boldsymbol{H}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$ of $\boldsymbol{H}_{\mu^j}(\cdot)$ such that for $\boldsymbol{V} \in \mathbb{R}^n$,

$$\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}). \tag{5.5}$$

Using (5.5), the algebraic residual vector can be written as follows

$$\boldsymbol{R}_{\text{alg}}^{\text{AIIP}}(\boldsymbol{V}) := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{V} = \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{V}) \end{bmatrix}, \qquad \boldsymbol{V} \in \mathbb{R}^n. \tag{5.6}$$

We now define the function $\boldsymbol{H} : \mathbb{R}^n \to \mathbb{R}^m$ by

$$\boldsymbol{H}(\boldsymbol{V}) := \boldsymbol{K}(\boldsymbol{V})\boldsymbol{G}(\boldsymbol{V}), \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{5.7}$$

and the total residual vector associated to the adaptive inexact interior-point method by

$$\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}(\boldsymbol{V}) \end{bmatrix}, \qquad \boldsymbol{V} \in \mathbb{R}^n. \tag{5.8}$$

Here again, the relative norm of a given vector $\boldsymbol{V} \in \mathbb{R}^n$ is given by $||\boldsymbol{V}||_{\text{r}} := ||\boldsymbol{V}|| / \|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^0)\|$.

5.3 An upper bound for the norm of the residual

In the same spirit as in Section 3.4, we decompose at each smoothing step $j \geq 1$, each linearization step $k \geq 1$, and each algebraic step $i \geq 1$ the total residual vector given by (5.8)

$$\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i}) = \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{H}_{\boldsymbol{\mu}^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothing}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}}$$
$$+ \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}.$$

We then define the smoothing, linearization, and algebraic estimators by

$$\eta_{\text{sm,AIIP}}^{j,k,i} := \left|\left|\boldsymbol{H}_{\boldsymbol{\mu}^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}} = \left|\left|\boldsymbol{\mu}^j\right|\right|_{\text{r}}, \tag{5.9a}$$

$$\eta_{\text{lin,AIIP}}^{j,k,i} := \left|\left|\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}, \tag{5.9b}$$

$$\eta_{\text{alg,AIIP}}^{j,k,i} := \left(\|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\|_{\text{r}}^2 + \|\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})\|_{\text{r}}^2\right)^{\frac{1}{2}}. \tag{5.9c}$$

Then we have an upper bound for the norm $\left|\left|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}$:

Theorem 5.1 *Let $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ be the approximation of $\boldsymbol{X}$ given by an iterative algebraic solver. Then we have*

$$\left|\left|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}} \leq \eta_{\text{AIIP}}^{j,k,i} := \eta_{\text{sm,AIIP}}^{j,k,i} + \eta_{\text{lin,AIIP}}^{j,k,i} + \eta_{\text{alg,AIIP}}^{j,k,i}.$$

5.4 Adaptive inexact interior-point algorithm

Our proposed adaptive inexact interior-point algorithm implements adaptive stopping criteria formulated using the error component estimators given by (5.9) is as follows:

Algorithm 2 Adaptive inexact interior-point algorithm

---

1. **Initialization**
   Choose a tolerance $\varepsilon > 0$ and parameters $\alpha \in]0,1[$ and $\alpha_{\text{lin}}, \alpha_{\text{alg}} \in]0,1]$.
   Fix $\mu^1 > 0$ and an initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \boldsymbol{0}$. Set $j := 1$.

2. **Smoothing loop**

   2.1 Set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^0$ as an initial guess for the linearization loop and $k := 1$.

   2.2 Interior-point linearization loop

   2.2.1 From $\boldsymbol{X}^{j,k-1}$ define $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ by (5.3).

   2.2.2 Consider the problem of finding $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{5.10}$$

   2.2.3 Set $\boldsymbol{X}^{j,k,0} := \boldsymbol{X}^{j,k-1}$ as initial guess for the iterative algebraic solver. Set $i := 1$.

   2.2.4 Algebraic solver loop

   i) Starting from $\boldsymbol{X}^{j,k-1}$ perform a step of the iterative algebraic solver for (5.10), yielding, at step $i \geq 1$, a vector $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\text{alg}}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i}).$$

   ii) Set $\boldsymbol{d}^{j,k,i} := \boldsymbol{X}^{j,k} - \boldsymbol{X}^{j,k-1}$ and compute $\kappa^{j,k,i} \in]0,1]$ such that

$$\boldsymbol{K}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}) \geq \boldsymbol{0} \text{ and } \boldsymbol{G}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}) \geq \boldsymbol{0}.$$

   Then set $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}$.

   iii) Compute the estimators given by (5.9).

   iv) If $\eta_{\text{alg,AIIP}}^{j,k,i} < \alpha_{\text{alg}} \eta_{\text{lin,AIIP}}^{j,k,i}$, set $\bar{i} := i$ and stop. If not, set $i := i + 1$ and go to **i)**.

   2.2.5 If $\eta_{\text{lin,AIIP}}^{j,k,\bar{i}} < \alpha_{\text{lin}} \eta_{\text{sm,AIIP}}^{j,k,\bar{i}}$, set $\bar{k} := k$ and stop. If not, set $k := k + 1$ and go to **2.2.1**.

   2.3 If $\left|\left|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\right|\right|_{\text{r}} < \varepsilon$, set $\bar{j} := j$ and stop. If not, set $j := j + 1$, $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$, and $\mu^j := \alpha\mu^{j-1}$. Then set $k := 1$ and go to **2.2.1**.

---

6 Numerical experiments: Problem of contact between two membranes

This section reports some numerical illustrations obtained using the algorithms previously presented. We consider here the model problem of contact between two membranes.

6.1 Problem statement

Let $\Omega = (a, b)$ be a one-dimensional domain. The problem reads: Find u_1, u_2, and λ such that

$$\left\{\begin{array}{rcll} -\mu_1 \Delta u_1 - \lambda &=& f_1 & \text{in } \Omega, \\ -\mu_2 \Delta u_2 + \lambda &=& f_2 & \text{in } \Omega, \\ (u_1 - u_2)\lambda &=& 0, & u_1 - u_2 \geq 0, \quad \lambda \geq 0 \quad \text{in } \Omega, \\ u_1 &=& g & \text{on } \partial\Omega, \\ u_2 &=& 0 & \text{on } \partial\Omega, \end{array}\right. \tag{6.1}$$

where u_1 and u_2 represent the vertical displacements of the two membranes and λ is a Lagrange multiplier characterizing the action of the second membrane on the first one, $-\lambda$ being the reaction. The constant parameters $\mu_1, \mu_2 > 0$ correspond to the tension of each membrane, whereas $f_1, f_2 \in L^2(\Omega)$ are given external forces. The boundary condition prescribed by a constant $g > 0$ ensures that, on the boundary $\partial\Omega$, the first membrane is above the second one. The third line of (6.1) represents the linear complementarity conditions which serve to distinguish two different physical situations: either the membranes are separated ($u_1 > u_2$ and $\lambda = 0$), or they are in contact ($u_1 = u_2$ and $\lambda > 0$). We discretize this problem by the finite volume method. The corresponding discretization can be written under the form of problem (1.1).

6.2 Test problem setting

Following [5], we set $\Omega = (-1, 1)$ and consider the following analytical solution for $x \in \Omega$

$$u_1(x) := g(2x^2 - 1), \quad u_2(x) := \begin{cases} 2g(1 - x^2)(2x^2 - 1) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ g(2x^2 - 1) & \text{otherwise,} \end{cases}$$

$$\lambda(x) := \begin{cases} 0 & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ 2g & \text{otherwise.} \end{cases}$$

This triple is the solution of (6.1) for the data f_1 and f_2 given by

$$f_1(x) := \begin{cases} -4g & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -6g & \text{otherwise,} \end{cases} \quad \text{and} \quad f_2(x) := \begin{cases} -12g(1 - 4x^2) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -2g & \text{otherwise.} \end{cases}$$

Throughout the computational experiments, the parameters μ_1 and μ_2 are set to 1 and the boundary condition g for the first membrane is taken equal to 0.1. Let N be the number of mesh elements. The initial guess $\boldsymbol{X}^0 \in \mathbb{R}^{3N}$ has its first N components equal to g and its other components equal to zero for the semismooth and smoothing Newton methods. For the nonparametric interior-point method (resp. the adaptive interior-point method), the initialization is given by $\boldsymbol{\mathcal{X}}^0 = [\mathbf{0.1\ 0\ 0.5\ 0.05}]^T \in \mathbb{R}^{3N+1}$ (resp. $\boldsymbol{X}^0 = [\mathbf{0.1\ 0\ 0.5}]^T \in \mathbb{R}^{3N}$). All the simulations are performed in MATLAB. We consider $N = 25000$ elements, leading to the matrix $\mathbb{A}$ of size $n = 75000$.

6.3 Semismooth Newton method

We start by presenting the numerical results of the semismooth Newton method described in Section 2, using the F–B function (2.2). The stopping criterion is on the total residual vector (2.6)

$$\|\boldsymbol{R}(\boldsymbol{X}^k)\|_{\mathrm{r}} < 10^{-8}. \tag{6.2}$$

To achieve this stopping criterion, 527 semismooth Newton-F–B iterations (CPU time: 68.9s) and 2232 Newton-min iterations (CPU time: 338.9s) are needed. Figure 2 represents the evolution of $\|\boldsymbol{R}(\boldsymbol{X}^k)\|_{\mathrm{r}}$ as a function of the semismooth Newton-F–B iterations. We can see that it decreases slowly during iterations, then the convergence gets extremely fast at the end.

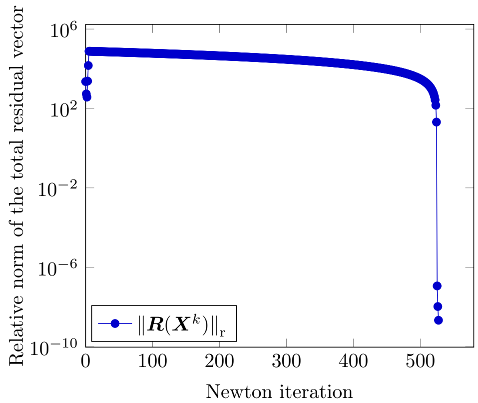

Figure 2: [Semismooth Newton method, F–B function (2.2), stopping criterion (6.2)] Relative norm of the total residual vector (2.6) as a function of semismooth Newton iterations.

6.4 Adaptive smoothing Newton method

We now test the adaptive smoothing Newton method, denoted by ASN, with the smoothed F–B function (3.2). This consists in employing the method presented in Section 3, summarized in Algorithm 1, but with an exact resolution of the nonlinear system (3.5). The linearization and smoothing estimators are respectively defined by

$$\eta^{j,k}_{\mathrm{lin,ASN}} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k})\right|\right|_{\mathrm{r}}, \tag{6.3a}$$

$$\eta^{j,k}_{\mathrm{sm,ASN}} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k}) - \boldsymbol{C}(\boldsymbol{X}^{j,k})\right|\right|_{\mathrm{r}}, \tag{6.3b}$$

and the total estimator by $\eta^{j,k}_{\mathrm{ASN}} := \eta^{j,k}_{\mathrm{sm,ASN}} + \eta^{j,k}_{\mathrm{lin,ASN}}$.

First, we analyze the performance of the adaptive stopping criterion based on the estimators for stopping the linearization steps. We compare it with the classical approach in where the linearization is continued until the relative norm of the linearization estimator becomes smaller than a threshold taken as 10^{-8}, i.e.,

$$\text{Classical stopping criterion: } \eta^{j,k}_{\mathrm{lin,ASN}} < 10^{-8}, \tag{6.4}$$

$$\text{Adaptive stopping criterion: } \eta^{j,k}_{\mathrm{lin,ASN}} < \alpha_{\mathrm{lin}} \eta^{j,k}_{\mathrm{sm,ASN}}. \tag{6.5}$$

We set $\mu^1 = 1, \varepsilon = 10^{-8}, \alpha_{\mathrm{lin}} = 1$, and $\alpha = 0.1$ in Algorithm 1. Figure 3 depicts the evolution of the estimators and the relative norm of the total residual vector $\boldsymbol{R}(\boldsymbol{X}^{j,k})$ given in (2.6) as a function of the smoothing Newton–F–B iterations, at a specific smoothing iteration $j = 1$ ($\mu^1 = 1$), left, and $j = 3$ ($\mu^3 = 10^{-2}$), right. We can observe from Figure 3, left, that, as expected, the smoothing estimator and $\|\boldsymbol{R}(\boldsymbol{X}^{j,k})\|_{\mathrm{r}}$ stagnates after few steps, since here the smoothing parameter μ^1 is equal to 1, whereas the linearization estimator steadily decreases. If we consider the classical stopping criterion (6.4), the linearization will only be stopped at step $k = 8$. On the other hand, with our adaptive stopping criterion (6.5), only one iteration is necessary. Clearly after a few linearization steps, the linearization estimator no longer affects significantly the smoothing estimator, and we can economize many useless iterations.

Next, we provide in Table 1 the results obtained using the adaptive stopping criterion (6.5) to stop the nonlinear solver. We terminate the smoothing iterations using the relative norm of the total residual vector (2.6)

$$\|\boldsymbol{R}(\boldsymbol{X}^{j,\overline{k}})\|_{\mathrm{r}} < 10^{-8}. \tag{6.6}$$

We present the cumulated number of Newton iterations Niter, the estimators (6.3), and the relative norm of the total residual vector (2.6) at each smoothing step j. In terms of numbers, 10 smoothing iterations and 36 cumulated Newton iterations (CPU time: 6.9s) are needed to achieve the stopping criterion (6.6). From Table 1, one can see that for each value of μ^j, the Newton iterations are stopped according to (6.5). $\|\boldsymbol{R}(\boldsymbol{X}^{j,\overline{k}})\|_{\mathrm{r}}$ decreases until lying below 10^{-8}. Figure 4 displays the curve of the estimators as a function of cumulated Newton iterations and smoothing iterations, as well as the relative norm of the total residual vector as a function of smoothing iterations. The improvement of the performance with respect to the semismooth Newton-F–B method of Section 6.3 is spectacular.

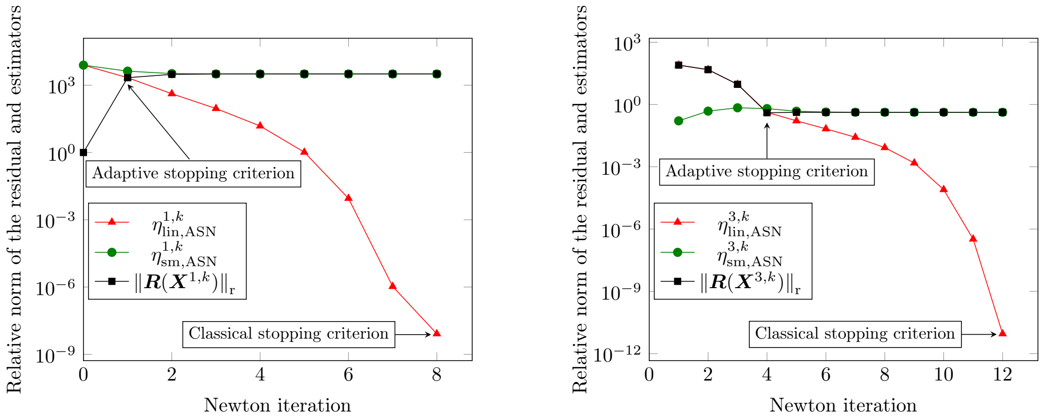

Figure 3: [Adaptive smoothing Newton method, smoothed F–B function (3.2), classical and adaptive stopping criteria (6.4) and (6.5)] Relative norm of the total residual vector (2.6) and estimators (6.3) as a function of Newton iterations k, at a specific smoothing iteration $j = 1$ ($\mu^1 = 1$), left, and at $j = 3$ ($\mu^3 = 10^{-2}$), right.

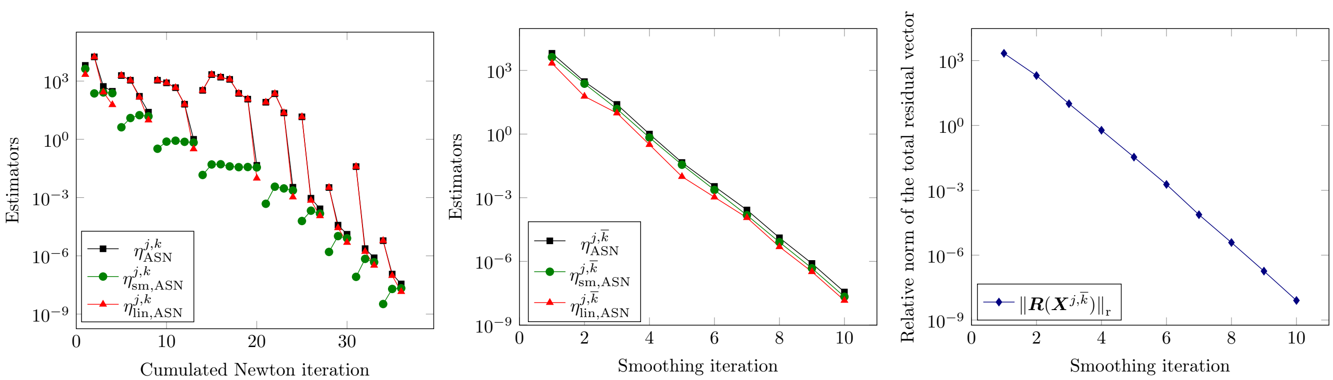

Figure 4: [Adaptive smoothing Newton method, smoothed F–B function (3.2), adaptive stopping criterion (6.5)] Estimators (6.3) as a function of cumulated Newton iterations (left). Estimators (6.3) (middle) and relative norm of the total residual vector (2.6) (right) as a function of smoothing iterations j at convergence of the linearization solver.

μ^j	Niter	$\eta^{j,\overline{k}}_{\mathrm{lin,ASN}}$	$\eta^{j,\overline{k}}_{\mathrm{sm,ASN}}$	$\|\boldsymbol{R}(\boldsymbol{X}^{j,\overline{k}})\|_{\mathrm{r}}$
1e+00	1	2.17e+03	4.24e+03	2.17e+03
1e-01	3	6.00e+01	2.37e+02	2.03e+02
1e-02	4	9.73e+00	1.53e+01	1.01e+01
1e-03	5	3.18e-01	6.84e-01	6.00e-01
1e-04	7	9.87e-03	3.58e-02	3.43e-02
1e-05	4	1.06e-03	2.33e-03	1.87e-03
1e-06	3	1.14e-04	1.50e-04	7.45e-05
1e-07	3	4.85e-06	8.04e-06	3.84e-06
1e-08	3	3.23e-07	4.72e-07	1.83e-07
1e-09	3	1.43e-08	2.15e-08	8.04e-09

Table 1: [Adaptive smoothing Newton method, smoothed F–B function (3.2), adaptive stopping criterion (6.5)] Number of Newton iterations Niter, estimators (6.3), and relative norm of the total residual vector (2.6) at each smoothing iteration j, at convergence of the linearization solver.

With the intention to compare the proposed method to existing methods, we complete the semismooth Newton method by a path-following strategy to solve problem (1.1), following [48]. For the sake of brevity, we shall not detail this here. The following test compares the semismooth Newton method (SSN) and the semismooth Newton method with path-following (SSN-pf) in which the linearization is stopped when the criterion (6.2) is satisfied, to the adaptive smoothing Newton method, using the smoothed min and F–B functions (3.1) and (3.2) and the stopping criteria (6.5) and (6.2) respectively for the linearization and smoothing iterations. We compare the number of cumulated linearization iterations and the global CPU time of the simulation for the different strategies. The results are displayed in Figure 5. They confirm the expected reduction of the computational cost of the numerical resolution with our adaptive approaches. Actually, we notice that the semismooth Newton method with path-following (red curve) and the adaptive smoothing Newton method (purple and dark blue curves) require significantly fewer cumulated Newton iterations and time to converge, in comparison with the semismooth Newton method (green and orange curves). Therefore, employing the path-following strategy or the adaptive strategy based on a posteriori error estimates enables to save many unnecessary additional iterations, and yield much better results than the pure semismooth Newton method. We note that, using the adaptive smoothing Newton method, one obtains similar computational results using both the smoothed F–B or the smoothed min function.

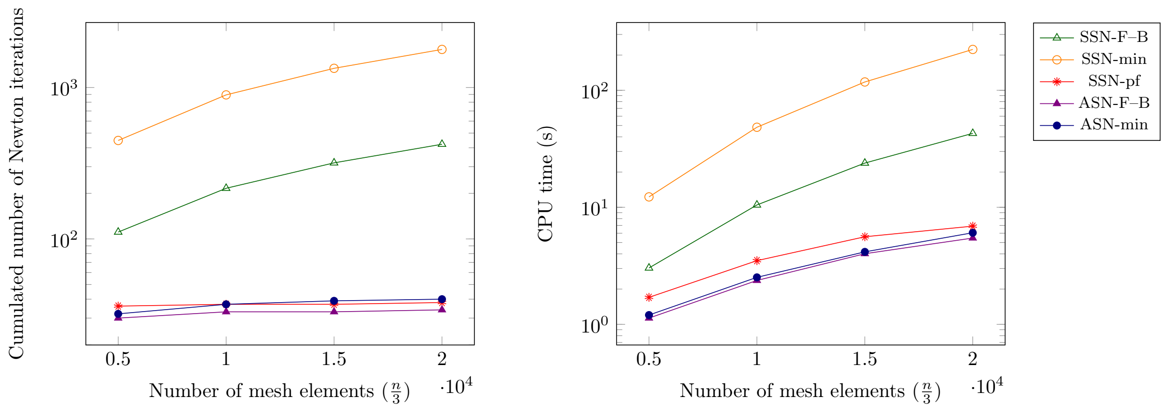

Figure 5: [Semismooth Newton method (with and without a path-following strategy) and adaptive smoothing method] Cumulated number of Newton iterations (left) and CPU time (right) as a function of the number of mesh elements.

6.5 Adaptive inexact smoothing Newton method

We focus in this section on the adaptive inexact Newton method introduced in Section 3 and investigate the performance of Algorithm 1 using the smoothed F–B function (3.2) together with the restarted GMRES method. Typically, we use a fixed restart parameter equal to 300. The behavior of the adaptive smoothing solvers can be improved dramatically by using good preconditioners. Here, we merely use an ILU preconditioner to speed-up the GMRES solver. For other possibilities for preconditioners, we refer to, e.g., [31] and the references therein. To point out the efficiency of the adaptivity, we test two approaches. First, we stop the algebraic iterations using the standard GMRES stopping criterion on the relative residual given by

$$\boldsymbol{R}_{\text{rel}} := \frac{\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,i}))\|}{\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k-1}))\|} \leq 10^{-10}, \tag{6.7}$$

where $\mathbb{M}_1$ and $\mathbb{M}_2$ are the preconditioner matrices. Second, we incorporate the adaptive stopping criteria (3.11a) for the algebraic solver in Algorithm 1. We set the parameters $\mu^1 = 1, \varepsilon = 10^{-5}, \alpha_{\text{alg}} = 10^{-3}, \alpha_{\text{lin}} = 1$, and $\alpha = 0.1$. Figure 6 depicts the evolution of the algebraic and linearization estimators and the GMRES relative residual during the algebraic resolution, for specific smoothing step j and linearization step k. For $j = 2$ and $k = 2$, we see that 22 GMRES iterations are needed to achieve the standard stopping criterion (6.7), whereas in the adaptive resolution case, only 10 GMRES iterations are required to satisfy the adaptive stopping criterion (3.11a). In this case, we can avoid many unnecessary iterations. One can also see from the right part of Figure 6, for $j = 3$ and $k = 1$, that the overall gain in terms of algebraic iterations obtained using our stopping criteria is quite significant.

Figure 7, left, shows the evolution of the estimators during smoothing iterations, at convergence of the nonlinear and linear solvers. As expected, the estimators decrease when μ decreases at each smoothing step. In the middle

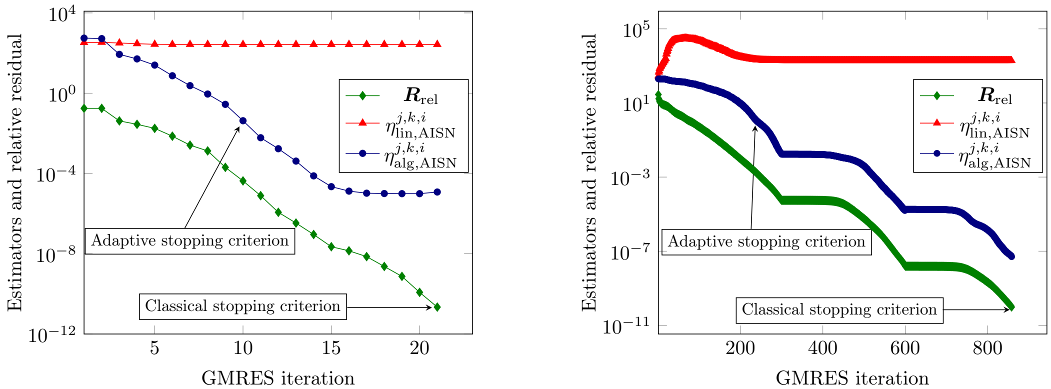

Figure 6: [Adaptive inexact smoothing Newton method, smoothed F–B function (3.2), Algorithm 1] Algebraic and linearization estimators (3.10) and GMRES relative residual as a function of the GMRES iterations i, for a fixed smoothing and linearization iterations, $j = 2, k = 2, i$ varies, left, and $j = 3, k = 1, i$ varies, right, using the classical stopping criterion (6.7) and the adaptive one (3.11a).

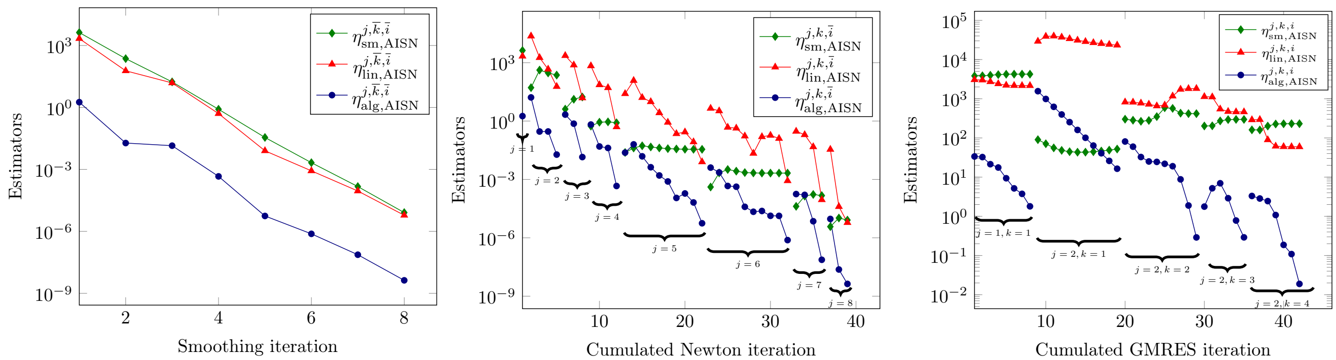

Figure 7: [Adaptive inexact smoothing Newton method, smoothed F–B function (3.2), Algorithm 1] Estimators (3.10) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers, left. Estimators as a function of cumulated Newton iterations at convergence of the algebraic solver, middle. Estimators as a function of cumulated GMRES iterations during the first two smoothing iterations ($j = 1$ and $j = 2$), right.

part of Figure 7, we can observe the behavior of the estimators at the end of the algebraic iterations, during the linearization iterations. We present 8 curves, each one corresponding to a specific value of μ^j. We can see that at each smoothing iteration j, the smoothing estimator $\eta_{\rm sm,AISN}^{j,k,\overline{i}}$ stagnates after about two iterations. The linearization estimator $\eta_{\rm lin,AISN}^{j,k,\overline{i}}$ decreases until becoming smaller than the smoothing estimator, satisfying the stopping criterion (3.11b). Finally, the detected behavior in terms of all smoothing iterations j, linearization iterations k, and algebraic solver iterations i is presented in Figure 7, right, for $j \leq 2$. The overall results are collected in Table 2. We present in particular the number of linearization and cumulated algebraic iterations per smoothing step j, Niter and Giter respectively, as well as the estimators (3.10) and the relative norm of the total residual vector (2.6) at the end of each smoothing step j. Using the adaptive stopping criteria (3.11), 8 smoothing iterations, 39 cumulated Newton iterations, and 5999 cumulated GMRES iterations are needed to ensure convergence. Figure 8 illustrates the performance of the adaptive inexact smoothing Newton method. It represents the ratio between: 1) the number of algebraic iterations (left) and the CPU time (right) using the classical GMRES stopping criterion (6.7) and 2) the number of algebraic iterations and the CPU time using the adaptive stopping criterion (3.11a) for GMRES, as a function of the number of elements. For larger systems, 20-times fewer iterations and 18-times faster execution time are achieved.

μ^j	Niter	Giter	$\eta^{j,\overline{k},\overline{i}}_{\text{lin,AISN}}$	$\eta^{j,\overline{k},\overline{i}}_{\text{sm,AISN}}$	$\eta^{j,\overline{k},\overline{i}}_{\text{alg,AISN}}$	$\left\|\left\|\boldsymbol{R}(\boldsymbol{X}^{j,\overline{k},\overline{i}})\right\|\right\|_{\text{r}}$
1e+00	1	8	2.16e+03	4.24e+03	1.80e+00	2.19e+03
1e-01	4	34	5.95e+01	2.31e+02	1.89e-02	1.80e+02
1e-02	3	391	1.54e+01	1.73e+01	1.41e-02	6.75e+00
1e-03	4	198	5.04e-01	8.16e-01	4.60e-04	5.95e-01
1e-04	10	796	7.99e-03	3.53e-02	5.58e-06	3.43e-02
1e-05	10	684	8.54e-04	2.12e-03	7.61e-07	1.94e-03
1e-06	4	513	9.03e-05	1.48e-04	7.42e-08	1.05e-04
1e-07	3	3375	6.04e-06	8.14e-06	4.27e-09	4.26e-06

Table 2: [Adaptive inexact smoothing Newton method, smoothed F–B function (3.2), Algorithm 1] Number of Newton iterations and cumulated GMRES iterations, estimators (3.10), and relative norm of the total residual vector (2.6) at each smoothing iteration j, at convergence of the algebraic and linearization solvers.

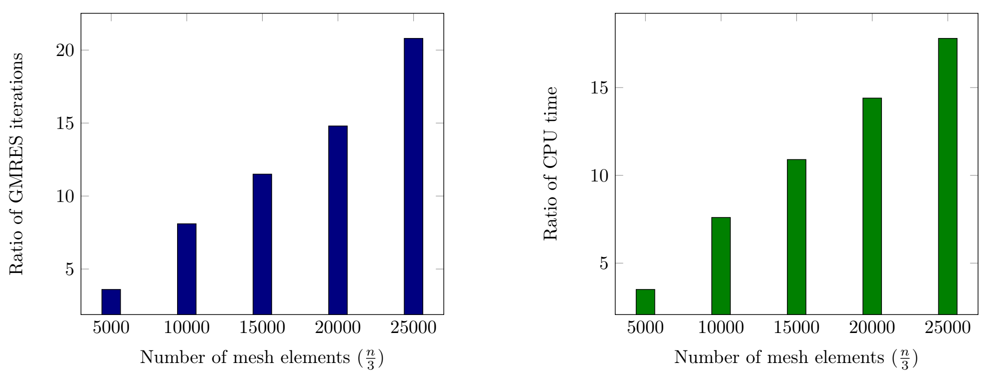

Figure 8: [Adaptive inexact smoothing Newton method, smoothed F–B function (3.2), Algorithm 1] Ratio between: the number of algebraic iterations (left) and CPU time (right) needed by the classical stopping criterion (6.7) to converge to the number and time needed by the adaptive stopping criterion (3.11a), as a function of the number of mesh elements.

6.6 Nonparametric interior-point method

We consider here the nonparametric interior-point approach of Section 4, where the dimension of the corresponding problem is $n = 3N + 1$. The value of the constant θ in the additional equation (4.2) is 10^{-1}. The stopping criterion is on the relative norm of the linearization residual vector

$$\|\boldsymbol{R}^{\text{IP}}(\boldsymbol{\mathcal{X}}^k)\|_{\text{r}} := \|\boldsymbol{R}^{\text{IP}}(\boldsymbol{\mathcal{X}}^k)\| / \|\boldsymbol{R}^{\text{IP}}(\boldsymbol{\mathcal{X}}^0)\| < 10^{-8}, \tag{6.8}$$

with

$$\boldsymbol{R}^{\text{IP}}(\boldsymbol{\mathcal{X}}^k) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^k \\ \boldsymbol{\mu} - \boldsymbol{K}(\boldsymbol{X}^k)\boldsymbol{G}(\boldsymbol{X}^k) \\ -\theta\mu^k - (\mu^k)^2 \end{bmatrix}.$$

Using this method, 19 Newton iterations (CPU time: 6.7s) are needed to reach the end of the simulation. Figure 9 shows that $\left\|\left\|\boldsymbol{R}^{\text{IP}}(\boldsymbol{\mathcal{X}}^k)\right\|\right\|_{\text{r}}$ decreases during the Newton interior-point iterations until satisfying the stopping criterion (6.8).

6.7 Adaptive interior-point method

Next, we consider the adaptive interior-point method, which is the method presented in Section 5, Algorithm 2 without applying an algebraic iterative solver to approximate the solution of the linear system (5.2). In this case, we can define the linearization and smoothing estimators respectively by

$$\eta^{j,k}_{\text{lin,AIP}} := \left\|\boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k})\right\|_{\text{r}}, \tag{6.9a}$$

$$\eta^{j,k}_{\text{sm,AIP}} := \left\|\boldsymbol{\mu}^j\right\|_{\text{r}}, \tag{6.9b}$$

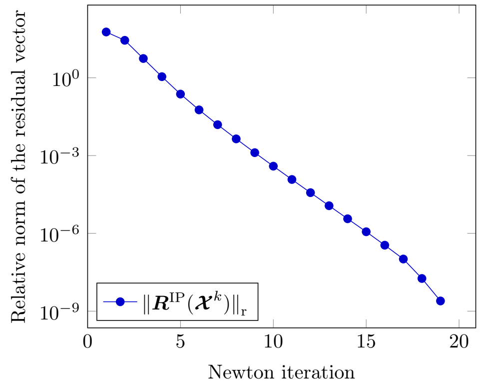

Figure 9: [Nonparametric interior-point method, stopping criterion (6.8)] Relative norm of the linearization residual vector (6.8) as a function of Newton iterations.

where $\boldsymbol{H}_{\mu^j}(\cdot)$ is defined in (5.1b), and the total estimator by $\eta_{\mathrm{AIP}}^{j,k} := \eta_{\mathrm{sm,AIP}}^{j,k} + \eta_{\mathrm{lin,AIP}}^{j,k}$. Recall from (5.8) the definition of the total residual vector for $\boldsymbol{V} \in \mathbb{R}^n$ as

$$\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}(\boldsymbol{V}) \end{bmatrix}, \tag{6.10}$$

where $\boldsymbol{H}(\cdot)$ is defined in (5.7). The adaptive stopping criterion

$$\eta_{\mathrm{lin,AIP}}^{j,k} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm,AIP}}^{j,k} \tag{6.11}$$

is used to stop the nonlinear solver and a criterion on the relative norm of the total residual vector is applied to stop the smoothing iterations

$$\|\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{X}^{j,\overline{k}})\|_{\mathrm{r}} < 10^{-8}. \tag{6.12}$$

The initial smoothing vector is $\boldsymbol{\mu}^1 = [1,\ldots,1]^T \in \mathbb{R}^N$ and $\alpha_{\mathrm{lin}} = 1$. Concerning the update of the smoothing parameter μ, we set $\alpha = 10^{-1}$. Table 3 summarizes the results. To achieve the stopping criterion (6.12), 11 smoothing iterations and 20 cumulated Newton iterations are needed (CPU time: 5.0s). In Figure 10, we plot the estimators (6.9) as a function of the cumulated Newton iterations (left), the smoothing iterations (middle), and the relative norm of the residual vector as a function of the smoothing iterations (right). The behavior of $\left\|\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{X}^{j,\overline{k}})\right\|_{\mathrm{r}}$ in Figure 10 appears a bit different from its behavior in Figure 4. This is related to the fact that the relative norm of the total residual given by (2.6) includes $\boldsymbol{C}(\boldsymbol{X})$ in the adaptive smoothing Newton method, whereas in this adaptive interior-point method, the relative norm of the total residual given by (6.10) includes $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X})$.

μ^j	Niter	$\eta_{\mathrm{lin,AIP}}^{j,\overline{k}}$	$\eta_{\mathrm{sm,AIP}}^{j,\overline{k}}$	$\left\|\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{X}^{j,\overline{k}})\right\|_{\mathrm{r}}$
1e+00	2	1.11e+01	2.00e+01	3.00e+01
1e-01	2	1.24e+00	2.00e+00	3.20e+00
1e-02	2	1.15e-01	2.00e-01	3.11e-01
1e-03	2	6.51e-03	2.00e-02	2.43e-02
1e-04	2	3.38e-04	2.00e-03	2.14e-03
1e-05	1	1.58e-04	2.00e-04	2.82e-04
1e-06	2	3.67e-06	2.00e-05	2.10e-05
1e-07	2	1.00e-07	2.00e-06	2.02e-06
1e-08	1	1.86e-07	2.00e-07	3.84e-07
1e-09	2	9.33e-10	2.00e-08	2.01e-08
1e-10	2	2.55e-11	2.00e-09	2.00e-09

Table 3: [Adaptive interior-point method] Number of Newton iterations, estimators (6.9), and relative norm of the total residual vector (6.10) at each smoothing step j, at convergence of the linearization solver.

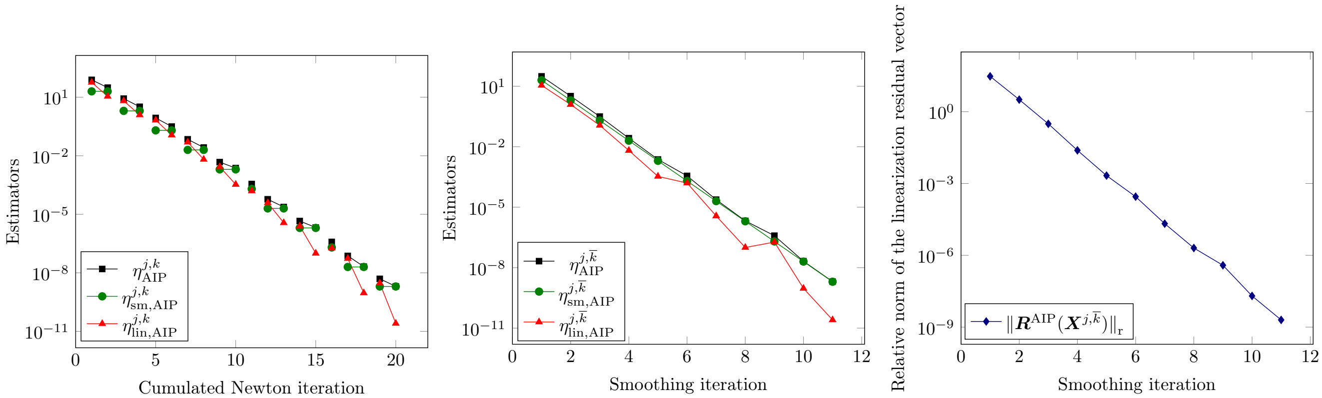

Figure 10: [Adaptive interior-point method] Estimators (6.9) as a function of cumulated Newton iterations (left). Estimators (6.9) (middle) and relative norm of the total residual vector (6.10) (right) as a function of smoothing iterations j at convergence of the linearization solver.

6.8 Adaptive inexact interior-point method

Let us now present the numerical results of the adaptive inexact interior-point method, detailed in Section 5. We employ Algorithm 2 with the GMRES algebraic solver and an ILU preconditioner. The parameters in Algorithm 2 are set as $\boldsymbol{\mu}^1 = [1, \ldots, 1]^T \in \mathbb{R}^N$, $\varepsilon = 10^{-5}$, $\alpha_{\mathrm{alg}} = 1$, $\alpha_{\mathrm{lin}} = 1$, and $\alpha = 0.1$. The restart parameter of restarted GMRES is chosen equal to 300. From Table 4, we can see that the method converged after 8 smoothing iterations, 20 cumulated linearization iterations, and 760 cumulated GMRES iterations. Figure 11, left, displays the curves of the estimators (5.9) as a function of the smoothing iteration. One can see that the estimators satisfy the adaptive stopping criteria incorporated in Algorithm 2. In Figure 11, right, the estimators are shown as a function of cumulated Newton iterations, at convergence of the linear solver.

μ^j	Niter	Giter	$\eta^{j,\overline{k},\overline{i}}_{\mathrm{lin,AIIP}}$	$\eta^{j,\overline{k},\overline{i}}_{\mathrm{sm,AIIP}}$	$\eta^{j,\overline{k},\overline{i}}_{\mathrm{alg,AIIP}}$	$\left\|\left\|\boldsymbol{R}^{\mathrm{AIIP}}(\boldsymbol{X}^{j,\overline{k},\overline{i}})\right\|\right\|_{\mathrm{r}}$
1e+00	3	7	1.15e+01	2.00e+01	3.36e+00	5.59e+00
1e-01	2	12	5.44e-01	2.00e+00	1.78e-02	2.00e+00
1e-02	3	20	9.75e-02	2.00e-01	2.80e-02	2.04e-01
1e-03	3	29	4.82e-03	2.00e-02	1.74e-03	2.01e-02
1e-04	3	56	2.52e-04	2.00e-03	2.19e-04	2.01e-03
1e-05	2	62	1.77e-04	2.00e-04	1.08e-04	2.29e-04
1e-06	2	110	1.49e-05	2.00e-05	1.42e-05	2.46e-05
1e-07	2	464	1.34e-06	2.00e-06	1.16e-06	2.31e-06

Table 4: [Adaptive inexact interior-point method, Algorithm 2] Number of cumulated Newton and GMRES iterations, estimators (5.9), and relative norm of the total residual vector (5.8) at each smoothing iteration j, at convergence of the algebraic and linearization solvers.

6.9 Comparison of the methods

This section is devoted to compare the semismooth Newton method (SSN), semismooth Newton method with path-following (SSN-pf), nonparametric interior-point method (IP), adaptive smoothing Newton method (ASN), and adaptive interior-point method (AIP). For this purpose, we introduce a unified residual vector, for $\boldsymbol{V} \in \mathbb{R}^n$

$$\boldsymbol{R}_{\mathrm{unif}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ \min(\boldsymbol{0}, \boldsymbol{K}(\boldsymbol{V})) \\ \min(\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{V})) \\ \boldsymbol{K}(\boldsymbol{V}) \cdot \boldsymbol{G}(\boldsymbol{V}) \end{bmatrix}, \tag{6.13}$$

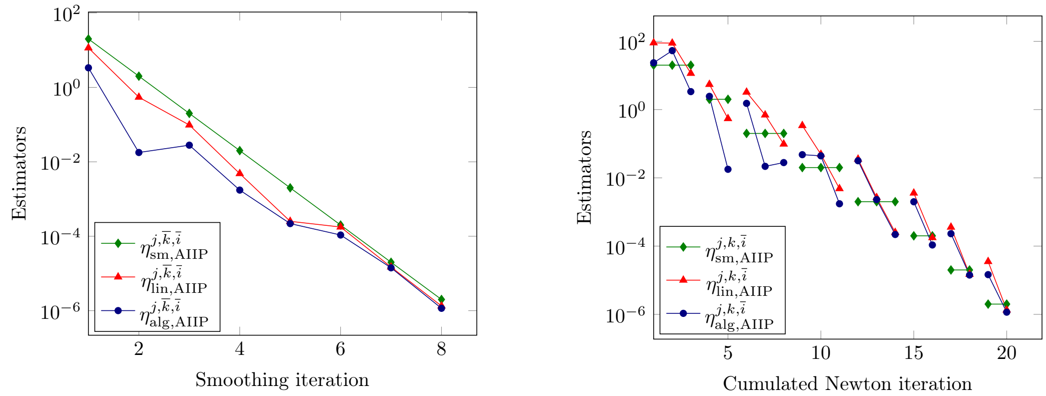

Figure 11: [Adaptive inexact interior-point method, Algorithm 2] Estimators (5.9) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers (left). Estimators as a function of cumulated Newton iterations k at convergence of the algebraic solver (right).

independent of the way the nonlinear complementarity constraints are reformulated. The stopping criterion of the nonlinear solver for the classical methods (SSN, SSN-pf, IP) is on the relative unified residual $\|\boldsymbol{R}_{\mathrm{unif}}(\boldsymbol{X}^k)\|_{\mathrm{r}}$ lying below 10^{-8}. Regarding the adaptive methods (ASN, AIP), to stop the nonlinear solver, we use the adaptive stopping criteria given respectively in (6.5) and (6.11). To stop the smoothing iterations, $\|\boldsymbol{R}_{\mathrm{unif}}(\boldsymbol{X}^{j,k})\|_{\mathrm{r}}$ is requested to become smaller than 10^{-8}.

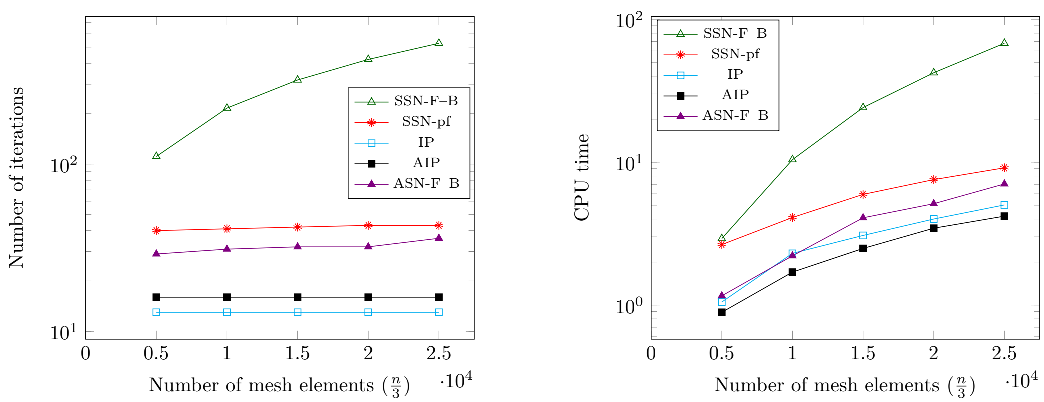

Figure 12: [Semismooth Newton method (F–B function (2.2)), semismooth Newton method with a path-following strategy, nonparametric interior-point method, adaptive interior-point method, and adaptive smoothing Newton method (smoothed F–B function (3.2))] Number of cumulated Newton iterations (left) and CPU time (right) as a function of the number of mesh elements, employing a stopping criterion on the relative norm of the unified residual vector (6.13).

In Figure 12, we plot the cumulated number of the Newton iterations (left) and the CPU time (right) required by each method, as a function of the number of mesh elements. It is clearly seen that the semismooth Newton method (green curve) is typically more costly, both in terms of the required number of iterations and the CPU time, in comparison with the other methods. Precisely, we can observe an important gain between the semismooth Newton method (green curve) and the adaptive smoothing Newton method (purple curve). Moreover, as we can remark from the red curve, the combination of a path-following strategy to the semismooth Newton method seems to be efficient. Finally, one does not see a remarkable difference between the results of the nonparametric interior-point method (cyan curve) and the adaptive interior-point method (black curve) in this test case.

7 Numerical experiments: Two-phase flow with phase transition

The second model problem that we consider in our numerical tests is a two-phase flow model (liquid–gas) with phase transition in porous media following [6, 13, 9]. Each of the liquid phase, denoted by l, and the gas phase, denoted by g, is composed of two components, water and hydrogen, denoted respectively by w and h.

7.1 Problem statement

The problem at hand can be formulated as a system of nonlinear partial differential equations with nonlinear complementarity constraints at each time step τ_ν. Let $\mathcal{T}_h$ be the spatial mesh, we denote respectively by S_K^ν, P_K^ν, and χ_K^ν the discrete elementwise unknowns approximating the values of the saturation S^{l}, the pressure P^{l}, and the molar fraction of hydrogen in the liquid phase $\chi_{\mathrm{h}}^{\mathrm{l}}$ in the element $K \in \mathcal{T}_h$ and on time step $1 \le \nu \le N_t$. Let N be the number of elements in the mesh $\mathcal{T}_h$. If one introduces the appropriate nonlinear function $H_{c,K}^\nu : \mathbb{R}^{3N} \to \mathbb{R},\ c \in \{\mathrm{w}, \mathrm{h}\}$, and suitable functions $F_K : \mathbb{R}^3 \to \mathbb{R}$ and $G_K : \mathbb{R}^3 \to \mathbb{R}$, the discrete problem written elementwise consists in finding $\boldsymbol{X}^\nu := (\boldsymbol{X}_K^\nu)_{K\in\mathcal{T}_h} \in \mathbb{R}^n$, where $n = 3N$, and $\boldsymbol{X}_K^\nu := [S_K^\nu, P_K^\nu, \chi_K^\nu] \in \mathbb{R}^3$, such that for all $K \in \mathcal{T}_h$

$$H_{c,K}^\nu(\boldsymbol{X}^\nu) = 0, \quad c \in \{\mathrm{w}, \mathrm{h}\}, \tag{7.1a}$$

$$F_K(\boldsymbol{X}_K^\nu) \ge 0, \quad G_K(\boldsymbol{X}_K^\nu) \ge 0, \quad F_K(\boldsymbol{X}_K^\nu)G_K(\boldsymbol{X}_K^\nu) = 0. \tag{7.1b}$$

The formulation (7.1) allows to model the transition from a single-phase flow to a two-phase flow during the appearance and disappearance of the gas phase and vice versa. As an example, a detailed finite volume discretization can be found in [7, Section 3.2]. The first $2N$ lines of system (7.1) can be written globally as

$$\mathcal{H}^\nu(\boldsymbol{X}^\nu) = 0,$$

where $\mathcal{H}^n : \mathbb{R}^{3N} \to \mathbb{R}^{2N}$ is defined elementwise by (7.1a).

Considering a C-function C^ν, for $1 \le \nu \le N_t$, we define a function $\mathcal{C}^\nu : \mathbb{R}^{3N} \to \mathbb{R}^N$ as $\mathcal{C}^\nu(\boldsymbol{X}^\nu) = C^\nu\left((F_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}, (G_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}\right)$. This leads us to apply a semismooth Newton method to find a solution for problem (7.1) written as

$$\begin{aligned} \mathcal{H}^\nu(\boldsymbol{X}^\nu) &= 0, \\ \mathcal{C}^\nu(\boldsymbol{X}^\nu) &= 0. \end{aligned} \tag{7.2}$$

The total residual vector $\boldsymbol{R}(\boldsymbol{V})$ of problem (7.2) is thus given by

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} -\mathcal{H}^\nu(\boldsymbol{V}) \\ -\mathcal{C}^\nu(\boldsymbol{V}) \end{bmatrix}, \quad \forall\, \boldsymbol{V} \in \mathbb{R}^n. \tag{7.3}$$

7.2 Adaptive smoothing Newton method

We introduce a function $\mathcal{C}_\mu^\nu : \mathbb{R}^{3N} \to \mathbb{R}^N$ defined as $\mathcal{C}_\mu^\nu(\boldsymbol{X}^\nu) = C_\mu^\nu\left((F_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}, (G_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}\right)$, for $1 \le \nu \le N_t$, where C_μ^ν is a smoothed C-function. Line (7.1b) can be approximated as a smoothed nonlinear equation $\mathcal{C}_\mu^\nu(\boldsymbol{X}^\nu) = 0$, making it possible to apply the standard Newton method to solve the resulting nonlinear system in the form: Find $\boldsymbol{X}^{\nu,j} \in \mathbb{R}^{3N}$ at each time step ν, $1 \le \nu \le N_t$, satisfying

$$\begin{aligned} \mathcal{H}^\nu(\boldsymbol{X}^{\nu,j}) &= 0, \\ \mathcal{C}_{\mu^{j_\nu}}^\nu(\boldsymbol{X}^{\nu,j}) &= 0. \end{aligned} \tag{7.4}$$

At each time step $1 \le \nu \le N_t$, each smoothing step $j \ge 1$, and each linearization step $k \ge 1$, fixing $\boldsymbol{X}^{\nu,j,0} \in \mathbb{R}^n$, we try to approach the solution of problem (7.4) by finding a solution $\boldsymbol{X}^{\nu,j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^{j_\nu}}^{\nu,j,k-1} \boldsymbol{X}^{\nu,j,k} = \boldsymbol{B}_{\mu^{j_\nu}}^{\nu,j,k-1}, \tag{7.5}$$

where the Jacobian matrix $\mathbb{A}_{\mu^{j_\nu}}^{\nu,j,k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_{\mu^{j_\nu}}^{\nu,j,k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_{\mu^{j_\nu}}^{\nu,j,k-1} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^\nu}(\boldsymbol{X}^{\nu,j,k-1}) \\ \mathbf{J}_{\mathcal{C}_{\mu^{j_\nu}}^\nu}(\boldsymbol{X}^{\nu,j,k-1}) \end{bmatrix}, \tag{7.6a}$$

$$\boldsymbol{B}_{\mu^{j_\nu}}^{\nu,j,k-1} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^\nu}(\boldsymbol{X}^{\nu,j,k-1})\boldsymbol{X}^{\nu,j,k-1} - \mathcal{H}^\nu(\boldsymbol{X}^{\nu,j,k-1}) \\ \mathbf{J}_{\mathcal{C}_{\mu^{j_\nu}}^\nu}(\boldsymbol{X}^{\nu,j,k-1})\boldsymbol{X}^{\nu,j,k-1} - \mathcal{C}_{\mu^{j_\nu}}^\nu(\boldsymbol{X}^{\nu,j,k-1}) \end{bmatrix}, \tag{7.6b}$$

with $\mathbf{J}_{\mathcal{H}^\nu}(\boldsymbol{X}^{\nu,j,k-1})$ and $\mathbf{J}_{\mathcal{C}_{\mu^{j_\nu}}^\nu}(\boldsymbol{X}^{\nu,j,k-1})$ the Jacobian matrices of the function $\mathcal{H}^\nu$ and the smoothed function $\mathcal{C}_{\mu^{j_\nu}}^\nu$, respectively, at the point $\boldsymbol{X}^{\nu,j,k-1}$ obtained by a Newton linearization.

7.3 Adaptive smoothing Newton algorithm

Let $\varepsilon > 0$ be the desired relative tolerance, $\alpha_{\text{lin}} \in]0,1]$ be the desired relative size of the linearization error, and $\alpha \in]0,1[$ the smoothing decrease parameter. The unsteady adaptive smoothing Newton algorithm reads as follows:

Algorithm 3 Unsteady adaptive smoothing Newton algorithm

Initialization: Fix $\varepsilon > 0,\ \alpha \in]0,1[$, and $\alpha_{\text{lin}} \in]0,1]$. Set $\nu := 1$ and $t_\nu := 0$. Choose $\boldsymbol{X}^{\nu,0} \in \mathbb{R}^n$.

Time loop

1. Fix $\mu^{j_\nu} > 0$ and set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu,0}$ and $k := 1$.

2.2 Newton linearization loop

2.2.1 From $\boldsymbol{X}^{\nu,j,k-1}$ define $\mathbb{A}^{\nu,j,k-1}_{\mu^{j_\nu}} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}^{\nu,j,k-1}_{\mu^{j_\nu}} \in \mathbb{R}^n$ given by (7.6).

2.2.2 Find $\boldsymbol{X}^{\nu,j,k}$ solution to the linear system

$$\mathbb{A}^{\nu,j,k-1}_{\mu^{j_\nu}} \boldsymbol{X}^{\nu,j,k} = \boldsymbol{B}^{\nu,j,k-1}_{\mu^{j_\nu}}.$$

2.2.3 Compute the estimators and check the stopping criterion for the nonlinear solver

$$\left(\eta^{\nu,j,k}_{\text{lin,ASN}} < \alpha_{\text{lin}} \eta^{\nu,j,k}_{\text{sm,ASN}}\right) \quad \text{or} \quad \left(\eta^{\nu,j,k}_{\text{lin,ASN}} < \varepsilon\right). \tag{7.7}$$

If satisfied, set $\overline{k} := k$ and stop. If not, set $k := k+1$ and go to **2.2.1**.

2.3 Check the stopping criterion for the smoothing iterations in the form:

$$\max\left\{\eta^{\nu,j,\overline{k}}_{\text{sm,ASN}}, \left|\left|\boldsymbol{R}(\boldsymbol{X}^{\nu,j,\overline{k}})\right|\right|_{\text{r}}\right\} < \varepsilon. \tag{7.8}$$

If satisfied, set $\overline{j} := j$ and stop. If not, set $j := j+1,\ \boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu,j-1,\overline{k}}$, and $\mu^{j_\nu} := \alpha\mu^{(j-1)_\nu}$. Then set $k := 1$ and go to **2.2.1**.

If $\nu = N_t$, stop. If not, set $\nu := \nu + 1,\ j = 1,\ \boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu-1,\overline{j}}$, and $t_\nu := \tau_\nu + t_{\nu-1}$. Then set $\mu^{j_\nu} = \mu^{\overline{j}_{\nu-1}},\ k = 1$, and go to **2.2.1**.

Description of Algorithm 3 For the first time step $\nu = 1$, starting with an initial approximation $\boldsymbol{X}^{\nu,0} \in \mathbb{R}^n$ and an initial smoothing parameter $\mu^{\nu,1} > 0$, we solve the smoothed nonlinear system (7.5) by the Newton linearization solver, and decrease the smoothing parameter $\mu^{\nu,j}$ at each smoothing step j, until the stopping criterion (7.8) on the smoothing estimator or the relative norm of the total residual vector is satisfied at step $\overline{j}$. Then, we continue the time loop, for $2 \le \nu \le N_t$, starting for $j = 1$ with $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu-1,\overline{j}}$ and $\mu^{j_\nu} := \mu^{\overline{j}_{\nu-1}}$, until satisfying the stopping criterion (7.8).

7.4 Numerical results

We consider a homogeneous porous medium in one dimension, supposed to be horizontal with length 2m, and a uniform spatial mesh with $N = 1000$ elements. The final time of simulation is $t_{\text{F}} =$ 100s, and the time step is constant $\tau_\nu = 10s$. We assume that the medium is initially saturated with liquid, $S^l = 1$, and containing no hydrogen, $\chi_h^l = 0$, on which we impose an injection of gas (hydrogen), constant in time, in the first cell of the mesh. The initial conditions are $S^{l,\nu=0} = 1, P^{l,\nu=0} = 10^6$Pa, and $\chi_h^{l,\nu=0} = 0$.

Semismooth Newton method. We begin by employing the semismooth Newton method presented in Section 2, with the min function (2.1) to solve the nonlinear system (7.2). On each time step $\nu \ge 1$, we request the relative norm of the total residual vector $\boldsymbol{R}(\boldsymbol{X}^{\nu,k})$ given by (7.3) to drop below 10^{-4}.

In Figure 13, the evolution of $\left|\left|\boldsymbol{R}(\boldsymbol{X}^{\nu,k})\right|\right|_{\mathrm{r}}$ is shown at each time step. 31 cumulated Newton iterations are needed.

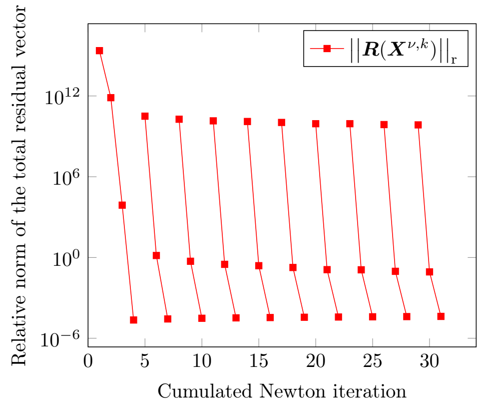

Figure 13: [Semismooth Newton method, min function (2.1)] Relative norm of the total residual vector (7.3) as a function of cumulated Newton iterations along the time steps ν.

Adaptive smoothing Newton method. Next, we present the results obtained using the adaptive smoothing Newton method, summarized in Algorithm 3, with the smoothed min function (3.1) to solve the smoothed nonlinear problem (7.4) at each time step τ_ν, $1 \leq \nu \leq N_t$. The parameters are set as $\mu^{j_1} = 10^{-1}, \varepsilon = 10^{-4}, \alpha_{\text{lin}} = 1$, and $\alpha = 0.1$.

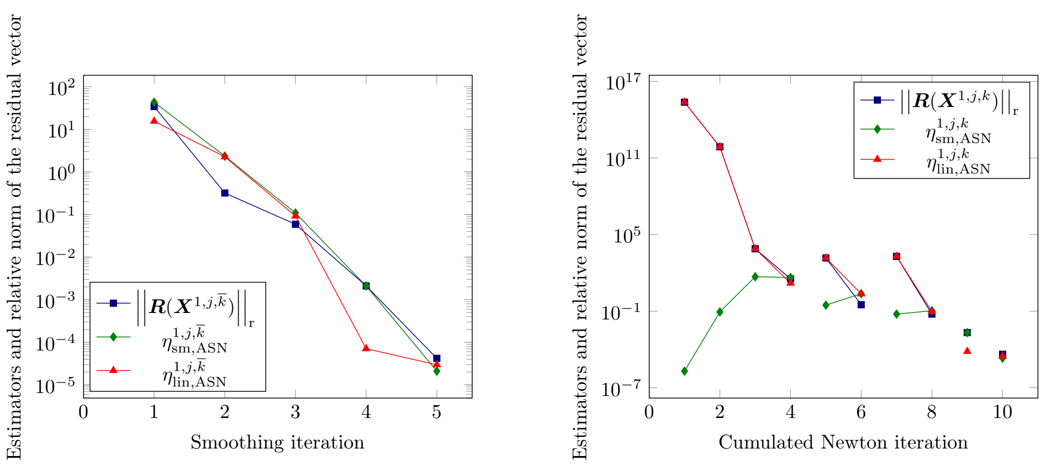

Figure 14: [Adaptive smoothing Newton method, smoothed min function (3.1), Algorithm 3] Estimators (6.3) and relative norm of the total residual vector (7.3) at the first time step $\nu = 1$ as a function of smoothing iterations j, at convergence of the linearization solver ($\nu = 1$ fixed, j varies, $k = \bar{k}$), left, and of cumulated Newton iterations, right, ($\nu = 1$ fixed, j and k vary).

ν	μ^{j_ν}	Niter	$\eta_{\mathrm{lin,ASN}}^{\nu,j,\overline{k}}$	$\eta_{\mathrm{sm,ASN}}^{\nu,j,\overline{k}}$	$\left\|\boldsymbol{R}(\boldsymbol{X}^{\nu,j,\overline{k}})\right\|_{\mathrm{r}}$
2	1e-05	3	2.15e-07	3.13e-07	2.86e-07
3	1e-05	3	3.13e-07	3.68e-07	4.24e-07
4	1e-05	3	3.93e-07	1.19e-07	3.47e-07
5	1e-05	3	4.62e-07	1.59e-07	4.01e-07
6	1e-05	3	5.04e-07	1.88e-06	1.87e-06
7	1e-05	3	5.58e-07	1.74e-07	4.94e-07
8	1e-05	3	5.94e-07	3.76e-07	7.08e-07
9	1e-05	3	6.64e-07	2.77e-07	7.50e-07
10	1e-05	3	7.01e-07	3.00e-07	7.89e-07

Table 5: [Adaptive smoothing Newton method, smoothed min function (3.1), Algorithm 3] Relative norm of the total residual vector (7.3) and estimators (6.3) at each time step ν, at convergence of the linearization solver.

From Figure 14, one can see that at the first time step $\nu = 1$ and at each smoothing step $j \leq 4$, the linearization estimator decreases until lying below the smoothing estimator. The smoothing iterations are thus stopped in the first possibility according to the stopping criterion (7.7). On the other hand, at the 5^{th} smoothing step, $\eta_{\mathrm{lin,ASN}}^{\nu,j,\overline{k}}$ is smaller than the fixed tolerance but not smaller than $\eta_{\mathrm{sm,ASN}}^{\nu,j,\overline{k}}$. Even after additional Newton iterations at this smoothing step, we will have the same observation. This justifies the modification applied in the adaptive stopping criterion (7.7). In Figure 14, right, we report the estimators and $\|\boldsymbol{R}(\boldsymbol{X}^{1,j,k})\|_{\mathrm{r}}$ as a function of cumulated Newton iteration for $\nu = 1$. The stopping criterion (7.8) is satisfied after 5 cumulated smoothing iterations, and 10 cumulated Newton iterations. Then, as presented in Table 5, starting at the second time step ($\nu = 2$) with $\mu^{j_\nu} = 10^{-5}$, the smoothing parameter does not decrease since the stopping criterion (7.8) is satisfied at each time step after one smoothing step only. To reach the end of the simulation, 9 cumulated smoothing steps and 31 cumulated linearization steps are needed.

As a conclusion, the results confirm the expected behavior of Algorithm 3 featuring an adaptive stopping criterion for the nonlinear solver. In this case, though, the stopping criteria in the adative smoothing Newton method do not bring the number of iterations down since the semismooth Newton method already behaves very well here.

8 Conclusion and outlook

In this work, we have considered nonlinear algebraic systems with inequalities in a form of complementarity constraints. We have considered some existing methods, like the semismooth Newton method, possibly combined with a path-following strategy, or a nonparametric interior-point method. Our goal was to propose a systematic way to drive such methods with adaptive stopping criteria and possibly inexact algebraic solvers. We have achieved this by a reformulation of the complementarity constraints using a smoothed function and a posteriori error estimate that enables to distinguish the different error components. Numerical experiments confirmed that the proposed adaptive approaches yield significant computational savings compared to some standard approaches from literature. Moreover, their numerical performance seems to be notably good across a range of test problems. In [8], we also take into account the discretization error of the considered problem, enabling to adaptively stop the outer smoothing loop in Algorithm 1, and employ the method to solve more involved problems.

References

[1] M. Aganagić, *Newton's method for linear complementarity problems*, Math. Programming, 28 (1984), pp. 349–362, https://doi.org/10.1007/BF02612339.

[2] M. Ainsworth and J. T. Oden, *A posteriori error estimation in finite element analysis*, Pure and Applied Mathematics (New York), Wiley-Interscience [John Wiley & Sons], New York, 2000, https://doi.org/10.1002/9781118032824.

[3] S. Bellavia, *Inexact interior-point method*, J. Optim. Th. Appl., 96 (1998), pp. 109–121, https://doi.org/10.1023/A:1022663100715.

[4] S. Bellavia, M. Macconi, and B. Morini, *An affine scaling trust-region approach to bound-constrained nonlinear systems*, Appl. Numer. Math., 44 (2003), pp. 257–280, https://doi.org/10.1016/S0168-9274(02)00170-8.

[5] F. Ben Belgacem, C. Bernardi, A. Blouza, and M. Vohralík, *On the unilateral contact between membranes. Part 2: a posteriori analysis and numerical experiments*, IMA J. Numer. Anal., 32 (2012), pp. 1147–1172, https://doi.org/10.1093/imanum/drr003.

[6] I. Ben Gharbia, *Résolution de problèmes de complémentarité. Application à un écoulement diphasique dans un milieu poreux*, PhD thesis, Université Paris Dauphine (Paris IX), 2012, http://tel.archives-ouvertes.fr/tel-00776617.

[7] I. Ben Gharbia, J. Dabaghi, V. Martin, and M. Vohralík, *A posteriori error estimates for a compositional two-phase flow with nonlinear complementarity constraints*, Comput. Geosci., 24 (2020), pp. 1031–1055, https://doi.org/10.1007/s10596-019-09909-5.

[8] I. Ben Gharbia, J. Ferzly, M. Vohralík, and S. Yousef, *Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality.* HAL Preprint 03696024, submitted for publication, 2022, https://hal.inria.fr/hal-03696024.

[9] I. Ben Gharbia and E. Flauraud, *Study of compositional multiphase flow formulation using complementarity conditions*, Oil Gas Sci. Technol., 74 (2019), p. 43, https://doi.org/10.2516/ogst/2019012.

[10] I. Ben Gharbia and J. C. Gilbert, *Nonconvergence of the plain Newton-min algorithm for linear complementarity problems with a P-matrix*, Math. Prog., 134 (2012), pp. 349–364, https://doi.org/10.1007/s10107-010-0439-6.

[11] I. Ben Gharbia and J. C. Gilbert, *An algorithmic characterization of P-matricity*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 904–916, https://doi.org/10.1137/120883025.

[12] I. Ben Gharbia and J. C. Gilbert, *An algorithmic characterization of P-matricity II: adjustments, refinements, and validation*, SIAM J. Matrix Anal. Appl., 40 (2019), pp. 800–813, https://doi.org/10.1137/18M1168522.

[13] I. Ben Gharbia and J. Jaffré, *Gas phase appearance and disappearance as a problem with complementarity constraints*, Math. Comput. Simul., 99 (2014), pp. 28–36, https://doi.org/10.1016/j.matcom.2013.04.021.

[14] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal, *Numerical optimization*, Universitext, Springer-Verlag, Berlin, second ed., 2006, https://doi.org/10.1007/978-3-540-35447-5.

[15] M. Bürg and A. Schröder, *A posteriori error control of hp-finite elements for variational inequalities of the first and second kind*, Comput. Math. Appl., 70 (2015), pp. 2783–2802, https://doi.org/10.1016/j.camwa.2015.08.031.

[16] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Society for Industrial and Applied Mathematics, 1990, https://doi.org/10.1137/1.9781611971309.

[17] J. Dabaghi, V. Martin, and M. Vohralík, *Adaptive inexact semismooth Newton methods for the contact problem between two membranes*, J. Sci. Comput., 84 (2020), p. Paper No. 28, https://doi.org/10.1007/s10915-020-01264-3.

[18] D. A. Di Pietro, M. Vohralík, and S. Yousef, *An a posteriori-based, fully adaptive algorithm with adaptive stopping criteria and mesh refinement for thermal multiphase compositional flows in porous media*, Comput. Math. Appl., 68 (2014), pp. 2331–2347, https://doi.org/10.1016/j.camwa.2014.08.008.

[19] J.-P. Dussault, M. Frappier, and J. C. Gilbert, *A lower bound on the iterative complexity of the Harker and Pang globalization technique of the Newton-min algorithm for solving the linear complementarity problem*, EURO J. Comput. Optim., 7 (2019), pp. 359–380, https://doi.org/10.1007/s13675-019-00116-6.

[20] J.-P. Dussault, M. Frappier, and J. C. Gilbert, *Polyhedral Newton-min algorithms for complementarity problems*, research report, Inria Paris, France, Université de Sherbrooke, Canada, HAL Preprint 02306526, submitted for publication, 2019, https://hal.archives-ouvertes.fr/hal-02306526.

[21] A. Ern and M. Vohralík, *Adaptive inexact Newton methods with a posteriori stopping criteria for nonlinear diffusion PDEs*, SIAM J. Sci. Comput., 35 (2013), pp. A1761–A1791, https://doi.org/10.1137/120896918.

[22] F. Facchinei and C. Kanzow, *A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems*, Math. Program., 76 (1997), pp. 493–512, https://doi.org/10.1016/S0025-5610(96)00058-5.

[23] F. Facchinei and J.-S. Pang, *Finite-dimensional variational inequalities and complementarity problems. Vol. I*, Springer Series in Operations Research, Springer-Verlag, New York, 2003, https://doi.org/10.1007/b97543.

[24] F. Facchinei and J.-S. Pang, *Finite-dimensional variational inequalities and complementarity problems. Vol. II*, Springer Series in Operations Research, Springer-Verlag, New York, 2003, https://doi.org/10.1007/b97544.

[25] A. Galántai, *Properties and construction of NCP functions*, Comput. Optim. Appl., 52 (2012), pp. 805–824, https://doi.org/10.1007/s10589-011-9428-9.

[26] Z. Ge, Q. Ni, and X. Zhang, *A smoothing inexact Newton method for variational inequalities with nonlinear constraints*, J. Inequal. Appl., (2017), pp. Paper No. 160, 12. , https://doi.org/10.1186/s13660-017-1433-9.

[27] M. Hintermüller, K. Ito, and K. Kunisch, *The primal-dual active set strategy as a semismooth Newton method*, SIAM J. Optim., 13 (2002), pp. 865–888 (2003), https://doi.org/10.1137/S1052623401383558.

[28] M. Hintermüller and K. Kunisch, *Path-following methods for a class of constrained minimization problems in function space*, SIAM J. Optim., 17 (2006), pp. 159–187, https://doi.org/10.1137/040611598.

[29] K. Ito and K. Kunisch, *Augmented lagrangian methods for nonsmooth, convex optimization in hilbert spaces*, Nonlinear Analysis: Theory, Methods & Applications, 41 (2000), pp. 591–616, https://doi.org/10.1016/S0362-546X(98)00299-5.

[30] K. Ito and K. Kunisch, *Lagrange multiplier approach to variational problems and applications*, vol. 15 of Advances in Design and Control, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008, https://doi.org/10.1137/1.9780898718614.

[31] S. Lacroix, Y. Vassilevski, J. Wheeler, and M. Wheeler, *Iterative solution methods for modeling multiphase flow in porous media fully implicitly*, SIAM J. Sci. Comput., 25 (2003), pp. 905–926, https://doi.org/10.1137/S106482750240443X.

[32] J. M. Martínez and L. Q. Qi, *Inexact Newton methods for solving nonsmooth equations*, J. Comput. Appl. Math., 60 (1995), pp. 127–145, https://doi.org/10.1016/0377-0427(94)00088-I. Linear/nonlinear iterative methods and verification of solution (Matsuyama, 1993).

[33] T. S. Munson, F. Facchinei, M. C. Ferris, A. Fischer, and C. Kanzow, *The semismooth algorithm for large scale complementarity problems*, INFORMS J. Comput., 13 (2001), pp. 294–311, https://doi.org/10.1287/ijoc.13.4.294.9734.

[34] R. H. Nochetto, K. G. Siebert, and A. Veeser, *Theory of adaptive finite element methods: an introduction*, in Multiscale, nonlinear and adaptive approximation, Springer, Berlin, 2009, pp. 409–542, https://doi.org/10.1007/978-3-642-03413-8_12.

[35] H.-D. Qi and L.-Z. Liao, *A smoothing Newton method for general nonlinear complementarity problems*, Comput. Optim. Appl., 17 (2000), pp. 231–253, https://doi.org/10.1023/A:1026554432668.

[36] L. Qi and D. Sun, *Smoothing functions and smoothing Newton method for complementarity and variational inequality problems*, J. Optim. Th. Appl., 113 (2002), pp. 121–147, https://doi.org/10.1023/A:1014861331301.

[37] S. Repin, *A posteriori estimates for partial differential equations*, vol. 4 of Radon Series on Computational and Applied Mathematics, Walter de Gruyter GmbH & Co. KG, Berlin, 2008, https://doi.org/10.1515/9783110203042.

[38] S. I. Repin, *Functional a posteriori estimates for elliptic variational inequalities*, J. Math. Sci., 152 (2008), pp. 702–712, https://doi.org/10.1007/s10958-008-9093-4.

[39] S.-P. Rui and C.-X. Xu, *A smoothing inexact Newton method for nonlinear complementarity problems*, J. Comput. Appl. Math., 233 (2010), pp. 2332–2338, https://doi.org/10.1016/j.cam.2009.10.018.

[40] G. Stadler, *Semismooth Newton and augmented Lagrangian methods for a simplified friction problem*, SIAM J. Optim., 15 (2004), pp. 39–62, https://doi.org/10.1137/S1052623403420833.

[41] G. Stadler, *Path-following and augmented Lagrangian methods for contact problems in linear elasticity*, J. Comput. Appl. Math., 203 (2007), pp. 533–547, https://doi.org/10.1016/j.cam.2006.04.017.

[42] D. Sun and L. Qi, *On NCP-functions*, Comput. Optim. Appl., 13 (1999), pp. 201–220, https://doi.org/10.1023/A:1008669226453.

[43] M. Ulbrich, *Semismooth Newton methods for variational inequalities and constrained optimization problems in function spaces*, vol. 11 of MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA, 2011, https://doi.org/10.1137/1.9781611970692.

[44] D. T. S. Vu, *Numerical resolution of algebraic systems with complementarity conditions. Application to the thermodynamics of compositional multiphase mixtures*, PhD thesis, Université Paris-Saclay, 2020, https://tel.archives-ouvertes.fr/tel-02987892.

[45] D. T. S. Vu, I. Ben Gharbia, M. Haddou, and Q. H. Tran, *A new approach for solving nonlinear algebraic systems with complementarity conditions. Application to compositional multiphase equilibrium problems*, Mathematics and Computers in Simulation, 190 (2021), pp. 1243–1274, https://doi.org/10.1016/j.matcom.2021.07.015.

[46] M. H. Wright, *The interior-point revolution in optimization: history, recent developments, and lasting consequences*, Bull. Amer. Math. Soc., 42 (2005), pp. 39–56, https://doi.org/10.1090/S0273-0979-04-01040-7.

[47] X. Xiao, Y. Li, Z. Wen, and L. Zhang, *A regularized semi-smooth Newton method with projection steps for composite convex programs*, J. Sci. Comput., (2018), pp. 364–389, https://doi.org/10.1007/s10915-017-0624-3.

[48] S. Zhang, Y. Yan, and R. Ran, *Path-following and semismooth Newton methods for the variational inequality arising from two membranes problem*, J. Inequal. Appl., (2019), pp. Paper No. 1, 13. , https://doi.org/10.1186/s13660-019-1955-4.

[49] X.-Y. Zhao, D. Sun, and K.-C. Toh, *A Newton-CG augmented Lagrangian method for semidefinite programming*, SIAM J. Optim., 20 (2010), pp. 1737–1765, https://doi.org/10.1137/080718206.