Revista Integración
Escuela de Matemáticas
Universidad Industrial de Santander
Vol. 39, N° 2, 2021, pág. 191–215

A global Jacobian smoothing algorithm for nonlinear complementarity problems

Wilmer Sánchez[a], Rosana Pérez[b] ✉, Héctor J. Martínez[c]

[a,b] Universidad del Cauca, Departamento de Matemáticas, Popayán, Colombia.

[c] Universidad del Valle, Departamento de Matemáticas, Cali, Colombia.

Abstract. In this paper, we use the smoothing Jacobian strategy to propose a new algorithm for solving complementarity problems based on its reformulation as a nonsmooth system of equations. This algorithm can be seen as a generalization of the one proposed in [18]. We develop its global convergence theory and under certain assumptions, we demonstrate that the proposed algorithm converges locally and, q-superlinearly or q-quadratically to a solution of the problem. Some numerical experiments show a good performance of this algorithm.

Keywords: Nonlinear complementarity problems, complementarity function, generalized Newton methods, Jacobian smoothing method, global convergence, superlinear convergence, quadratic convergence.

MSC2010: 49M15, 90C06, 90C30.

Un algoritmo global con jacobiano suavizado para problemas de complementariedad no lineal

Resumen. En este artículo, usamos la estrategia del jacobiano suavizado para proponer un nuevo algoritmo para resolver problemas de complementariedad no lineal basado en su reformulación como un sistema de ecuaciones no lineales. Este algoritmo puede verse como una generalización del propuesto en [18]. Desarrollamos su teoría de convergencia global y bajo ciertas hipótesis, demostramos que el algoritmo converge local y q superlineal o q cuadráticamente a la solución del problema. Pruebas numéricas muestran un buen desempeño del algoritmo propuesto.

Palabras clave: Complementariedad no lineal, función de complementariedad, método de Newton generalizado, Jacobiano suavizado, convergencia global, convergencia superlineal, convergencia cuadrática.

E-mail: wilmersanchez@unicauca.edu.co [a], rosana@unicauca.edu.co [b]✉, hector.martinez@correounivalle.edu.co [c].
Received: 11 october 2020, Accepted: 14 june 2021.
To cite this article: W. Sánchez, R. Pérez & H. J. Martínez, A global Jacobian smoothing algorithm for nonlinear complementarity problems, *Rev. Integr. Temas Mat.,* 39 (2021), No. 2, 191-215. doi: 10.18273/revint.v39n2-20210004

1. Introduction

The Nonlinear Complementarity Problem, (NCP), which in some contexts is synonymous with system in equilibrium, arises among others, in applications to Physics, Engineering, and Economics [3], [8], [14], [19]. The problem is to find a vector $\boldsymbol{x} \in \mathbb{R}^n$ such that $\boldsymbol{x} \geq 0$, $F(\boldsymbol{x}) \geq 0$ and $\boldsymbol{x}^T F(\boldsymbol{x}) = 0$, with $F\colon \mathbb{R}^n \to \mathbb{R}^n$ continuously differentiable. Here, a vector is nonnegative if all its components are nonnegative. A widely used technique for solving the NCP is to reformulate it as a system of nonlinear equations using a complementarity function $\varphi\colon \mathbb{R}^2 \to \mathbb{R}$ such that

$$\varphi(a,b) = 0 \Longleftrightarrow a \geq 0, \quad b \geq 0, \quad ab = 0. \tag{1}$$

Then, we consider $\Phi\colon \mathbb{R}^n \to \mathbb{R}^n$ and define the nonlinear system of equations

$$\Phi\left(\boldsymbol{x}\right) = \begin{pmatrix} \varphi(x_1, F_1(\boldsymbol{x})) \\ \vdots \\ \varphi(x_n, F_n(\boldsymbol{x})) \end{pmatrix} = \mathbf{0}, \tag{2}$$

which is a nondifferentiable system due to the lack of smoothness of φ. From (1), a vector $\boldsymbol{x}_*$ is a solution of (2), if, and only if, $\boldsymbol{x}_*$ it is a solution of the NCP. To solve (2) and thus, to solve the NCP, a nonsmooth algorithms type *Newton* [27], [29] and quasi-*Newton* [20], [21], among others [1], [6], [22], [26], [31] have been proposed. The natural merit function [24] $\Psi\colon \mathbb{R}^n \to \mathbb{R}$, defined by $\Psi(\boldsymbol{x}) = \frac{1}{2}\left\|\Phi(\boldsymbol{x})\right\|_2^2$, is used in the globalization of these methods. Thus, $\Psi(\boldsymbol{x})$ is minimized in $\mathbb{R}^n$. These methods use the concept of *generalized Jacobian* [10] defined by the set

$$\partial G(\boldsymbol{x}) = conv\left\{\lim_{k\to\infty} G'(\boldsymbol{x}_k) \in \mathbb{R}^{n\times n}\colon \lim_{k\to\infty} \boldsymbol{x}_k \to \boldsymbol{x},\, \boldsymbol{x}_k \in D_G\right\}, \tag{3}$$

for a *Lipschitz* continuous function $G\colon \mathbb{R}^n \to \mathbb{R}^n$, in $\boldsymbol{x}$, where D_G denotes the set of all points where is G is differentiable and $hull\,\{A\}$ is the convex envelope of A. This set is nonempty, convex, and compact [11]. Usually, the set (3) is difficult to compute, for this reason, we use the overestimation $\partial G(\boldsymbol{x})^T \subseteq \partial G_1(\boldsymbol{x}) \times \cdots \times \partial G_n(\boldsymbol{x})$ given in [11], where the right side, for short $\partial_C G(\boldsymbol{x})^T$ [18], called *C-sub differential* of G at $\boldsymbol{x}$, is the set of matrices $\mathbb{R}^{n\times n}$, whose i-th column is the generalized gradient of the i-th component of the function G.

Another strategy to solve (2) is to smooth the Jacobian proposed in [9] and called *Jacobian smoothing* in [18]. The general idea of methods using this strategy is to approximate Φ by a smooth function $\Phi_\mu\colon \mathbb{R}^n \to \mathbb{R}^n$, where $\mu > 0$ is the smoothing parameter, and then solving a sequence of smooth nonlinear equation systems,

$$\Phi_\mu(\boldsymbol{x}) = \begin{pmatrix} \varphi_\mu(x_1, F_1(\boldsymbol{x})) \\ \vdots \\ \varphi_\mu(x_n, F_n(\boldsymbol{x})) \end{pmatrix} = \mathbf{0}, \tag{4}$$

with μ going to zero and φ_μ a smoothing function of φ used in (2). The system (4) is solved at each iteration by solving the mixed Newton equation) $\Phi_\mu^{'}(\boldsymbol{x}_k)\boldsymbol{s}_k = -\Phi(\boldsymbol{x}_k)$.

In [18], the authors present a new algorithm with good numerical performance based on the Jacobian smoothing strategy to solve the NCP, by reformulating it as a system of nonlinear equations using the *Fischer-Burmeister* complementary function defined by $\varphi(a,b) = \sqrt{a^2+b^2} - a - b$. Motivated by the results obtained with this function, and since it is a particular case of the following family of complementary functions [17]

$$\varphi_\lambda(a,b) = \sqrt{(a-b)^2 + \lambda ab} - a - b, \qquad \lambda \in (0,4)\,, \tag{5}$$

corresponding to $\lambda = 2$, in this paper we use this strategy with the family (5) to propose a new algorithm that solves complementarity problems by reformulating it as a nonlinear, nondifferentiable system of equations. This algorithm can be seen as a generalization of the one proposed in [18] to any member of family φ_λ, with λ in $(0, 4)$. Under certain hypotheses, we demonstrate that the proposed algorithm converges local and, q-superlinear or q-quadratically to a solution of the complementarity problem. In addition, we analyze the numerical performance of the proposed algorithm.

The organization of this paper is as follows. In Section 2, we present the Jacobian smoothing strategy applied to a function Φ_λ, we described the Jacobian matrix of its smoothing and find an upper bound of the parameter μ which will be very important in the algorithmic proposal. In Section 3, we present some preliminary results that we use to develop the convergence theory of the algorithm proposed. In Section 4, we present a new Jacobian smoothing algorithm to solve nonlinear complementarity problems that generalize the one presented in [18] to all members of family (5). Moreover, we prove that our algorithm is well defined. In Section 5, we develop its global convergence theory. In Section 6, under some hypotheses, we prove that the algorithm converges local and q-superlinear or q-quadratically to the solution of the complementarity problem. In Section 8, we analyze the numerical performance of the proposed algorithm. Finally, In Section 9, we present our concluding remarks.

2. Smoothing Jacobian strategy for $\Phi_\lambda(\boldsymbol{x}) = \boldsymbol{0}$

We consider the reformulation (2) of the NCP as a nonsmooth nonlinear system of equations. If $\varphi = \varphi_\lambda$, the family (5), we obtain the system $\Phi_\lambda(\boldsymbol{x}) = \mathbf{0}$. The basic iteration of a generalized Newton method to solve this system has the form,

$$H_k \boldsymbol{s}_k = -\Phi_\lambda(\boldsymbol{x}^k), \tag{6}$$

where $H_k \in \partial\Phi_\lambda(\boldsymbol{x}^k)$ or $H_k \in \partial_C\Phi_\lambda(\boldsymbol{x}^k)$. Here, we use $H_k \in \partial_C\Phi_\lambda(\boldsymbol{x}^k)$. In order to define a smoothing Jacobian method for $\Phi_\lambda(\boldsymbol{x}) = \mathbf{0}$, we follow the basic idea given in [18] and we consider smoothing φ_λ as proposed in [4]: for all $\lambda \in (0,4)$ and $\mu > 0$,

$$\varphi_{\lambda\mu}(a,b) = \sqrt{(a-b)^2 + \lambda ab + (4-\lambda)\mu} - a - b = G_{\lambda\mu}(a,b) - a - b. \tag{7}$$

As expected, the distance between φ_λ and its smoothing function, $\varphi_{\lambda\mu}$, is upper bounded by a constant that depends on the parameters λ and μ. This is a particular case of the following proposition that will be useful in the convergence theory.

Proposition 2.1. *Function $\varphi_{\lambda\mu}$ satisfies the inequality*

$$|\varphi_{\lambda\mu_1}(a,b) - \varphi_{\lambda\mu_2}(a,b)| \le \sqrt{4-\lambda}\,|\sqrt{\mu_1} - \sqrt{\mu_2}|,$$

for all $(a,b) \in \mathbb{R}^2$ *and* $\mu_1, \mu_2 \geq 0$. *In particular,* $|\varphi_{\lambda\mu} - \varphi_\lambda| \leq \sqrt{4-\lambda}\,\sqrt{\mu}$, *for all* $(a,b) \in \mathbb{R}^2$, $\lambda \in (0,4)$ *and* $\mu \geq 0$.

Proof. Let $(a,b) \in \mathbb{R}^2$, μ_1 and μ_2 nonnegatives, such that $\mu_1 \neq \mu_2$.

$$|\varphi_{\lambda\mu_1}(a,b) - \varphi_{\lambda\mu_2}(a,b)| = |G_{\lambda\mu_1}(a,b) - G_{\lambda\mu_2}(a,b)| = \left| \frac{(\mu_1 - \mu_2)(4-\lambda)}{G_{\lambda\mu_1}(a,b) + G_{\lambda\mu_2}(a,b)} \right|.$$

Observe that $G_{\lambda\mu}(a,b) \geq \sqrt{\mu(4-\lambda)}$. Then

$$|\varphi_{\lambda\mu_1}(a,b) - \varphi_{\lambda\mu_2}(a,b)| \leq \frac{|(\mu_1 - \mu_2)(4-\lambda)|}{(\sqrt{\mu_1} + \sqrt{\mu_2})\sqrt{4-\lambda}} \leq |\sqrt{\mu_1} - \sqrt{\mu_2}|\sqrt{4-\lambda}\,.$$

In particular, if $\mu_1 = \mu$ and $\mu_2 = 0$ then $|\varphi_{\lambda\mu} - \varphi_\lambda| \leq \sqrt{4-\lambda}\,\sqrt{\mu}$. ☑

From (7), we define the function $\Phi_{\lambda\mu} \colon \mathbb{R}^n \to \mathbb{R}^n$ by,

$$\Phi_{\lambda\mu}(\boldsymbol{x}) = \begin{pmatrix} \varphi_{\lambda\mu}(x_1, F_1(\boldsymbol{x})) \\ \vdots \\ \varphi_{\lambda\mu}(x_n, F_n(\boldsymbol{x})) \end{pmatrix}. \tag{8}$$

The next proposition gives an upper bound for the distance between Φ_λ and its approximation $\Phi_{\lambda\mu}$.

Proposition 2.2. *The function* $\Phi_{\lambda\mu}$ *satisfies the following inequalities*

i) $\|\Phi_{\lambda\mu_1}(\boldsymbol{x}) - \Phi_{\lambda\mu_2}(\boldsymbol{x})\| \leq \kappa|\sqrt{\mu_1} - \sqrt{\mu_2}|$.

ii) $\|\Phi_{\lambda\mu}(\boldsymbol{x}) - \Phi_\lambda(\boldsymbol{x})\| \leq \kappa\sqrt{\mu}$.

for all $\boldsymbol{x} \in \mathbb{R}^n$, *and* $\mu, \mu_1, \mu_2 \geq 0$, *where* $\kappa = \sqrt{n(4-\lambda)}$.

Proof. Using the Euclidean norm and Proposition 2.1,

$$\begin{aligned} \|\Phi_{\lambda\mu_1}(\boldsymbol{x}) - \Phi_{\lambda\mu_2}(\boldsymbol{x})\| &= \sqrt{\sum_{i=1}^{n} \left[\varphi_{\lambda\mu_1}(x_i, F_i(\boldsymbol{x})) - \varphi_{\lambda\mu_2}(x_i, F_i(\boldsymbol{x}))\right]^2} \\ &\leq \sqrt{n\left[(\sqrt{\mu_1} - \sqrt{\mu_2})\sqrt{4-\lambda}\right]^2} = \sqrt{n(4-\lambda)}|\sqrt{\mu_1} - \sqrt{\mu_2}|. \end{aligned}$$

The second part of the proposition is obtained by choosing $\mu_1 = \mu$ and $\mu_2 = 0$. ☑

Now, the basic iteration of a smoothing Jacobian method for solving $\Phi_\lambda(\boldsymbol{x}) = \mathbf{0}$ is as follows

$$\begin{aligned} \Phi'_{\lambda\mu}(\boldsymbol{x}^k)\boldsymbol{s}_k &= -\Phi_\lambda(\boldsymbol{x}^k), \\ \boldsymbol{x}_{k+1} &= \boldsymbol{s}_k + \boldsymbol{x}^k, \end{aligned} \tag{9}$$

where $\Phi'_{\lambda\mu}(\boldsymbol{x}^k)$ is the Jacobian matrix of $\Phi_{\lambda\mu}$ at $\boldsymbol{x}^k$. From (6) and (9), we have that this method solves the reformulation $\Phi_\lambda(\boldsymbol{x}) = 0$, replacing $H_k \in \partial_C \Phi_\lambda(\boldsymbol{x})$, with an approximation $\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)$. Thus, these methods can be seen as quasi-Newton. The Jacobian matrix $\Phi'_{\lambda\mu}(\boldsymbol{x})$ is given by

$$\Phi'_{\lambda\mu}(\boldsymbol{x}) \;=\; \begin{pmatrix} \nabla\varphi_{\lambda\mu}(x_1, F_1(\boldsymbol{x}))^T \\ \vdots \\ \nabla\varphi_{\lambda\mu}(x_n, F_n(\boldsymbol{x}))^T \end{pmatrix},$$

where $\nabla\varphi_{\lambda\mu}(x_i, F_i(\boldsymbol{x}))^T \;=\; (\alpha_{\lambda\mu}(x_i, F_i(\boldsymbol{x})) - 1)\mathbf{e}_i^T + (\beta_{\lambda\mu}(x_i, F_i(\boldsymbol{x})) - 1)\,\nabla F_i(\boldsymbol{x})^T$, with

$$\alpha_{\lambda\mu}(x_i, F_i(\boldsymbol{x})) = \frac{2(x_i - F_i(\boldsymbol{x})) + \lambda F_i(\boldsymbol{x})}{G_{\lambda\mu}(x_i, F_i(\boldsymbol{x}))} \quad \text{and} \quad \beta_{\lambda\mu}(x_i, F_i(\boldsymbol{x})) = \frac{-2(x_i - F_i(\boldsymbol{x})) + \lambda x_i}{G_{\lambda\mu}(x_i, F_i(\boldsymbol{x}))},$$

where $G_{\lambda\mu}(x_i, F_i(\boldsymbol{x})) = 2\sqrt{(x_i - F_(\boldsymbol{x}))^2 + \lambda x_i F_i(\boldsymbol{x}) + (4-\lambda)\mu}$.

The next proposition guarantees that, if μ tends to zero, the distance between $\Phi'_{\lambda\mu}(\boldsymbol{x})$ and $\partial_C\Phi_\lambda(\boldsymbol{x})$ also tends to zero. Thus, it makes sense to replace the Newton iteration (6) with (9).

Proposition 2.3 ([4]). *Let $\boldsymbol{x} \in \mathbb{R}^n$ be arbitrary but fixed. Then we have* $\lim_{\mu\to 0} dist(\Phi'_{\lambda\mu}(\boldsymbol{x}), \partial_C\Phi_\lambda(\boldsymbol{x})) \;=\; 0$.

From this proposition, for every $\delta > 0$ there exists $\bar\mu = \bar\mu(\boldsymbol{x}, \delta) > 0$ such that $dist(\Phi'_{\lambda\mu}(\boldsymbol{x}), \partial_C\Phi_\lambda(\boldsymbol{x})) \le \delta$, for all $0 < \mu \le \bar\mu$. In our algorithmic proposal is very important to obtain an expression of $\bar\mu$ since it gives an upper bound of μ. For this, we proceed as in [18] and we obtain the following proposition.

Proposition 2.4 ([30]). *Let $\boldsymbol{x} \in \mathbb{R}^n$ be arbitrary but fixed. Assume that $\boldsymbol{x}$ is not a solution of the* NCP *and define*

$$\begin{aligned} \gamma(\boldsymbol{x}) &= \frac{1}{2}\max_{i\notin\beta(\boldsymbol{x})}\left\{\left\|\left[2(x_i - F_i(\boldsymbol{x})) + \lambda F_i(\boldsymbol{x})\right]\mathbf{e}_i + \left[-2(x_i - F_i(\boldsymbol{x})) + \lambda x_i\right]\nabla F_i(\boldsymbol{x})\right\|\right\} \\ \alpha(\boldsymbol{x}) &= \min_{i\notin\beta(\boldsymbol{x})}\left\{(x_i - F_i(\boldsymbol{x}))^2 + \lambda x_i F_i(\boldsymbol{x})\right\} > 0, \end{aligned}$$

with $\beta(\boldsymbol{x}) = \{i\colon \boldsymbol{x}_i = 0 = F_i(\boldsymbol{x})\}$. Let $\delta > 0$ and define

$$\overline{\mu}\,(\boldsymbol{x}, \delta, \lambda) \;=\; \begin{cases} 1 & \text{if} \quad \dfrac{n\gamma(\boldsymbol{x})^2}{\delta^2} - \alpha(\boldsymbol{x}) \le 0. \\ \dfrac{\alpha(\boldsymbol{x})^2}{4-\lambda}\left(\dfrac{\delta^2}{n\gamma(\boldsymbol{x})^2 - \delta^2\alpha(\boldsymbol{x})}\right) & \text{other case.} \end{cases}$$

Then $dist_F(\Phi'_{\lambda\mu}(\boldsymbol{x}), \partial_C\Phi_\lambda(\boldsymbol{x})) \le \delta$, for all μ such that $0 < \mu < \overline{\mu}\,(\boldsymbol{x}, \delta, \lambda)$.

Since our algorithmic proposal is a global algorithm for solving the NCP, indirectly through its reformulation $\Phi_\lambda(\boldsymbol{x}) = \mathbf{0}$, we consider the natural merit function $\Psi_\lambda\colon \mathbb{R}^n \to \mathbb{R}$ defined by $\Psi_\lambda(\boldsymbol{x}) \;=\; \frac{1}{2}\Phi_\lambda(\boldsymbol{x})^T\Phi_\lambda(\boldsymbol{x})$. The idea is to solve the NCP by minimizing

Ψ_λ. But, there is a problem: the direction computed from (9), is no necessarily a descent direction for Ψ_λ at $\boldsymbol{x}^k$. Following [18], an alternative is to use this direction to reduce the related merit function

$$\Psi_{\lambda\mu}(\boldsymbol{x}) = \frac{1}{2}\Phi_{\lambda\mu}(\boldsymbol{x})^T\Phi_{\lambda\mu}(\boldsymbol{x}). \tag{10}$$

3. Preliminaries

In this section, we present some definitions, propositions, and lemmas related to the nonlinear complementarity which will be useful in the development of the convergence theory of our algorithmic proposal.

Definition 3.1. Let $A \in \mathbb{R}^{n\times n}$. The *Frobenius* norm of A is defined by

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\text{trace}\,(A^TA)}.$$

Definition 3.2. Let $A \in \mathbb{R}^{n\times n}$ and $\mathcal{M} \subseteq \mathbb{R}^{n\times n}$ be a nonempty set of matrices. The distance between A and $\mathcal{M}$ is defined by, $dist(A,\mathcal{M}) = \inf_{B\in\mathcal{M}}\{\|A-B\|\}$, where $\|\cdot\|$ is a matrix norm.

Definition 3.3. Let $\{t_k\} \subseteq \mathbb{R}$, if there exists a number U such that

1. For every $\epsilon > 0$ there exists an integer N such that $k > N$ implies $t_k < U + \epsilon$.
2. Given $\epsilon > 0$ and $m > 0$, there exists an integer $k > m$ such that $t_k > U - \epsilon$.

Then U is called the superior limit of $\{t_k\}$ and we write $U = \limsup_{k\to\infty} t_k$.

Related with a solution $\boldsymbol{x}^*$ of the NCP, we have the following index sets,

$$\hat{\alpha} = \{i\colon \boldsymbol{x}_i^* > 0 = F_i(\boldsymbol{x}^*)\}, \qquad \hat{\beta} = \{i\colon \boldsymbol{x}_i^* = 0 = F_i(\boldsymbol{x}^*)\}, \qquad \hat{\gamma} = \{i\colon \boldsymbol{x}_i^* < 0 = F_i(\boldsymbol{x}^*)\}.$$

When $\beta \neq \emptyset$, $\boldsymbol{x}^*$ is called a *degenerate solution.*

Definition 3.4. Let $\boldsymbol{x}^*$ be a solution of the *NCP.*

1. If all matrices in $\partial_B\Phi_\lambda(\boldsymbol{x}^*)$ are nonsingular, $\boldsymbol{x}^*$ is called a BD-regular solution.
2. If the submatrix [1] $F'(\boldsymbol{x}^*)_{\hat{\alpha}\hat{\alpha}}$ is nonsingular and the Schur complement

$$F'(\boldsymbol{x}^*)_{\hat{\beta}\hat{\beta}} - F'(\boldsymbol{x}^*)_{\hat{\beta}\hat{\alpha}}F'(\boldsymbol{x}^*)^{-1}_{\hat{\alpha}\hat{\alpha}}F'(\boldsymbol{x}^*)_{\hat{\alpha}\hat{\beta}}$$

is a P-matrix[2], $\boldsymbol{x}^*$ is called an *R-regular solution.*

[1] Given $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ and the index sets η and τ, $A_{\eta\tau}$ is the matrix with components a_{ij}, $i \in \eta$ and $j \in \tau$.

[2] The matrix $M \in \mathbb{R}^{n\times m}$ is called a *P-matrix*, if for every nonzero vector *and* $\in \mathbb{R}^n$ exists an index $i_0 = i_0(\boldsymbol{y}) \in \{1,\ldots,n\}$ such that $y_{i_0}[M\boldsymbol{y}]_{i_0} \geq 0$.

Definition 3.5. Given the sequences $\{\alpha_k\}$ and $\{\beta_k\}$ such that $\beta_k \geq 0$, for all k, and $\alpha_k = O(\beta_k)$, if there exists a positive constant M such that $|\alpha_k| \leq M\beta_k$, for all k. We write $\alpha_k = o(\beta_k)$, if $\lim_{k\to\infty} \alpha_k/\beta_k = 0$.

Proposition 3.6 ([17], [28]). *Assume that $\{\boldsymbol{x}^k\} \subseteq \mathbb{R}^n$ is a sequence converging to $\boldsymbol{x}^*$. Then,*

1. *The function Φ_λ is semismooth, so $\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_\lambda(\boldsymbol{x}^*) - H_k(\boldsymbol{x}^k - \boldsymbol{x}^*)\| = o(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|)$, for any $H_k \in \partial_C \Phi_\lambda(\boldsymbol{x}^k)$.*
2. *If F is continuously differentiable with locally Lipchitz Jacobian then Φ_λ is strongly semismooth; that is, $\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_\lambda(\boldsymbol{x}^*) - H_k(\boldsymbol{x}^k - \boldsymbol{x}^*)\| = O(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2)$, for any $H_k \in \partial_C \Phi_\lambda(\boldsymbol{x}^k)$.*

Proposition 3.7 ([18]). *Given $\boldsymbol{x} \in \mathbb{R}^n$ and $\mu > 0$ arbitrary but fixed, then*

$$\left[dist_F(\nabla \Phi_{\lambda\mu}(\boldsymbol{x}), \partial_C \Phi(\boldsymbol{x})^T)\right]^2 = \sum_{i=1}^{n} \left[dist_2([\nabla \Phi_{\lambda\mu}(\boldsymbol{x})]_i, \partial \Phi_{\lambda,i}(\boldsymbol{x}))\right]^2.$$

Proposition 3.8 ([23]). *If $\boldsymbol{x}^* \in \mathbb{R}^n$ is an isolated accumulation point [3] of a sequence $\{\boldsymbol{x}^k\} \subseteq \mathbb{R}^n$ such that $\{\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|\}_L$ converges to zero, for any subsequence $\{\boldsymbol{x}^k\}_L$ converging to $\boldsymbol{x}^*$. Then $\{\boldsymbol{x}^k\}$ converges to $\boldsymbol{x}^*$.*

Proposition 3.9 ([13]). *Let $G\colon \mathbb{R}^n \longrightarrow \mathbb{R}^n$ be locally Lipschitz and $\boldsymbol{x}^* \in \mathbb{R}^n$ such that $G(\boldsymbol{x}^*) = \boldsymbol{0}$, satisfary all matrices in $\partial G(\boldsymbol{x}^*)$ are nonsingular and assume that there exist two sequences $\{\boldsymbol{x}^k\} \subseteq \mathbb{R}^n$ and $\{\boldsymbol{d}^k\} \subseteq \mathbb{R}^n$ with*

$$\lim_{k\to\infty} \boldsymbol{x}^k = \boldsymbol{x}^* \qquad \text{and} \qquad \|\boldsymbol{x}^k + \boldsymbol{d}^k - \boldsymbol{x}^*\| = o\left(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|\right).$$

Then $\|G(\boldsymbol{x}^k + \boldsymbol{d}^k)\| = o\left(\|G(\boldsymbol{x}^k)\|\right)$.

Proposition 3.10 ([30]). *Let $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n$ such that $\|\boldsymbol{x} - \boldsymbol{z}\| \leq \alpha \|\boldsymbol{x}\|$, $\alpha \in (0,1)$. Then $\boldsymbol{x}^T(\boldsymbol{x} - \boldsymbol{z}) \leq \alpha \|\boldsymbol{x}\|^2$.*

Proposition 3.11. [17] *Let $\boldsymbol{x} \in \mathbb{R}^n$ arbitrary. Then*

$$\partial_C \Phi_\lambda(\boldsymbol{x}) = D_a(\boldsymbol{x}) + D_b(\boldsymbol{x}) F'(\boldsymbol{x}),$$

where $D_a = diag(a_1(\boldsymbol{x}), \ldots, a_n(\boldsymbol{x}))$, $D_b = diag(b_1(\boldsymbol{x}), \ldots, b_n(\boldsymbol{x}))$ are diagonal matrices in $\mathbb{R}^{n\times n}$.

- *If $(\boldsymbol{x}_i, F(\boldsymbol{x}_i)) \neq (0,0)$, then*

$$a_i(\boldsymbol{x}) = \frac{2(x_i - F_i(\boldsymbol{x})) + \lambda F_i(\boldsymbol{x})}{2\sqrt{(x_i - F_i(\boldsymbol{x}))^2 + \lambda x_i F_i(\boldsymbol{x})}} - 1 = \alpha_\lambda(x_i, F_i(\boldsymbol{x})) - 1,$$

$$b_i(\boldsymbol{x}) = \frac{-2(x_i - F_i(\boldsymbol{x})) + \lambda x_i}{2\sqrt{(x_i - F_i(\boldsymbol{x}))^2 + \lambda x_i F_i(\boldsymbol{x})}} - 1 = \beta_\lambda(x_i, F_i(\boldsymbol{x})) - 1.$$

[3] If Ω is the set of accumulation points of $\{\boldsymbol{x}^k\}$, we say that $\boldsymbol{x}^* \in \Omega$ is an isolated accumulation point, if exist $\delta > 0$ such that $\mathcal{B}(\boldsymbol{x}; \delta) = \{\boldsymbol{x}^*\}$.

- *If* $(\boldsymbol{x}_i, F(\boldsymbol{x}_i)) = (0,0)$ *then* $a_i(\boldsymbol{x}) = \sigma_i - 1$ *and* $b_i(\boldsymbol{x}) = \rho_i - 1$, *for any* $(\sigma_i, \rho_i) \in \mathbb{R}^2$ *such that* $\|(\sigma_i, \rho_i)\| \leqslant \sqrt{c_\lambda}$, $c_\lambda \in (0,2)$.

Proposition 3.12 ([17]). *The merit function* Ψ_λ *is continuously differentiable and* $\nabla\Psi_\lambda(\boldsymbol{x}) = V^T\Phi_\lambda(\boldsymbol{x})$, *for any* $V \in \partial_C\Phi_\lambda(\boldsymbol{x})$.

Lemma 3.13 ([30]). *Let* $\mu > 0$ *and* $\lambda \in (0,4)$. *The function* $h\colon (0,\infty) \to \mathbb{R}$, *defined by*

$$h(t) \;=\; \frac{1}{\sqrt{t}} - \frac{1}{\sqrt{t + (4-\lambda)\mu}},$$

is strictly decreasing.

Lemma 3.14 ([30]). *Let* $\{\boldsymbol{x}^k\} \subseteq \mathbb{R}^n$ *and* $\{\mu_k\} \subseteq \mathbb{R}$ *two sequences such that* $\{\boldsymbol{x}^k\} \to \boldsymbol{x}^*$ *for some* $\boldsymbol{x}^* \in \mathbb{R}^n$ *and* $\{\mu_k\} \to 0$. *Then* $\lim_{k\to\infty} \nabla\Psi_{\lambda\mu_k}(\boldsymbol{x}^k) = \nabla\Psi_\lambda(\boldsymbol{x}^*)$ *and* $\lim_{\mu\to 0} \Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\Phi_\lambda(\boldsymbol{x}^k) = \nabla\Psi_\lambda(\boldsymbol{x}^*)$.

Lemma 3.15 ([18]). *Let* $\{\boldsymbol{x}^k\}, \{\boldsymbol{d}^k\} \subseteq \mathbb{R}^n$ *and* $\{t_k\} \subseteq \mathbb{R}$ *with* $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + t_k\boldsymbol{d}^k$ *such that* $\boldsymbol{x}^k \to \boldsymbol{x}^*$, $\boldsymbol{d}^k \to \boldsymbol{d}^*$ *and* $\{t_k\} \to 0$ *for some vectors* $\boldsymbol{x}^*$, $\boldsymbol{d}^* \in \mathbb{R}^n$. *Moreover, consider* $\{\mu_k\} \subseteq \mathbb{R}$ *a sequence such that* $\{\mu_k\} \to 0$. *Then*

$$\lim_{k\to\infty} \frac{\Psi_{\lambda\mu_k}(\boldsymbol{x}^k + t_k\boldsymbol{d}^k) - \Psi_{\lambda\mu_k}(\boldsymbol{x}^k)}{t_k} \;=\; \nabla\Psi_\lambda(\boldsymbol{x}^*)^T\boldsymbol{d}^*.$$

4. New algorithm

In this section, we propose a new algorithm for solving the NCP. Basically, the proposal is a generalization of the smoothing Jacobian method proposed in [18], and its basic iteration is given in (9). To guarantee the algorithm to be well-defined for an arbitrary NCP, we use a gradient step for Ψ_λ when linear system (11) solution does not exist or gives a poor descent direction for $\Psi_{\lambda\mu}$.

Algorithm 1. (*Smoothing Jacobian method*),

S0. $\boldsymbol{x}^0 \in \mathbb{R}^n$; θ, α, η and $\rho \in (0,1)$, $\gamma > 0$, $\sigma \in \left(0, \frac{1-\alpha}{2}\right)$; $p > 2$ and $\epsilon \geq 0$. Let $\beta_0 = \|\Phi_\lambda(\boldsymbol{x}^0)\|$, $\kappa = \sqrt{(4-\lambda)\eta}$, $\mu_0 = \left(\frac{\alpha}{2\kappa}\beta_0\right)^2$ and $k = 0$.

S1. If $\|\nabla\Psi_\lambda(\boldsymbol{x}^k)\| \leq \epsilon$, stop.

S2. Find $\boldsymbol{d}^k \in \mathbb{R}^n$ solving the linear system of equations,

$$\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\boldsymbol{d}^k \;=\; -\Phi_\lambda(\boldsymbol{x}^k). \tag{11}$$

If the system (11) is not solvable or if the condition

$$\Phi_\lambda(\boldsymbol{x}^k)^T\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\boldsymbol{d}^k \leq -\rho\|\boldsymbol{d}^k\|^p \tag{12}$$

is not satisfied, set

$$\boldsymbol{d}^k \;=\; -\nabla\Psi_\lambda(\boldsymbol{x}^k). \tag{13}$$

S3. Find the smallest $m_k \in \{0, 1, 2, \ldots\}$ such that

$$\Psi_{\lambda\mu_k}(\boldsymbol{x}^k + \theta^{m_k}\boldsymbol{d}^k) \leq \Psi_{\lambda\mu_k}(\boldsymbol{x}^k) - 2\sigma\theta^{m_k}\Psi_\lambda(\boldsymbol{x}^k), \tag{14}$$

if $\boldsymbol{d}_k$ is given by (11), and such that

$$\Psi_\lambda(\boldsymbol{x}^k + \theta^{m_k}\boldsymbol{d}^k) \leq \Psi_\lambda(\boldsymbol{x}^k) - \sigma\theta^{m_k}\|\boldsymbol{d}^k\|^2 \tag{15}$$

if $\boldsymbol{d}^k$ is given by (13). Set $t_k = \theta^{m_k}$ and $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + t_k\boldsymbol{d}^k$.

S4. If

$$\|\Phi_\lambda(\boldsymbol{x}^{k+1})\| \leq \max\left\{\eta\beta_k, \frac{1}{\alpha}\|\Phi_\lambda(\boldsymbol{x}^{k+1}) - \Phi_{\lambda\mu_k}(\boldsymbol{x}^{k+1})\|\right\}, \tag{16}$$

set $\beta_{k+1} = \|\Phi_\lambda(\boldsymbol{x}^{k+1})\|$ and choose μ_{k+1} such that,

$$0 < \mu_{k+1} \leq \min\left\{\left(\frac{\alpha}{2\kappa}\beta_{k+1}\right)^2, \frac{\mu_k}{4}, \bar{\mu}\left(\boldsymbol{x}^{k+1}, \gamma\beta_{k+1}\right)\right\}. \tag{17}$$

If (16) is not satisfied and $\boldsymbol{d}_k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ then set $\beta_{k+1} = \beta_k$, and choose μ_{k+1} such that

$$0 < \mu_{k+1} \leq \min\left\{\left(\frac{\alpha}{2\kappa}\|\Phi_\lambda(\boldsymbol{x}^{k+1})\|\right)^2, \left(\frac{\|\Phi_\lambda(\boldsymbol{x}^k)\| - \|\Phi_\lambda(\boldsymbol{x}^{k+1})\|}{2\kappa}\right)^2, \frac{\mu_k}{4}\right\}. \tag{18}$$

If none of the above conditions is met, set $\beta_{k+1} = \beta_k$ and $\mu_{k+1} = \mu_k$.

S5. Set $k = k + 1$ and return to **S1**.

In **S0**, we introduce the parameters and initialize the variables. In **S1**, it is natural to think that the algorithm stops when the gradient of the merit function becomes too small. However, in the implementation, we add other classic criteria like maximum number of allowed iterations and one that prevents the algorithm from no finding an adequate step size. In **S2**, the calculus of a search direction is perhaps the main step of the algorithm: we find $\boldsymbol{d}_k$ by mixed Newton equation (11). In case of (11) is not solvable or the direction does not satisfy descent criteria (12), we use the steepest descent direction of the merit function (13), which guarantees a descent direction of Ψ_λ.

After finding the descent direction, the algorithm is globalized in step **S3** using a line search which depends on this direction: if it is obtained by the Newton equation (11), the line search is made using (14), which is also used in [9] as a global strategy. On the other hand, if it is the steepest descent direction (13), the line search (15) is type *Armijo* [12]. The update of μ_k, in **S4**, starts with the condition (16) used in en [9]. If it is satisfied, μ_k is updated by (17). This guarantees that the distance between the subdifferential and smoothing Jacobian is small, and that μ_k tends to zero. If (16) is not satisfied, μ_k is updated by (18). The conditions (18) and (16) are essential to guarantee the algorithm to be well defined and to converge globally. For the convergence analysis of the algorithm, we define the following set

$$K = \{0\} \cup \left\{k \in \mathbb{N}\colon \|\Phi_\lambda(\boldsymbol{x}^k)\| \leq \max\left\{\eta\,\beta_{k-1},\, \frac{1}{\alpha}\,\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_{\lambda\mu_{k-1}}(\boldsymbol{x}^k)\|\right\}\right\}, \tag{19}$$

which is motivated by condition (16). Moreover, $K = \{0\} \cup K_1 \cup K_2$ (disjoint union), where

$$\begin{aligned} K_1 &= \left\{k \in K:\ \eta\beta_{k-1} \geq \tfrac{1}{\alpha}\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_{\lambda\mu_{k-1}}(\boldsymbol{x}^k)\|\right\}, \\ K_2 &= \left\{k \in K:\ \eta\beta_{k-1} < \tfrac{1}{\alpha}\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_{\lambda\mu_{k-1}}(\boldsymbol{x}^k)\|\right\}. \end{aligned}$$

The next proposition is useful to demonstrate that the algorithm is well-defined. Its demonstration is analogous to that given in [18].

Proposition 4.1. *The following inequalities hold:*

a) $\|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_{\lambda\mu_k}(\boldsymbol{x}^k)\| \leq \alpha\|\Phi_\lambda(\boldsymbol{x}^k)\|$, *for all* $k \geq 0$.

b) $dist_F(\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k), \partial_C\Phi_\lambda(\boldsymbol{x}^k)) \leq \gamma\|\Phi_\lambda(\boldsymbol{x}^k)\|$, *for all* $k \in K$ *with* $k \geq 1$.

The following result guarantees that Algorithm 1 is well-defined: it ends in a finite number of steps.

Proposition 4.2. Algorithm 1 *is well-defined.*

Proof. It is essentially the same given in [9]. It is sufficient to prove that m_k in **S3** is finite, for all $k \in \mathbb{N}$. In effect, if a descent direction is given by (13) then the condition Armijo-type guarantees that m_k is finite [18]. On the other hand, let assume that the direction is given by (11). Since $\Psi_{\lambda\mu_k}$ is continuously differentiable and its gradient is given by $\nabla\Psi_{\lambda\mu_k}(\boldsymbol{x}^k) = \Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_{\lambda\mu_k}(\boldsymbol{x}^k)$, by Taylor's Theorem, we have

$$\Psi_{\lambda\mu_k}(\boldsymbol{x}^{k+1}) - \Psi_{\lambda\mu_k}(\boldsymbol{x}^k) = t\nabla\Psi_{\lambda\mu_k}(\boldsymbol{x}^k)^T\boldsymbol{d}^k + o(t) = t\Phi_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\boldsymbol{d}^k + o(t). \quad (20)$$

Using the Newton's direction (11) in (20),

$$\begin{aligned} t\Phi_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\boldsymbol{d}^k + o(t) &= -t\Phi_\lambda(\boldsymbol{x}^k)^T\Phi_{\lambda\mu_k}(\boldsymbol{x}^k) + o(t) \\ &= -t\Phi_\lambda(\boldsymbol{x}^k)^T\left(\Phi_{\lambda\mu_k}(\boldsymbol{x}^k) - \Phi_\lambda(\boldsymbol{x}^k) + \Phi_\lambda(\boldsymbol{x}^k)\right) + o(t) \\ &= -t\Phi_\lambda(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k) + \\ &\qquad t\Phi_\lambda(\boldsymbol{x}^k)^T\left(\Phi_{\lambda\mu_k}(\boldsymbol{x}^k) - \Phi_\lambda(\boldsymbol{x}^k)\right) + o(t) \\ &\leq -2t(1-\alpha)\Psi_\lambda(\boldsymbol{x}^k) + o(t), \end{aligned} \quad (21)$$

where the last inequality follows from Propositions 3.10 and 4.1. On the other hand,

$$\sigma < \frac{1}{2}(1-\alpha) < 1 - \alpha. \quad (22)$$

Therefore, from (20), (21) and (22), $\Psi_{\lambda\mu_k}(\boldsymbol{x}^{k+1}) - \Psi_{\lambda\mu_k}(\boldsymbol{x}^k) \leq -2t\sigma\Psi_\lambda(\boldsymbol{x}^k) + o(t)$. This allows to conclude that exists a nonnegative integer m_k such that (14) is satisfied. ☑

5. Global convergence

In this section, we present the convergence results of the algorithm proposed. Basically, we prove that any accumulation point of the sequence generated by Algorithm 1 is a stationary point of Ψ_λ. These results generalize those presented in [18] for an algorithm using the Fischer-Burmeister function, which in turn is based on the theory developed in [9] for smoothing Newton-type methods. The proofs of the theorems and lemmas in this section and in the next section use the same steps as the proofs of the corresponding theorems in [18].

We start with two lemmas whose proofs use the updating rules of Algorithm 1.

Lemma 5.1. *Assume that a sequence generated by* Algorithm 1 *has an accumulation point* $\boldsymbol{x}^*$, *which is a solution of the* NCP. *Then the index set* K *defined by* (19) *is infinite and the sequence* $\{\mu_k\} \to 0$.

Proof. Assume that K is finite. For the updating rule (16) there exists an integer k_0, such that $\beta_k = \beta_{k_0}$. Moreover, $\|\Phi_\lambda(\boldsymbol{x}^{k+1})\| > \eta\beta_{k_0}$, for all $k \geq k_0$, which contradicts that $\boldsymbol{x}^*$ is a solution of the NCP. ☑

Lemma 5.2. *The following statements hold:*

a) *if* $\boldsymbol{d}^k$ *is given by* (11) *then* $\|\Phi_{\lambda\mu_k}(\boldsymbol{x}^{k+1})\| < \|\Phi_{\lambda\mu_k}(\boldsymbol{x}^k)\|$.

b) *if* $\boldsymbol{d}^k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ *and* μ_k *is updated by* (18) *then* $\|\Phi_{\lambda\mu_{k+1}}(\boldsymbol{x}^{k+1})\| \leq \|\Phi_{\lambda\mu_{k+1}}(\boldsymbol{x}^k)\|$.

Proof. Part a) is verified using the updating rule (14), and part b) is true if (16) is satisfied. In addition, we use Proposition 2.2 and some algebraic manipulations. ☑

The next corollary is an important consequence of the previous result.

Corollary 5.3. *If* $k \notin K$ *then* $\|\Phi_{\lambda\mu_k}(\boldsymbol{x}^k)\| \leq \|\Phi_{\lambda\mu_k}(\boldsymbol{x}^{k-1})\|$.

Proof. The proof is the same as Corollary 5.3 in [18]. ☑

The two following results are technical lemmas; they give some bounds that we use in the proof of Proposition 5.6. Details of the proofs are given in [30].

Lemma 5.4 ([30]). *Assume that* K *is ordered like this* $k_0 = 0 < k_1 < k_2 < \ldots$. *Let* $k \in \mathbb{N}$ *be arbitrary but fixed and* k_j *the largest integer in* K *such that* $k_j \leq k$ *then* $\|\Phi_\lambda(\boldsymbol{x}^k)\| \leq \beta_{k_j} + 2\kappa\sqrt{\mu_{k_j}}$.

Lemma 5.5 ([30]). *Assume that* K *is ordered like this* $k_0 = 0 < k_1 < k_2 < \ldots$. *Let* $k \in \mathbb{N}$ *be arbitrary but fixed and* k_j *the largest integer in* K *such that* $k_j \leq k$. *Then* $\sqrt{\mu_{k_j}} \leq 2^{-(j+1)}\,\frac{\alpha}{\kappa}\|\Phi_\lambda(\boldsymbol{x}^0)\|$ *and* $\beta_{k_j} \leq r^j\|\Phi_\lambda(\boldsymbol{x}^0)\|$, *where*

$$r = \max\left\{\frac{1}{2}, \eta\right\}. \tag{23}$$

Using the two previous lemmas, we prove that the iterations $\boldsymbol{x}^k$ stay at the same level set. This is important because we minimize different merit functions and a decrease in one does not imply a decrease in the other. This set can be arbitrary close to the level set $\Psi_\lambda(\boldsymbol{x}^0)$.

Proposition 5.6. *The sequence generated by* Algorithm 1 *stays in the level set*

$$L_0 = \left\{\boldsymbol{x} \in \mathbb{R}^n \colon\ \Psi_\lambda(\boldsymbol{x}) \leq (1+\alpha)^2 \Psi_\lambda(\boldsymbol{x}^0)\right\}.$$

Proof. We assume, without loss of generality, that the set K given by (19) is ordered like this $k_0 = 0 < k_1 < k_2 < \ldots$, which does not indicate that K is infinite. We consider $k \in \mathbb{N}$, arbitrary but fixed and k_j the largest integer in K such that $k_j \leq k$. From Lemmas 5.4 and 5.5, we have $\|\Phi_\lambda(\boldsymbol{x}^k)\| \leq \beta_{k_j} + 2\,\kappa\,\sqrt{\mu_{k_j}} \leq r^j\,\|\Phi_\lambda(\boldsymbol{x}^0)\| + \frac{\alpha}{2^j}\,\|\Phi_\lambda(\boldsymbol{x}^0)\|$, where r is defined by (23), that is.

$$\|\Phi_\lambda(\boldsymbol{x}^k)\| \leq r^j(1+\alpha)\|\Phi_\lambda(\boldsymbol{x}^0)\|, \tag{24}$$

then $\|\Phi_\lambda(\boldsymbol{x}^k)\| \leq (1+\alpha)\|\Phi_\lambda(\boldsymbol{x}^0)\|$. Therefore, $\boldsymbol{x}^k \in L_0$. ☑

The following proposition is a consequence of inequality (24).

Proposition 5.7. *Let* $\{\boldsymbol{x}^k\}$ *be a sequence generated by* Algorithm 1 *and assume that the set* K *is infinite. Then each accumulation point of* $\{\boldsymbol{x}^k\}$ *is a solution of the* NCP.

Proposition 5.8. *Let* $\{\boldsymbol{x}^k\}$ *be a sequence generated by* Algorithm 1 *and let* $\{\boldsymbol{x}^k\}_L$ *be a subsequence converging to* $\boldsymbol{x}^* \in \mathbb{R}^n$. *If* $\boldsymbol{d}^k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ *for all* $k \in L$, *then* $\boldsymbol{x}^*$ *is a stationary point of* Ψ_λ.

Proof. If K is infinite, the accumulation point, $\boldsymbol{x}^*$ is a solution of the NCP by Proposition 5.7. Thus, this is a global minimum and therefore, a stationary point of Ψ_λ.

If K is finite, we can assume that $K \cap L = \emptyset$ since the sequence has an infinite number of elements. Therefore, the updating rule (18) is satisfied for all $k \in L$, and consequently, the sequence $\{\mu_k\}$ converges to zero. Since K is finite there exists the largest element that we call $\hat{k}$. Using the updating rules defined in step **S4** of Algorithm 1, we have the following inequalities

$$\mu_k \leq \mu_{\hat{k}}, \quad \beta_k = \beta_{\hat{k}} = \|\Phi_\lambda(\boldsymbol{x}^{\hat{k}})\| \tag{25}$$

$$\|\Phi_\lambda(\boldsymbol{x}^k)\| > \eta\,\beta_{k-1} = \eta\,\|\Phi_\lambda(\boldsymbol{x}^{\hat{k}})\| > 0. \tag{26}$$

$$\alpha\|\Phi_\lambda(\boldsymbol{x}^k)\| > \|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_{\lambda\mu_{k-1}}(\boldsymbol{x}^k)\|. \tag{27}$$

Let assume by contradiction that $\boldsymbol{x}^*$ is not a stationary point of Ψ_λ. That is, $\nabla\Psi_\lambda(\boldsymbol{x}^*) \neq 0$. First, we prove that $\liminf_{k\in L} t_k = 0$. For this, let assume $\liminf_{k\in L} t_k = t^* > 0$. Since $d^k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ for all $k \in L$, using Armijo-rule (15), we have that

$$\Psi_\lambda(\boldsymbol{x}^{k+1}) - \Psi_\lambda(\boldsymbol{x}^k) \leq -\sigma t_k \|\nabla\Psi_\lambda(\boldsymbol{x}^k)\|^2 \leq -\frac{c}{2}, \tag{28}$$

for all $k \in L$ sufficiently large, where $c = \sigma\, t_* \,\|\nabla\Psi_\lambda(\boldsymbol{x}^*)\|^2 > 0$. Since $\{\mu_k\}$ converges to zero, Proposition 2.2 guarantees that, for all $k \in \mathbb{N}$, sufficiently large,

$$|\Psi_{\lambda\mu_k}(\boldsymbol{x}^{k+1}) - \Psi_\lambda(\boldsymbol{x}^{k+1})| \,\leq\, \frac{c}{4} \qquad \text{and} \qquad |\Psi_{\lambda\mu_k}(\boldsymbol{x}^{k}) - \Psi_\lambda(\boldsymbol{x}^{k})| \,\leq\, \frac{c}{4}.$$

Using again that $\{\mu_k\}$ converges to zero, the sequence $\{\|\Phi_\lambda(\boldsymbol{x}^k)\|\}$ is bounded; since K is finite, we have, by Proposition 5.6, that $2\,\kappa\,\sqrt{\mu_k}\,\|\Phi_\lambda(\boldsymbol{x}^k)\| + 2\kappa^2\mu_k \,\leq\, \frac{c}{4}$, for all $k \in \mathbb{N}$, sufficiently large. If $L = \{l_0, l_1, \dots\}$ then, for all l_j, sufficiently large, we have, in analogous way than in Proposition 5.6, that,

$$\begin{aligned}\Psi_\lambda(\boldsymbol{x}^{l_{j+1}}) &= \tfrac{1}{2}\|\Phi_\lambda(\boldsymbol{x}^{l_{j+1}})\| \leq \tfrac{1}{2}(\|\Phi_\lambda(\boldsymbol{x}^{l_j+1})\| + 2\kappa\sqrt{\mu_{l_j+1}})^2 \\ &= \Psi_\lambda(\boldsymbol{x}^{l_j+1}) + 2\kappa\sqrt{\mu_{l_j+1}}\,\|\Phi_\lambda(\boldsymbol{x}^{l_j+1})\| + 2\kappa^2\mu_{l_j+1} \,\leq\, \Psi_\lambda(\boldsymbol{x}^{l_j+1}) + \frac{c}{4}.\end{aligned} \tag{29}$$

From (28) and (29), $\Psi_\lambda(\boldsymbol{x}^{l_{j+1}}) - \Psi_\lambda(\boldsymbol{x}^{l_j}) = [\Psi_\lambda(\boldsymbol{x}^{l_{j+1}}) - \Psi_\lambda(\boldsymbol{x}^{l_j+1})] + [\Psi_\lambda(\boldsymbol{x}^{l_j+1}) - \Psi_\lambda(\boldsymbol{x}^{l_j})] \leq -c/4$, for all l_j, sufficiently large. Then the sequence $\{\Psi_\lambda(\boldsymbol{x}^{l_j})\} \to -\infty$ as $j \to \infty$, which contradicts that $\Psi_\lambda(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x} \in \mathbb{R}^n$. Therefore, $\liminf_{k\in L} t_k = 0$.

If necessary, we can assume, from a subsequence, that $\lim_{k\in L} t_k = 0$. Hence, a full stepsize never is accepted for all k $k \in L$, sufficiently large. From Armijo-rule (15), we obtain $\Psi_\lambda(\boldsymbol{x}^k + \lambda^{m_k-1}\boldsymbol{d}^k) > \Psi_\lambda(\boldsymbol{x}^k) - \sigma\theta^{m_k-1}\|\boldsymbol{d}^k\|^2$ or, equivalently,

$$\frac{\Psi_\lambda(\boldsymbol{x}^k + \lambda^{m_k-1}\boldsymbol{d}^k) - \Psi_\lambda(\boldsymbol{x}^k)}{\theta^{m_k-1}} \,>\, -\sigma\|\boldsymbol{d}^k\|^2. \tag{30}$$

By taking the limit $k \to \infty$ on L, we obtain from (30), the continuous differentiability of Ψ_λ, $\boldsymbol{d}^k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ for all $k \in L$ and the fact that $\theta^{m_k-1} \to 0$ for $k \to \infty$, in L,

$$-\nabla\Psi_\lambda(\boldsymbol{x}^*)^T\nabla\Psi_\lambda(\boldsymbol{x}^*) \,\geq\, -\sigma\nabla\Psi_\lambda(\boldsymbol{x}^*)^T\nabla\Psi_\lambda(\boldsymbol{x}^*).$$

This implies that $\sigma \geq 1$, which is clearly a contradiction with the initial choice of parameter σ. Therefore,

$$\nabla\Psi_\lambda(\boldsymbol{x}^*) \,=\, 0. \qquad \boxtimes$$

The following lemmas are useful results for the proof of the main global convergence theorem.

Lemma 5.9. *Let $\{\boldsymbol{x}^k\}$ be a sequence generated by* Algorithm 1 *and let $\{\boldsymbol{x}^k\}_L$ be a subsequence converging to $\boldsymbol{x}^* \in \mathbb{R}^n$. If $\boldsymbol{d}^k$ is a Newton direction for all $k \in L$, and the set K is finite, then the sequence $\{\|\boldsymbol{d}^k\|\}_L$ is bounded, that is, there exist positive constants m and M such that*

$$0 \,<\, m \,\leq\, \|\boldsymbol{d}^k\| \,\leq\, M, \qquad \forall\, k \in L. \tag{31}$$

Proof. Let assume that $\boldsymbol{d}^k$ is a Newton direction for all $k \in L$, Thus, for these indices, we have that

$$\|\Phi_\lambda(\boldsymbol{x}^k)\| = \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\,\boldsymbol{d}^k\| \leq \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\|\|\boldsymbol{d}^k\|. \tag{32}$$

If $\{\|\boldsymbol{d}^k\|\}_{\hat{L}} \to 0$ on a subset $\hat{L} \subseteq L$ then, from (32), $\{\|\Phi_\lambda(\boldsymbol{x}^k)\|\}_{\hat{L}} \to 0$ because $\{\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\|\}_{\hat{L}}$ is bounded on the convergent sequence $\{\boldsymbol{x}^k\}_{\hat{L}}$. The continuity of Φ_λ

would imply that $\Phi_\lambda(\boldsymbol{x}^*) = \mathbf{0}$ and, from Lemma 5.1, K would be infinite, contradicting that K is finite. On the other hand, from (12), for all $k \in L$,

$$-\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\|\|\boldsymbol{d}^k\| \leq \Phi_\lambda(\boldsymbol{x}^k)^T\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)\boldsymbol{d}^k \leq -\rho\|\boldsymbol{d}^k\|^p. \tag{33}$$

Since $\{\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\}_L$ is convergent (either by Lemma 3.14 or because μ_k is constant, for k sufficiently large) and therefore bounded, there exists a positive constant C such that $\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\| \leq C$ for all $k \in L$. From this and (33), we have $\rho\|\boldsymbol{d}^k\|^p \leq \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\|\|\boldsymbol{d}^k\| \leq C\|\boldsymbol{d}^k\|$, for all $k \in L$. Since $p > 1$, $\{\|\boldsymbol{d}^k\|\}_L$ is bounded, which guarantees (31). ☑

Lemma 5.10 ([30]). *Let $\{\boldsymbol{x}^k\}$ be generated by* Algorithm 1 *and $\{\boldsymbol{x}^k\}_L$ a subsequence converging to $\boldsymbol{x}^* \in \mathbb{R}^n$. If $\boldsymbol{d}^k$ is a Newton direction for $k \in L$, and K is finite, then* $\liminf_{k\in L} t_k = 0$.

Proof. The details of the proof are given in [30]. ☑

The next theorem is the main convergence result of the proposed algorithm.

Theorem 5.11. *Let $\{\boldsymbol{x}^k\}$ a sequence generated by* Algorithm 1. *Then each accumulation point of $\{\boldsymbol{x}^k\}$ is a stationary point of Ψ_λ.*

Proof. If K is infinite, Proposition 5.7 guarantees the conclusion of this theorem. If K is finite and $\hat{k}$ denote its largest index then (25) to (27) are satisfied for all $k \geq \hat{k}$. On the other hand, let $\boldsymbol{x}^*$ be an accumulation point of $\{\boldsymbol{x}^k\}$. There exists a subsequence $\{\boldsymbol{x}^k\}_L$ of $\{\boldsymbol{x}^k\}$ converging to $\boldsymbol{x}^*$. If $\boldsymbol{d}^k = -\nabla\Psi_\lambda(\boldsymbol{x}^k)$ for a finite number of index $k \in L$, Proposition 5.8 guarantees that $\boldsymbol{x}^*$ is a stationary point of Ψ_λ. Without loss of generality, we assume that $\boldsymbol{d}^k$ is a Newton direction for all $k \in L$. Since K is finite, we can assume that for all $k \in L$, it is satisfied that $k \notin K$, thus the updating rules (17) and (18) do not apply to these indices.

We proceed by contradiction and assume that $\boldsymbol{x}^*$ is not a stationary point of the function Ψ_λ. That is, $\Psi_\lambda(\boldsymbol{x}^*) \neq 0$. By Lemma 5.10, $\liminf_{k\in L} t_k = 0$. Let L_0 be a subsequence of L such that $\{t_k\}_{L_0}$ converges to zero. Then, $m_k > 0$ for all $k \in L_0$, sufficiently large, where $m_k \in \mathbb{N}$ is the exponent in (14). Then, $-2\,\sigma\,\theta^{m_k-1}\,\Psi_\lambda(\boldsymbol{x}^k) < \Psi_{\lambda\mu_k}(\boldsymbol{x}^k + \theta^{m_k-1}\boldsymbol{d}^k) - \Psi_{\lambda\mu_k}(\boldsymbol{x}^k)$ for all $k \in L_0$, sufficiently large. Dividing both inequalities by θ^{m_k-1}, we obtain

$$-2\,\sigma\,\Psi_\lambda(\boldsymbol{x}^k) < \frac{\Psi_{\lambda\mu_k}(\boldsymbol{x}^k + \theta^{m_k-1}\boldsymbol{d}^k) - \Psi_{\lambda\mu_k}(\boldsymbol{x}^k)}{\theta^{m_k-1}}.$$

Let μ_* be the limit of $\{\mu_k\}$ and if $\mu_* = 0$, we denote $\nabla\Psi_{\lambda\mu_*}(\boldsymbol{x}^*)$ for the gradient of the unperturbed function Ψ_λ in $\boldsymbol{x}^*$. From (31), we can assume (through a subsequence) that $\{\boldsymbol{d}^k\}_{L_0} \to \boldsymbol{d}^* \neq 0$. By taking the limit $k \to \infty$, we obtain

$$-2\,\sigma\,\Psi_\lambda(\boldsymbol{x}^*) \leq \nabla\Psi_{\lambda\mu_*}(\boldsymbol{x}^*)^T\boldsymbol{d}^*. \tag{34}$$

For $\mu_* = 0$, this follows from Lemma 3.15. If $\mu_* > 0$, then $\mu_k = \mu_*$ for k sufficiently large, then (34) follows from the Mean Value Theorem. Using (11), (27) and

Proposition 2.2, we have that, for all $k \in L_0$, $\nabla\Psi_{\lambda\mu_k}(\boldsymbol{x}^k)^T\boldsymbol{d}^k \leq -2(1-\sigma)\Psi_\lambda(\boldsymbol{x}^k) + \kappa\|\Phi_\lambda(\boldsymbol{x}^k)\|(\sqrt{\mu_{k-1}} - \sqrt{\mu_k})$; passing to the limit $k \to \infty$, $k \in L_0$, from (34) we obtain (and from Lemma 3.14, if $\mu_* = 0$),

$$-2\sigma\Psi_\lambda(\boldsymbol{x}^*) \leq \nabla\Psi_{\lambda\mu_*}(\boldsymbol{x}^*)^T\boldsymbol{d}^* \leq -2(1-\alpha)\Psi_\lambda(\boldsymbol{x}^*). \tag{35}$$

From Proposition 5.6, $\{\Psi_\lambda(\boldsymbol{x}^k)\}$ is bounded. Moreover, $(\sqrt{\mu_{k-1}} - \sqrt{\mu_k}) \to 0$, since $\{\mu_k\}$ converges. We have that $\Psi_\lambda(\boldsymbol{x}^*) > 0$, (in another case, K would be infinite). Therefore, from (35), $\sigma \geq (1-\alpha)$, which contradicts that $\sigma < (1-\alpha)$. This completes the proof. ☑

6. Local convergence

In this section, we prove under certain hypotheses that the algorithm proposed converges locally and q-superlinearly or q-quadratically. The first result gives a sufficient condition for the convergence of a sequence $\{\boldsymbol{x}^k\}$ generated by Algorithm 1.

Theorem 6.1. *If one of the accumulation points of* $\{\boldsymbol{x}^k\}$, *let us say* $\boldsymbol{x}^*$, *is an isolated solution of the* NCP, *then* $\{\boldsymbol{x}^k\}$ *converges to* $\boldsymbol{x}^*$.

Proof. From Lemma 5.1, the index set K defined by (19) is infinite and $\{\mu_k\}$ converges to zero. Therefore, Proposition 5.7 guarantees that each accumulation point of $\{\boldsymbol{x}^k\}$ is also a solution of the NCP. Thus, $\boldsymbol{x}^*$ must be an isolated point of $\{\boldsymbol{x}^k\}$. Let assume that $\{\boldsymbol{x}^k\}_L$ is an arbitrary subsequence of $\{\boldsymbol{x}^k\}$ converging to $\boldsymbol{x}^*$. Using the updating rule of **S3** in Algorithm 1, we have

$$\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| = \theta^{m_k}\|\boldsymbol{d}^k\| \leq \|\boldsymbol{d}^k\|. \tag{36}$$

From (36), it is enough to prove that $\{\boldsymbol{d}^k\}_L \to 0$. We have that

$$\{\nabla\Psi_\lambda(\boldsymbol{x}^k)\}_L \to \nabla\Psi_\lambda(\boldsymbol{x}^*) = 0 \tag{37}$$

because Ψ_λ is continuous differentiable and the solution $\boldsymbol{x}^*$ is a stationary point of Ψ_λ. If $\{\boldsymbol{d}^k\}_L$ has a finite number of Newton directions then it converges to zero. Because of this, let us assume that there exists a subsequence $\{\boldsymbol{d}^k\}_{L_0}$ of $\{\boldsymbol{d}^k\}_L$ such that $\boldsymbol{d}^k$ is a solution of (11) for all $k \in L_0$. From (13), $\rho\|\boldsymbol{d}^k\|^p \leq -\left(\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\right)^T\boldsymbol{d}^k \leq \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\|\|\boldsymbol{d}^k\|$, for all $k \in L_0$, thus,

$$\|\boldsymbol{d}^k\| \leq \left(\rho^{-1}\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k)\|\right)^{\frac{1}{p-1}}, \tag{38}$$

since $p > 1$. Since $\{\boldsymbol{x}^k\} \to \boldsymbol{x}^*$ and $\{\mu_k\} \to 0$, with $k \in L_0$, the Lemma 3.14 allows to conclude that,

$$\lim_{k\to\infty,\ k\in L_0} \Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^T\Phi_\lambda(\boldsymbol{x}^k) = \nabla\Psi_\lambda(\boldsymbol{x}^*) = 0,$$

and this implies that the right side of (38) converges to zero; therefore, $\{\boldsymbol{d}^k\}_{L_0} \to 0$. From (38), we have that $\{\boldsymbol{d}^k\}_{L\setminus L_0} \to 0$, if $L \setminus L_0$ is infinite. Thus, from (36), we have that $\{\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|\}_L \to 0$, and therefore, from Proposition 3.8, $\{\boldsymbol{x}^k\}$ converges to $\boldsymbol{x}^*$. ☑

The two following results are technical lemmas that we will use in the proof of Theorem 6.6. The first one guarantees that, for all $k \in K$, the matrices $\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)$ are nonsingular and its inverses are bounded. The second guarantees that the Newton direction satisfies the descent condition (12), for all $k \in K$, with k sufficiently large. The details of the proof of each lemma are given in [30].

Lemma 6.2 ([30]). *Let $\{\boldsymbol{x}^k\}$ be a sequence generated by* Algorithm 1. *If one of the limit points, lets us say, $\boldsymbol{x}^*$, is a R-regular solution of the* NCP *then for all $k \in K$; sufficiently large, the matrices $\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)$ are nonsingular and its inverses satisfies that $\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^{-1}\| \leq 2c$, for some positive constant c.*

Lemma 6.3 ([30]). *Under the hypotheses of* Lemma 6.2, *the solution of linear system* (11) *satisfies the descent condition* (12) *for all $k \in K$ sufficiently large.*

The following result will be useful to verify that Algorithm 1 eventually takes the full stepsize $t_k = 1$. Its proof is the same of Lemma 3.2 en [9].

Lemma 6.4 ([30]). *If there exists a scalar $\omega \in \left[\frac{1}{2} - \frac{(1-\alpha-2\sigma)^2}{2(2+\alpha)^2}, \frac{1}{2}\right]$ such that $\Psi_\lambda(\boldsymbol{y}) \leq \Psi_\lambda(\boldsymbol{x}^k) - 2\omega\Psi_\lambda(\boldsymbol{x}^k)$, for some $k \in K$ and $\boldsymbol{y} \in \mathbb{R}^n$, then $\Psi_{\lambda\mu_k}(\boldsymbol{y}) \leq \Psi_{\lambda\mu_k}(\boldsymbol{x}^k) - 2\sigma\Psi_\lambda(\boldsymbol{x}^k)$, where μ_k is the smoothing parameter in the k-th step of* Algorithm 1.

The next lemma guarantees that the indices of the iterations $\boldsymbol{x}^k$ remain in K. By repeating this argument, we will guarantee that $k \in K$ and $t_k = 1$ for all $k \in \mathbb{N}$, sufficiently large.

Lemma 6.5. *Assume the hypotheses of* Lemma 6.2. *There exists an index $\hat{k} \in K$ such that for all $k \geq \hat{k}$, the index $k+1$ also remain in K and $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \boldsymbol{d}^k$.*

Proof. By Lemma 6.2, there is $c > 0$ such that $\|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^{-1}\| \leq 2c$, for all $k \in K$ sufficiently large. For this k, from the Algorithm 1, we have

$$\begin{aligned}
\|\boldsymbol{x}^k + \boldsymbol{d}^k - \boldsymbol{x}^*\| &= \|\boldsymbol{x}^k - \boldsymbol{x}^* - \Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^{-1}\Phi_\lambda(\boldsymbol{x}^k)\| \\
&\leq \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^{-1}\|(\|(\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k)^{-1} - H_k)(\boldsymbol{x}^k - \boldsymbol{x}^*)\| \\
&\qquad + \|H_k(\boldsymbol{x}^k - \boldsymbol{x}^*) - \Phi_\lambda(\boldsymbol{x}^k) + \Phi_\lambda(\boldsymbol{x}^*)\|) \\
&\leq 2c(\gamma\beta_k\|\boldsymbol{x}^k - \boldsymbol{x}^*\| + \|H_k(\boldsymbol{x}^k - \boldsymbol{x}^*) - \Phi_\lambda(\boldsymbol{x}^k) + \Phi_\lambda(\boldsymbol{x}^*)\|).
\end{aligned} \tag{39}$$

Here, $H_k \in \partial_C\Phi_\lambda(\boldsymbol{x}^k)$ is such that $dist_F(\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k), \partial_C\Phi_\lambda(\boldsymbol{x}^k)) = \|\Phi'_{\lambda\mu_k}(\boldsymbol{x}^k) - H_k\|_F \leq \gamma\,\beta_k$, where the inequality is obtained by the part b) of the Proposition 4.1. Moreover, using Proposition 3.6 and since $\beta_k \to 0$, we have

$$\|\boldsymbol{x}^k + \boldsymbol{d}^k - \boldsymbol{x}^*\| = o(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|), \tag{40}$$

with $k \to \infty$ for $k \in K$. From this and Proposition 3.9 we have

$$\|\Phi_\lambda(\boldsymbol{x}^k + \boldsymbol{d}^k)\| = o(\|\Phi_\lambda(\boldsymbol{x}^k)\|), \tag{41}$$

with $k \to \infty$. for $k \in K$. Let $\omega = \max\left\{\frac{1}{2} - \frac{(1-\alpha-2\sigma)^2}{2(2+\alpha)^2}, \frac{(1-\eta)^2}{2}\right\}$, where α, η and σ are the constants defined by Algorithm 1. From (41), there exists an index $\hat{k} \in K$ such that

$$\Psi_\lambda(\boldsymbol{x}^k + \boldsymbol{d}^k) \leq \Psi_\lambda(\boldsymbol{x}^k) - 2\,\omega\,\Psi_\lambda(\boldsymbol{x}^k), \tag{42}$$

for all $k \in K$ with $k \geq \hat{k}$. Then, from Lemma 6.4, $\Psi_{\lambda\mu_k}(\boldsymbol{x}^k + \boldsymbol{d}^k) \leq \Psi_{\lambda\mu_k}(\boldsymbol{x}^k) - 2\,\sigma\,\Psi_\lambda(\boldsymbol{x}^k)$. Thus, full step is accepted for all $k \geq \hat{k},\ k \in K$. In particular, $\boldsymbol{x}^{\hat{k}+1} = \boldsymbol{x}^{\hat{k}} + \boldsymbol{d}^{\hat{k}}$. From (42) and the definition of ω, we have $\|\Phi_\lambda(\boldsymbol{x}^{\hat{k}+1})\| \leq \sqrt{1-2\omega}\|\Phi_\lambda(\boldsymbol{x}^{\hat{k}})\| \leq \eta\,\|\Phi_\lambda(\boldsymbol{x}^{\hat{k}})\| = \eta\beta_{\hat{k}}$, which implies $\hat{k}+1 \in K$. By repeating this process, we can prove that, for all $k \geq \hat{k}$, itholds $k \in K$ and $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \boldsymbol{d}^k$. ☑

The next result gives sufficient conditions for the rate of convergence of Algorithm 1.

Theorem 6.6. *Let* $\{\boldsymbol{x}^k\}$ *be generated by* Algorithm 1. *If one of its limit points, let us say* $\boldsymbol{x}^*$, *is a R-regular solution of the* NCP, *then* $\{\boldsymbol{x}^k\}$ *converges to* $\boldsymbol{x}^*$ *at least q-superlinearly to* $\boldsymbol{x}^*$. *If* F *is continuous differentiable with Jacobian matrix locally Lipschitz, the convergence is q-quadratic.*

Proof. Lemma 6.5 guarantees that $k \in K$ and $t_k = 1$, for all $k \in \mathbb{N}$ sufficiently large. Then the q-superlinear convergence is obtained from (40). If F is continuous differentiable with Jacobian matrix locally Lipschitz, then by Proposition 3.6, we have $\|H_k(\boldsymbol{x}^k + \boldsymbol{d}^k) - \Phi_\lambda(\boldsymbol{x}^k) + \Phi_\lambda(\boldsymbol{x}^*)\| = O(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2)$. Since Φ_λ is locally Lipschitz, $\beta_k = \|\Phi_\lambda(\boldsymbol{x}^k)\| = \|\Phi_\lambda(\boldsymbol{x}^k) - \Phi_\lambda(\boldsymbol{x}^*)\| = O(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|)$. Using these two inequalities in (39), $\|\boldsymbol{x}^k + \boldsymbol{d}^k - \boldsymbol{x}^*\| = O(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2)$; therefore, $\{\boldsymbol{x}^k\}$ converges q-quadratically to $\boldsymbol{x}^*$. ☑

7. Numerical results

In this section, we analyze the global numerical performance of Algorithm 1 and compare it with three global methods for solving the NCP. The first, a nonsmooth quasi-newton method proposed in [5] which we call Algorithm 2. The second, a smoothing Jacobian method proposed in [18] which, unlike our proposal, uses a smoothing of the Fischer function (Algorithm 1 with $\lambda = 2$), which Algorithm 3 and, the third, a smooth Newton method proposed recently in [32], which we call Algorithm 4. We vary λ in two forms obtaining two versions of our algorithm, namely, Method 1: we use the dynamic choice of λ used in [5], (this strategy combines the efficiency of Fisher function far from the solution with that of the minimum function near to it), Method 2: we vary randomly λ in the interval $(0,\ 4)$. We finalized the section with a local analysis of our algorithmic proposal.

For the parameters, we use the following values: $\rho = 10^{-18}$, $p = 2.1$, $\theta = 0.5$, $\sigma = 10^{-4}$, $\gamma = 30$, $\alpha = 0.95$, $\eta = 0.9, \epsilon_1 = 10^{-12}$ $\epsilon_2 = 10^{-6}$, $k_{\max} = 300$, $t_{\min} = 10^{-16}$ (minimum stepsize in linear search), where ϵ_1, ϵ_2 are the tolerances for $\Phi_\lambda(\boldsymbol{x}^k)$ and $\|\nabla\Psi_\lambda(\boldsymbol{x}^k)\|$, respectively [18].

For the numerical test, we consider nine complementarity problems associated with the functions *Kojima-Shindo* (*Koj-Shi*), *Kojima-Josephy* (*Koj-Jo*), *Mathiesen modificado* (*Math mod*), *Mathiesen* (*Mathiesen*) *Billups* (*Billups*) [7], [25]; *Nash-Cournot* (*Nash-Co*) [16], *Hock-Schittkowski* (*HH 66*) [32], *Geiger-Kanzow* (*Geiger-Kanzow*) [15], *Ahn* (*Ahn*) [2]. We implemented Algorithms 1 (with Methods 1 and 2) and the test functions in MATLAB and use the following starting points taking from [5], [32],

$\boldsymbol{x}_1 = (1\ \ 1\ \ 1\ \ 1)^T$
$\boldsymbol{x}_2 = 1$
$\boldsymbol{x}_3 = (10\ \ 10\ \ 10\ \ 10\ \ 10)^T$
$\boldsymbol{x}_4 = (10\ \ 10\ \ \cdots\ \ 10)^T$
$\boldsymbol{x}_5 = 10\,\boldsymbol{x}_1$
$\boldsymbol{x}_6 = (-1\ \cdots\ -1)^T$
$\boldsymbol{x}_7 = 6\,\boldsymbol{x}_1$
$\boldsymbol{x}_8 = (1\ \ 2\ \ 3\ \ 4)^T$
$\boldsymbol{x}_9 = (2\ \ -3\ \ -3\ \ 2)^T$
$\boldsymbol{x}_{10} = 100\,\boldsymbol{x}_1$
$\boldsymbol{x}_{11} = (1\ \ 0\ \ 1\ \ 0)^T$
$\boldsymbol{x}_{12} = (1\ \ 0\ \ 0\ \ 0)^T$
$\boldsymbol{x}_{13} = (1\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0)^T$
$\boldsymbol{x}_{14} = 0$
$\boldsymbol{x}_{15} = (1\ \ 1\ \ 1\ \ 1\ \ 1)^T$
$\boldsymbol{x}_{16} = (100\ \cdots\ 100)^T$
$\boldsymbol{x}_{17} = -2\,\boldsymbol{x}_1$
$\boldsymbol{x}_{18} = (1\ \ 4\ \ 4\ \ 1)^T$
$\boldsymbol{x}_{19} = 3\,\boldsymbol{x}_1$
$\boldsymbol{x}_{20} = (0\ \ \cdots\ \ 0)^T$
$\boldsymbol{x}_{21} = (-1\ \cdots\ -1\ \ 1\cdots 1)^T.$

To analyze the global convergence of Algorithm 1 and the variation of λ, we generate 100 random initial vectors for each problem with each of the methods described previously. We present the results in Table 1, which contains the problem and the method used to choose the parameter λ; the execution average time in seconds (t); the average number of iterations (k), and the percentage of times that the algorithm finds a solution to the problem (*Success* (%)).

Problem	**Method**	k	t (*sec*)	*Success* (%)
Koj-Shi	**1**	14.87	0.0141	99
	2	12.85	0.0055	97
Koj-Jo	**1**	14.66	0.0170	100
	2	24.06	0.0101	100
Math mod	**1**	9.5135	0.0123	74
	2	8.8028	0.0066	71
Billups	**1**	17.65	0.0093	100
	2	38.65	0.0040	61
Nash-Co (*5*)	**1**	8.09	0.0148	100
	2	8.36	0.0096	100
Nash-Co (*10*)	**1**	12.02	0.0325	100
	2	12.71	0.0398	100
Mathiesen	**1**	10.91	0.0129	99
	2	10.91	0.0056	100
HH 66	**1**	10.54	0.0128	100
	2	12.1	0.0096	100
Geiger-Kanzow	**1**	6	1.342	100
	2	6.61	1.179	100

Table 1. Algorithm 1 *varying* λ *with random starting points.*

The results of Table 1 show that the number of iterations with Methods 1 and 2 are similar, except for two of the problems for which Method 2 increases them. In average time, except for one problem, Method 2 always is better (even with the problems where there are more iterations). In *Succes(%)*, they are similar, except in the case of Billups Problem, where it decreases by 31 % with Method 2. Thus, for this set of numerical test, Method 1, generally used to choose λ in nonsmooth Newton-type methods for NCP, is not better than a random choice of this parameter (Method 2), which indicates that it would be convenient to find an alternative to choosing λ.

Now, we compare Algorithm 1 with Algorithms 2, 3, and 4. For this comparison, first, we consider Algorithm 1 with Method 1 versus the other three algorithms. Afterward,

we consider Algorithm 1 with Method 2. We do numerical tests using the fixed starting points described previously. In addition, we take $\lambda = 2$ in our algorithm to obtain Algorithm 3 [18]. We measure the average of iterations and the average algorithm execution time. We present the results in Table 2.

To compare with Algorithm 2, we consider the results associate with *Koj-Jo*, *Koj-Shi*, *Math mod,* and *Billups* problems, which were also used in [5] to analyze the performance of its algorithm (Tabla 4.2 in [5]). Algorithm 1, with Methods 1 and 2, converges for some starting points for which Algorithm 2 does not converge. For example, with Method 1, the *Koj-Jo* problem with $\boldsymbol{x}_{10}$ and the *Billups* problem with x_{14}, and with Method 2, the *Koj-Jo* problem with $\boldsymbol{x}_{20}$. To compare with Algorithm 3, we consider the results in Table 2 for Methods 1 and 2 versus those in the same table for $\lambda = 2$. We observe that with Method 1, the average number of iterations and the execution time of Algorithms 1 and 3 are similar. Now, Algorithm 3 compared to Algorithm 1 using Method 2 presents better performance for some problems in the average number of iterations, (*Mathiesen* and *Geiger-Kanzow* problems). On the other hand, the execution time with these two methods is very similar, except for 3 problems (*Mathiesen Mod*, *Billups*, *H66*), where it is less using Method 2. Thus, being able to choose λ brings advantages.

Problem	**Method**	$\boldsymbol{x}_0$	$\bar{t}$ (sec)	k
Koj-Shi	$\lambda = 2$	$\boldsymbol{x}_7$	0.0090	14.00
Koj-Shi	**1**	$\boldsymbol{x}_7$	0.0064	11.00
Koj-Shi	**2**	$\boldsymbol{x}_7$	0.0062	11.82
Koj-Shi	$\lambda = 2$	$\boldsymbol{x}_8$	0.0062	11.00
Koj-Shi	**1**	$\boldsymbol{x}_8$	0.0069	12.00
Koj-Shi	**2**	$\boldsymbol{x}_8$	0.0081	13.92
Koj-Shi	$\lambda = 2$	$\boldsymbol{x}_9$	0.0057	10.00
Koj-Shi	**1**	$\boldsymbol{x}_9$	0.0062	10.00
Koj-Shi	**2**	$\boldsymbol{x}_9$	0.0412	30.56
Koj-Jo	$\lambda = 2$	$\boldsymbol{x}_{10}$	0.0354	31.00
Koj-Jo	**1**	$\boldsymbol{x}_{10}$	0.0360	31.00
Koj-Jo	**2**	$\boldsymbol{x}_{10}$	0.0381	32.58
Koj-Jo	$\lambda = 2$	$\boldsymbol{x}_{11}$	0.0025	6.00
Koj-Jo	**1**	$\boldsymbol{x}_{11}$	0.0027	6.00
Koj-Jo	**2**	$\boldsymbol{x}_{11}$	0.0026	6.08
Koj-Jo	$\lambda = 2$	$\boldsymbol{x}_{12}$	0.0041	10.00
Koj-Jo	**1**	$\boldsymbol{x}_{12}$	0.0046	10.00
Koj-Jo	**2**	$\boldsymbol{x}_{12}$	0.0044	10.31
Math mod (4)	$\lambda = 2$	$\boldsymbol{x}_{10}$	0.0129	9.00
Math mod (4)	**1**	$\boldsymbol{x}_{10}$	0.0130	9.00
Math mod (4)	**2**	$\boldsymbol{x}_{10}$	0.0094	10.68
Math mod (4)	$\lambda = 2$	$\boldsymbol{x}_1$	0.0017	4.00
Math mod (4)	**1**	$\boldsymbol{x}_1$	0.0018	4.00
Math mod (4)	**2**	$\boldsymbol{x}_1$	0.0016	3.84
Math mod (4)	$\lambda = 2$	$\boldsymbol{x}_{11}$	0.0017	4.00
Math mod (4)	**1**	$\boldsymbol{x}_{11}$	0.0015	3.00

Problem	**Method**	$\boldsymbol{x}_0$	$\bar{t}$ (sec)	k
Math mod (4)	**2**	$\boldsymbol{x}_{11}$	0.0015	3.64
Billups	$\lambda = 2$	$\boldsymbol{x}_{14}$	0.0092	20.00
Billups	**1**	$\boldsymbol{x}_{14}$	0.0097	19.00
Billups	**2**	$\boldsymbol{x}_{14}$	0.0065	16.62
Billups	$\lambda = 2$	$\boldsymbol{x}_{2}$	0.0005	4.00
Billups	**1**	$\boldsymbol{x}_{2}$	0.0008	5.00
Billups	**2**	$\boldsymbol{x}_{2}$	0.0005	3.99
Nash-Co (5)	$\lambda = 2$	$\boldsymbol{x}_{15}$	0.0097	8.00
Nash-Co (5)	**1**	$\boldsymbol{x}_{15}$	0.0103	8.00
Nash-Co (5)	**2**	$\boldsymbol{x}_{15}$	0.0095	8.50
Nash-Co (5)	$\lambda = 2$	$\boldsymbol{x}_{3}$	0.0069	6.00
Nash-Co (5)	**1**	$\boldsymbol{x}_{3}$	0.0081	6.00
Nash-Co (5)	**2**	$\boldsymbol{x}_{3}$	0.0070	6.07
Nash-Co (5)	$\lambda = 2$	$\boldsymbol{x}_{16}$	0.0117	9.00
Nash-Co (5)	**1**	$\boldsymbol{x}_{16}$	0.0131	9.00
Nash-Co (5)	**2**	$\boldsymbol{x}_{16}$	0.0118	9.24
Nash-Co (10)	$\lambda = 2$	$\boldsymbol{x}_{17}$	0.0120	10.00
Nash-Co (10)	**1**	$\boldsymbol{x}_{17}$	0.0117	7.00
Nash-Co (10)	**2**	$\boldsymbol{x}_{17}$	0.0109	7.82
Nash-Co (10)	$\lambda = 2$	$\boldsymbol{x}_{18}$	0.0181	11.00
Nash-Co (10)	**1**	$\boldsymbol{x}_{18}$	0.0189	10.00
Nash Co (10)	**2**	$\boldsymbol{x}_{18}$	0.0194	11.31
Nash-Co (10)	$\lambda = 2$	$\boldsymbol{x}_{19}$	0.0981	7.00
Nash-Co (10)	**1**	$\boldsymbol{x}_{19}$	0.0907	7.00
Nash-Co (10)	**2**	$\boldsymbol{x}_{19}$	0.1100	8.90
Mathiesen	$\lambda = 2$	$\boldsymbol{x}_{6}$	0.0034	8.00
Mathiesen	**1**	$\boldsymbol{x}_{6}$	0.0036	8.00
Mathiesen	**2**	$\boldsymbol{x}_{6}$	0.0188	19.50
Mathiesen	$\lambda = 2$	$\boldsymbol{x}_{20}$	0.0021	5.00
Mathiesen	**1**	$\boldsymbol{x}_{20}$	0.0020	4.00
Mathiesen	**2**	$\boldsymbol{x}_{20}$	0.0021	4.67
Mathiesen	$\lambda = 2$	$\boldsymbol{x}_{21}$	0.0115	11.00
Mathiesen	**1**	$\boldsymbol{x}_{21}$	–	–
Mathiesen	**2**	$\boldsymbol{x}_{21}$	0.0355	21.18
HH 66	$\lambda = 2$	$\boldsymbol{x}_{6}$	0.0077	9.00
HH 66	**1**	$\boldsymbol{x}_{6}$	0.0063	8.00
HH 66	**2**	$\boldsymbol{x}_{6}$	0.0062	9.53
HH 66	$\lambda = 2$	$\boldsymbol{x}_{4}$	0.0053	8.00
HH 66	**1**	$\boldsymbol{x}_{4}$	0.0049	7.00
HH 66	**2**	$\boldsymbol{x}_{4}$	0.0056	8.89
HH 66	$\lambda = 2$	$\boldsymbol{x}_{16}$	0.0045	8.00
HH 66	**1**	$\boldsymbol{x}_{16}$	0.0042	7.00
HH 66	**2**	$\boldsymbol{x}_{16}$	0.0049	8.80
Geiger-Kanzow	$\lambda = 2$	$\boldsymbol{x}_{6}$	1.2476	5.00

Problem	**Method**	$\boldsymbol{x}_0$	$\bar{t}$ (sec)	k
Geiger-Kanzow	**1**	$\boldsymbol{x}_6$	1.0896	4.00
Geiger-Kanzow	**2**	$\boldsymbol{x}_6$	2.8282	13.97
Geiger-Kanzow	$\lambda = 2$	$\boldsymbol{x}_4$	1.1148	5.00
Geiger-Kanzow	**1**	$\boldsymbol{x}_4$	0.8795	4.00
Geiger-Kanzow	**2**	$\boldsymbol{x}_4$	4.2205	15.03

Table 2: Comparison of the Algorithms 1 and 3.

Taking into account the results for *Koj-Shi*, *Mathiesen*, *HH 66*, and *Geiger-Kanzow* problems, reported in Table 2 for Algorithm 1 and in Tables 1 to 4 in [32] for Algorithm 4, we observe that in all cases, our algorithm with Methods 1 finds a solution in fewer iterations. However, with Method 2, it generally exceeds the number of iterations reported in [32].

Finally, we analyze the local performance of Algorithm 1. In Section 6, we prove, under certain hypotheses, that Algorithm 1 converges superlinear and even quadratically, which is desirable for an iterative method. To illustrate this types of convergence we calculate the quotients

$$R1_k = \frac{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^k - \boldsymbol{x}^*\|} \qquad \text{and} \qquad R2_k = \frac{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2},$$

which are related to the definitions of superlinear and quadratic convergence of a vector sequence, respectively. In Table 3, we present the results for *Billups* problem (for more examples, see [30]). In this Table, $R1_k$, clearly, converges to zero, which means that Algorithm 1 converges at least superlinearly, and $R2_k$ seems to be bounded, which means that Algorithm 1 may converge quadratically. This fast convergence of Algorithm 1 is due to (closely related to what was proved in Section 6: near the solution, the Newton step is full ($t_k = 1$). Moreover, this is illustrated in Table **4**, where the column np indicates the full number of steps in the linear search which also correspond to the last steps.

k	$R1_k$	$R2_k$
1	0.8909	0.8494
2	0.1629	0.1744
3	0.0634	0.4163
4	0.0045	0.4713
5	0.0000	0.1238

Table 3. *Convergence rate for* Algorithm 1 *with Billups Problem.*

It is important to mention that in all cases, Algorithm 1 uses the Newton direction which is desirable because it makes its convergence fast. In the 77% of numerical tests, the number of full Newton steps in the monotone linear search (np) is greater than or equal to half the number of iterations ($k/2,$) which explains the fast convergence of our algorithm for this set of tests.

Problema	$\boldsymbol{x}_0$	**Método**	**Dim**	k	**np**
Koj-Shi	$\boldsymbol{x}_1$	**1**	4	7	3
		2	4	7	3
Koj-Jo	$\boldsymbol{x}_1$	**1**	4	6	3
		2	4	6	3
Math mod	$\boldsymbol{x}_1$	**1**	4	4	4
		2	4	4	4
Billups	$\boldsymbol{x}_2$	**1**	1	5	4
		2	1	4	4
Nash Co (5)	$\boldsymbol{x}_3$	**1**	5	6	5
		2	5	7	6
Nash Co (10)	$\boldsymbol{x}_4$	**1**	10	10	3
		2	10	11	4
Mathiesen	$\boldsymbol{x}_5$	**1**	4	6	5
		2	4	8	8
HH 66	$\boldsymbol{x}_6$	**1**	8	8	7
		2	8	9	8
Geiger-Kanzow	$\boldsymbol{x}_6$	**1**	500	4	4
		2	500	5	5

Table 4. *Full Newton steps of Algoritmo 1 varying* λ.

In the other part, we use three indices to collect additional information to compare Algorithm 1, varying λ (Methods 1 and 2) with Algorithm 3 (Algorithm 1 with $\lambda = 2$). These indices are the follows [6]:

- Robustness index:

$$R_j = \frac{t_j}{n_j} \tag{43}$$

- Efficiency index:

$$E_j = \sum_{i=1 \ r_{ij} \neq 0}^{m} \left(\frac{r_{ib}}{r_{ij}} \right) / r_j \tag{44}$$

- Combined robustness and efficiency index:

$$E_j \times R_j = \sum_{i=1, \ r_{ij} \neq 0}^{m} \left(\frac{r_{ib}}{r_{ij}} \right) / n_j, \tag{45}$$

where r_{ij} is the number of iterations required to solve the problem i by the method j, $r_{ib} = \min_j r_{ij}$, t_j is the number of successes by method j and n_j is the number of problems attempted by method j.

For the calculation of the indices, we use the data from Table 4 and Table 7.1 in [30], the latter for $\lambda = 2$. In Table 5, we present the results obtained.

	Método 1	**Método 2**	$\lambda = 2$
t_j	9	9	9
R_j	1	1	1
E_j	0.9777	0.9116	0.9394
$E_j \times R_j$	0.9777	0.9116	0.9394

Table 5. Indices for the Algorithm 1 varying λ.

On this set of problems, the results in Table 5 confirm robustness of Algorithm 1 with the three choices of parameter λ. Also, the algorithm is more efficient with the dynamic choice of λ (Method 1) of the three, although further testing will indicate to what extent this is true of broader classes of problems.

8. Final Remarks

One way to deal with the non-differentiability of the NCP is using the smoothed Jacobian strategy [9], [18], which allows us to approximate the reformulation of the problem by a succession of differentiable nonlinear systems that depend on a positive parameter. In this article, we proposed a global smoothed Jacobian algorithm that generalizes the one proposed in [18] to all members of the family defined in (5). We developed their convergence theory and did numerical tests to analyze their global and local performance. Our proposal presents some advantages in terms of global convergence compared to other methods as those proposed in [5], [18], and [32].

Finally, we believe that it is necessary to study a relationship between the parameter λ and the nonlinear complementarity problem, which may improve the convergence of the algorithm. It would also be interesting to apply the smoothing Jacobian strategy to equation-reformulation of the nonlinear complementarity problem presented in [32].

Acknowledgments. The authors are grateful an anonymous referee for valuable suggestions and comments.

References

[1] Abaffy J., Broyden C. and Spedicato E., "A class of direct methods for linear systems", *Numer. Math.*, 45 (1984), No. 3, 361–376. doi:10.1007/BF01391414.

[2] Ahn B.H., "Iterative methods for linear complementarity problems with upperbounds on primary variables", *Math. Program.*, 26 (1983) No. 3, 295–315. doi:10.1007/BF02591868.

[3] Anitescu M., Cremer J.F. and Potra F.A., "On the existence of solutions to complementarity formulations of contact problems with friction", *Complementary and Variational Problems: State of the Art.*, 92 (1997), 12–21.

[4] Arenas F., Martínez H.J. and Pérez R., "A local Jacobian smoothing method for solving Nonlinear Complementarity Problems", *Universitas Scientiarum*, 25 (2020), No. 1, 149–174. doi: 10.11144/Javeriana.SC25-1.aljs.

[5] Arias C.A., "Un algoritmo cuasi Newton global para problemas de complementariedad no lineal", Thesis (MSc.), Universidad del Cauca, Popayán, 2014, 44 p.

[6] Buhmiler S. and Krejić N., "A new smoothing quasi-Newton method for nonlinear complementarity problems", *J. Comput. Appl. Math.*, 211 (2008), No. 2, 141–155. doi: 10.1016/j.cam.2006.11.007.

[7] Billups S.C., "Algorithms for Complementarity Problems and Generalized Equations", Thesis (Ph.D.), University of Wisconsin, Madison, 1994. 159 p.

[8] Chen A., Oh J.S., Park D. and Recker W., "Solving the bicriteria traffic equilibrium problem with variable demand and nonlinear path costs", *Appl. Math. Comput.*, 217 (2010), No. 7, 3020–3031. doi: 10.1016/j.amc.2010.08.035.

[9] Chen X., Qi L. and Sun D., "Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities", *Math. Comp.*, 67 (1998), No. 222, 519–540. doi: 10.1090/S0025-5718-98-00932-6.

[10] Clarke F.H., "Generalized gradients and applications", *Trans. Amer. Math. Soc.*, 205 (1975), 247–262. doi:10.1090/S0002-9947-1975-0367131-6.

[11] Clarke F.H., *Optimization and Nonsmooth Analysis*, SIAM, 1990.

[12] Dennis J.E. and Schnabel R.B., *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, 1st ed., Philadelphia, 1996.

[13] Facchinei F. and Soares J., "A New Merit Function For Nonlinear Complementarity Problems And A Related Algorithm", *SIAM J. Optim.*, 7 (1997), No. 1, 225–247. doi: 10.1137/S1052623494279110.

[14] Ferris M.C. and Pang J.S., "Engineering and Economic Applications of Complementarity Problems", *SIAM Rev.*, 39 (1997), No. 4, 669–713. doi: 10.1137/S0036144595285963.

[15] Geiger C. and Kanzow C., "On the resolution of monotone complementarity problems", *Comput. Optim. Appl.*, 5 (1996), No. 2, 155–173. doi: 10.1007/BF00249054.

[16] Harker P.T., "Accelerating the convergence of the diagonalization and projection algorithms for finite-dimensional variational inequalities", *Math. Program.*, 41 (1988), No. 1, 29–59. doi: 10.1007/BF01580752.

[17] Kanzow C. and Kleinmichel H., "A New Class of Semismooth Newton-Type Methods for Nonlinear Complementarity Problems", *Comput. Optim. Appl.*, 11 (1998), No. 3, 227–251. doi:10.1023/A:1026424918464.

[18] Kanzow C. and Pieper H., "Jacobian Smoothing Methods for Nonlinear Complementarity Problems", *SIAM J. Optim.*, 9 (1999), No. 2, 342–373. doi: 10.1137/S1052623497328781.

[19] Kostreva M.M., "Elasto-hydrodynamic lubrication: A non-linear complementarity problem", *Internat. J. Numer. Methods Fluids.*, 4 (1984), No. 4, 377–397. doi: 10.1002/fld.1650040407.

[20] Li D.H. and Fukushima M., "Globally Convergent Broyden-Like Methods for Semismooth Equations and Applications to VIP, NCP and MCP", *Ann. Oper. Res.*, 103 (2001), No. 1, 71–97. doi: 10.1023/A:1012996232707.

[21] Lopes V.L., Martínez J.M. and Pérez R., "On the local convergence of quasi-Newton methods for nonlinear complementarity problems", *Appl. Numer. Math.*, 30 (1999), No. 1, 3–22. doi:10.1016/S0168-9274(98)00080-4.

[22] Ma C., "A new smoothing quasi-Newton method for nonlinear complementarity problems", *Appl. Math. Comput.*, 171 (2005), No. 2, 807–823. doi: 10.1016/j.amc.2005.01.088.

[23] Moré J.J. and Sorensen D.C., "Computing a Trust Region Step", *SIAM J. Sci. and Stat. Comput.*, 4 (1983), No. 3, 553–572. doi: 10.1137/0904038.

[24] Nocedal J. and Wright S., *Numerical optimization*, Springer Science & Business Media, 2nd ed., 2006.

[25] Pang J.S. and Qi L., "Nonsmooth Equations: Motivation and Algorithms", *SIAM J. Optim.*, 3 (1993), No. 3, 443–465. doi: 10.1137/0803021.

[26] Pérez R. and Lopes V.L., "Recent Applications and Numerical Implementation of Quasi-Newton Methods for Solving Nonlinear Systems of Equations", *Numer. Algorithms.*, 35 (2004), No. 2, 261–285. doi: 10.1023/B:NUMA.0000021762.83420.40.

[27] Qi L., "Convergence Analysis of Some Algorithms for Solving Nonsmooth Equations", *Math. Oper. Res.*, 18 (1993), No. 1, 227–244. doi: 10.1287/moor.18.1.227.

[28] Qi L., *C-differentiability, C-differential operators and generalized Newton methods*, School of Mathematics University of New South Wales, Sydney, 1996.

[29] Sherman A.H., "On Newton-iterative methods for the solution of systems of nonlinear equations", *SIAM J. Numer. Anal.*, 15 (1978), No. 4, 755–771. doi: 10.1137/0715050.

[30] Sánchez G.W., "Un algoritmo global con jacobiano suavizado para complementariedad no lineal", Thesis (MSc), Universidad del Cauca, 2019.

[31] Schraudolph N.N., Yu J. and Günter S., "A stochastic quasi-Newton method for online convex optimization", Eleventh International Conference on Artificial Intelligence and Statistics, San Juan, Puerto Rico, 2, 436–443, March, 2007.

[32] Zhu J. and Hao B., "A new smoothing method for solving nonlinear complementarity problems", *Open Math.*, 17 (2019), No. 1, 104–119. doi: 10.1515/math-2019-0011.