A numerical algorithm for a Signorini problem associated with Maxwell-Norton materials by using generalized Newton's methods [1]

P. Barral [a], C. Moreno [b], P. Quintela [a] and M. T. Sánchez [a]

[a] *Departamento de Matemática Aplicada. Facultade de Matemáticas. Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain.*

[b] *Departamento de Estadística, Investigación Operativa y Cálculo Numérico. Facultad de Ciencias. Universidad Nacional de Educación a Distancia. Senda del Rey s\n, 28040 Madrid, Spain.*

Abstract

The goal of this work is to present new numerical approaches to obtain computationally efficient adaptive procedures to solve contact problems associated with Maxwell-Norton materials. We will describe in detail the numerical methods used to solve such problems and derive iterative algorithms. To validate the proposed algorithms we will present some test examples with known analytical solutions. Finally, we will compare the efficiency of the algorithms presented in this paper with others previously proposed by the authors in [3,4].

Key words: frictionless contact, Maxwell-Norton materials, nonsmooth equations, Newton's methods, adimensionalization.

1 Introduction

This work is a continuation of the research initiated in our previous articles [2]-[5] of a problem of unilateral contact between a Maxwell-Norton elastic-viscoplastic body and a frictionless rigid foundation arising from an aluminium casting process. We must overcome two main difficulties: the unilateral contact condition and Norton-Hoff's nonlinear law. Various mathematical aspects

[1] This research was supported by Xunta de Galicia (projects PGIDT00PXI20701PR, PGIDT02PXIC20701PN) and CYCIT-FEDER (projects DPI2001-2908, DPI2004-01993).

of this subject have been studied: the detailed formulation as a variational inequality, the identification of the spaces for which this formulation is well defined, the establishment of sufficient conditions for the existence of a solution, the approximation of the variational inequality and the description of a numerical scheme to solve it. An approximate solution of this problem was obtained by using an implicit Euler scheme in time and a finite element method in space. To deal with the nonlinearities, the numerical solution was based on iterative algorithms involving two multipliers which were approximated by using fixed point algorithms (see [3,4,6]). Given the considerable computational demands of the three-dimensional industrial casting problem, in this work we will present the development of various numerical methods in order to obtain computationally efficient adaptive procedures. Two approaches will be employed to achieve this objective:

(1) approximating the contact multiplier using generalized Newton's methods together with a penalization technique to conserve the matrix's symmetry ([14,16,20]), and
(2) approximating the viscoplasticity multiplier with standard Newton's techniques without modification of the stiffness matrix at each iteration.

Several numerical results will be reported to show the applicability of the algorithms to the problem in question. These results will allow us to compare the different methods in terms of error versus the number of iterations and the computing time.

The outline of the paper is as follows; in Section 2, we will show the mathematical model for the mechanical phenomena and will distinguish three submodels: contact (CP), viscoplasticity (VP) and both effects (CVP). To simplify the presentation of this paper we will omit the thermal effects and, thus, will only consider a quasistatic evolution problem of an elastic-viscoplastic body with contact posed on a domain independent of time. In Section 3 we will propose a weak formulation for the aforementioned sub-models. Section 4 describes the numerical procedure to approximate the solution of the (CP) and (VP) problems. Firstly, we will make a space discretization by a finite element method. The contact condition will be treated with the Bermúdez-Moreno algorithm (see [6]) in which the contact multiplier is a fixed point of a nonlinear equation; to approximate this fixed point we will use a generalized Newton method for nonsmooth equations and a penalization technique. Afterwards, we will study the (VP) problem applying an implicit Euler scheme together with maximal monotone operator techniques to the constitutive law. We will solve the nonlinear equation associated with the viscoplasticity multiplier by using a standard Newton method. The resulting numerical algorithm to solve the complete (CVP) system will be detailed in Section 5. To validate the proposed algorithms we will present some test examples with known analytical solutions in Section 6. These examples have been designed to reproduce a be-

haviour analogous to semicontinuous real casting behaviour: large gradients of stresses with respect to time –for which it is necessary to employ an adimensionalization technique, an Armijo rule and an optimization of the time step–, and a gap between the computational domain and the rigid foundation which grows with time. Numerical results will be given for the three sub-models to compare the efficiency of these methodologies with the fixed point technique used in [3,4].

2 Mathematical model

Let Ω be a bounded domain in $\mathbb{R}^n$, $n = 2$ or 3, with a smooth boundary Γ split into three disjointed pieces Γ_C, Γ_D and Γ_N such that the $(n-1)$-dimensional measures of Γ_C and Γ_D are positive. We refer the motion of the body to a fixed system of rectangular Cartesian axes $Ox_i, i = 1, ..., n$. For the sake of simplicity, except whereas otherwise indicated, $n = 3$. Hereafter, Latin subscripts are understood to range over the integers $\{1, 2, 3\}$, and summation over repeated subscripts is implied. From here on out, we will use a superposed dot to denote differentiation with respect to time.

We consider an elastic-viscoplastic homogeneous material which at the initial time occupies the region Ω. The configuration of the body at this time is taken as the reference configuration. Let u_i denote the i component of the displacement vector field; then the components of the infinitesimal strain field are given by

$$\varepsilon_{ij}(\mathbf{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right).$$

The Maxwell-Norton elastic-viscoplastic law can be described in the following manner: we assume that the strain rate tensor in the body, $\dot{\boldsymbol{\varepsilon}}(\mathbf{u})$, may be decomposed into elastic and viscoplastic effects: $\boldsymbol{\varepsilon}(\mathbf{u}) = \boldsymbol{\varepsilon}^e(\mathbf{u}) + \boldsymbol{\varepsilon}^p(\mathbf{u})$. We assume that $\boldsymbol{\varepsilon}^e$ is related to the stress tensor $\boldsymbol{\sigma}$ through a linear elastic law: $\boldsymbol{\varepsilon}^e = \Lambda\boldsymbol{\sigma}$, where Λ is the inverse of the fourth order linear elasticity tensor defined in terms of Young's modulus, E, and Poisson's ratio ν by

$$\Lambda\boldsymbol{\tau} = \frac{1+\nu}{E}\boldsymbol{\tau} - \frac{\nu}{E}\tau_{kk}\mathbf{I}, \ \forall\boldsymbol{\tau} \in \mathbb{R}^9_s, \tag{1}$$

$\mathbb{R}^9_s$ being the set of second order symmetric tensors operating on $\mathbb{R}^3$.

The viscoplastic part of the strain rate field, $\dot{\boldsymbol{\varepsilon}}^p$, is governed by Norton-Hoff's law. To describe this law, let us introduce the dissipation potential Φ_q defined on $\mathbb{R}^9_s$ by

$$\Phi_q(\boldsymbol{\tau}) = \frac{\theta_0}{q} \mid \boldsymbol{\tau} \mid^q,$$

where q is a fixed real number greater than 2 and θ_0 denotes a positive material parameter. Norton-Hoff's law may be written as (see [12])

$$\dot{\boldsymbol{\varepsilon}}^p(\mathbf{u}) = \nabla\Phi_q(\boldsymbol{\sigma}^D) = \theta_0 \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D,$$

where $\boldsymbol{\sigma}^D$ denotes the deviatoric part of $\boldsymbol{\sigma}$ given by

$$\boldsymbol{\sigma}^D = \boldsymbol{\sigma} - \frac{1}{3}\sigma_{kk}\mathbf{I},$$

and $\nabla\Phi_q$ denotes the gradient of Φ_q relative to the inner product $\boldsymbol{\sigma} : \boldsymbol{\tau} = \sigma_{ij}\tau_{ij}$.

Summing up, the constitutive law can be formulated as

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \Lambda\dot{\boldsymbol{\sigma}} + \theta_0 \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D \text{ in } \Omega. \tag{2}$$

The quasistatic equilibrium of the body under consideration is then governed by the equation (2) and by the usual equilibrium equations

$$-\mathrm{Div}(\boldsymbol{\sigma}) = \mathbf{f} \text{ in } \Omega,$$

where $\mathbf{f}$ denotes the external forces acting on the body.

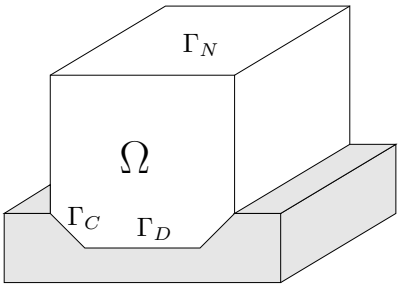

Fig. 1. Computational domain.

The body is clamped on the portion Γ_D and is submitted to traction surface forces of density $\mathbf{g}$ on Γ_N (see Figure 1). Γ_C is the possible contact surface with a rigid barrier; on this boundary, we impose a Signorini frictionless contact condition (several contact conditions with and without friction are detailed in [16]). Finally, we will add initial conditions for $\mathbf{u}$ and $\boldsymbol{\sigma}$. Then, the problem we must solve is the following:

At each time $t \in (0,T]$, find $\mathbf{u}$ and $\boldsymbol{\sigma}$ such that:

$$\begin{cases} -\mathrm{Div}(\boldsymbol{\sigma}) = \mathbf{f} \text{ in } \Omega, \\ \mathbf{u} = \mathbf{0} \text{ on } \Gamma_D, \\ \boldsymbol{\sigma}\mathbf{n} = \mathbf{g} \text{ on } \Gamma_N, \\ \boldsymbol{\sigma}_\tau = \mathbf{0},\ \sigma_n \leq 0,\ u_n \leq 0,\ \sigma_n u_n = 0 \text{ on } \Gamma_C, \\ \dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \Lambda\dot{\boldsymbol{\sigma}} + \theta_0 \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D \text{ in } \Omega, \\ \mathbf{u}(x,0) = \mathbf{u}_0(x), \quad \boldsymbol{\sigma}(x,0) = \boldsymbol{\sigma}_0(x) \text{ in } \Omega, \end{cases} \tag{3}$$

where $(0,T]$ is the time interval of interest; $\mathbf{n}$ is the unit outward normal to the boundary; u_n and $\mathbf{u}_\tau$ (respectively, σ_n and $\boldsymbol{\sigma}_\tau$) must be understood to mean the normal and the tangential components of $\mathbf{u}$ (respectively, of $\boldsymbol{\sigma}\mathbf{n}$) on Γ_C.

We now distinguish three sub-models:

- **(CVP)** corresponding to the aforementioned model.
- **(CP)** corresponding to problem (3) replacing the constitutive law by

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \Lambda\boldsymbol{\sigma}. \tag{4}$$

- **(VP)** corresponding to problem (3) replacing the contact condition on Γ_C by a homogeneous Dirichlet boundary condition.

Remark 1 *If we remove the viscoplastic behaviour from the constitutive equation of the problem (3), we have*

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \Lambda\dot{\boldsymbol{\sigma}}.$$

Nevertheless, if we integrate this equation in time, assuming that $\mathbf{u}_0$, $\boldsymbol{\sigma}_0$ *verify the elastic law at* $t = 0$, *we obtain the constitutive equation (4).*

3 Weak formulation

Let $2 \leq q < +\infty$ be the exponent of viscoplastic law and p, $1 < p \leq 2$, its conjugate exponent. We consider the space of the displacement fields as

$$\mathbf{V}^p = \{\mathbf{v} \in [W^{1,p}(\Omega)]^3;\ \mathrm{Div}(\mathbf{v}) \in L^2(\Omega)\},$$

and the subset of kinematically admissible displacements as

$$\begin{cases} \mathbf{U}_{ad}^p = \mathbf{V}_0^p \text{ for the (VP) sub-model,} \\ \mathbf{U}_{ad}^p = \{\mathbf{v} \in \mathbf{V}_0^p;\ v_n \leq 0 \text{ on } \Gamma_C\} \text{ for the other contact sub-models,} \end{cases}$$

where $\mathbf{V}_0^p$ takes into account the Dirichlet condition for each sub-model. The corresponding spaces of the stress fields are defined as

$$\begin{aligned} \mathbf{X}^q &= \{\boldsymbol{\xi} = (\xi_{ij});\ \xi_{ij} = \xi_{ji},\ \xi_{ij}^D \in L^q(\Omega),\ \mathrm{tr}(\boldsymbol{\xi}) \in L^2(\Omega)\}, \\ \mathbf{H}^q &= \{\boldsymbol{\xi} \in \mathbf{X}^q;\ \mathrm{Div}(\boldsymbol{\xi}) \in [L^q(\Omega)]^3\}, \end{aligned}$$

and are Banach spaces with their natural norms (see [13]).

In order to give a variational formulation of the three sub-models, we assume that the following hypotheses are satisfied:

(H1) $\mathbf{f} \in W^{1,\infty}(0,T;[L^q(\Omega)]^3)$ and $\mathbf{g} \in W^{1,\infty}(0,T;[W^{-\frac{1}{q},q}(\Gamma)]^3 \cap [L^q(\Gamma_N)]^3)$.
(H2) $\Lambda \in [L^\infty(\Omega)]^{81}$ is time independent, symmetric and positively defined.
(H3) $\boldsymbol{\sigma}_0 \in \mathbf{H}^q$, $\mathbf{u}_0 \in \mathbf{U}_{ad}^p$ and verify the natural compatibility conditions for each sub-model.

Then, the usual variational inequality associated with the (CVP) and (CP) sub-models is (see [1])

$$\begin{cases} \text{Find } \mathbf{u} \in W^{1,\infty}(0,T;\mathbf{V}_0^p) \text{ and } \boldsymbol{\sigma} \in W^{1,\infty}(0,T;\mathbf{H}^q) \text{ such that } \mathbf{u} \in \mathbf{U}_{ad}^p \\ \text{and} \\ \displaystyle\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{v}-\mathbf{u})\,\mathrm{d}x \geq \int_\Omega \mathbf{f}\cdot(\mathbf{v}-\mathbf{u})\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}\cdot(\mathbf{v}-\mathbf{u})\,\mathrm{d}\gamma,\ \forall \mathbf{v} \in \mathbf{U}_{ad}^p, \end{cases} \tag{5}$$

for all $t \in (0,T]$, together with the corresponding constitutive law and the initial conditions. For the (VP) sub-model, since there is no contact, we must replace the inequality in (5) by the equality

$$\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x = \int_\Omega \mathbf{f}\cdot\mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}\cdot\mathbf{v}\,\mathrm{d}\gamma,\ \forall \mathbf{v} \in \mathbf{U}_{ad}^p.$$

Existence and uniqueness of solution

We summarize the results of existence and uniqueness of the three sub-models:

- The contact problem in elasticity, (CP) sub-model, has a unique solution. The proof can be found in [16] or [17].

- In [9] it is proved that the elastic-viscoplastic problem without contact, (VP) sub-model, has also a unique solution.
- The proof of the existence of solution for the elastic-viscoplastic problem with contact, (CVP) problem, can be found in [5]. The uniqueness of solution is an open problem.

Some other elastic-viscoplastic laws have been studied, for example, in [15]. The corresponding contact problem can be found in [21].

4 Numerical solution

4.1 ***Discretization in space***

Finite element approximations to the three sub-models are considered in the usual way. We construct a family of finite dimensional subspaces $\mathbf{V}_h^p$ of $\mathbf{V}^p$ by approximating the test functions $\mathbf{v}$ by piecewise polynomials of degree one over a tetrahedral mesh $\mathcal{T}_h$ on the computational domain. The stresses are assumed constant within each element. We will denote by $\mathbf{H}_h^q$ the discretized space of stresses. We will not detail the discretized problems here since the method was described in [4].

4.2 ***Algorithm to the (CP) problem***

The nonlinearities due to the contact condition and the viscoplastic law induce both a nonlipschitzian problem. A simple way to transform these nonlinearities is to use the following lemma. This lemma is the main idea of the algorithm developed in [4] and, in the present work, this result is used combined with Newton techniques based on the semismoothness of the nonlinear functions.

Lemma 2 *Let V be a real Hilbert space and G a maximal monotone operator. If λ is a real positive number, the following statements are equivalent:*

- $u \in G(v),\ u, v \in V.$
- $u = G_\lambda(v + \lambda u),$

where G_λ denotes the Moreau-Yosida approximation of G.

Proof. Since G is maximal monotone, the range of the operator $I+\lambda G$ is V. Hence, $u \in G(v)$ if and only if

$$v = (I+\lambda G)^{-1}(v+\lambda u),$$

and, from the definition of G_λ

$$G_\lambda(v+\lambda u) = \frac{I-(I+\lambda G)^{-1}}{\lambda}(v+\lambda u) = u. \qquad \square$$

Firstly, we will focus on the (CP) contact problem. In [23] it can be found an overview of different methods to solve contact problems. For numerical reasons, following [6,10], to deal with the inequality in (5), we introduce a Lagrange multiplier. For that purpose we consider the following notations:

$$E_h = \{p_h \in L^\infty(\Gamma_C);\ p_h|_C \in P_0(C),\ \forall C \in \mathcal{S}_h\},$$
$$Q_h = \{p_h \in E_h;\ p_h|_C \leq 0,\ \forall C \in \mathcal{S}_h\},$$

$\mathcal{S}_h$ being the triangulation induced by $\mathcal{T}_h$ on Γ_C and $P_0(C)$ the space of polynomials of degree zero defined on face C. To simplify we will omit the subscript h except where it may cause confusion.

In our previous work [4] we proved that the discretized problem in space associated with the weak inequality in (5) can be formulated as a weak equality in the following manner:

Theorem 3 *Every solution of the discretized problem in space associated with (5) is a solution of the problem:*

Find $\mathbf{u} \in W^{1,\infty}(0,T;\mathbf{V}^p_{0h})$, $\boldsymbol{\sigma} \in W^{1,\infty}(0,T;\mathbf{H}^q_h)$ *and* $p \in W^{1,\infty}(0,T;E_h)$ *such that:*

$$\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x = \int_\Omega \mathbf{f}\cdot\mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}\cdot\mathbf{v}\,\mathrm{d}\gamma - \int_{\Gamma_C} p\ v_n\,\mathrm{d}\gamma,\ \forall \mathbf{v} \in \mathbf{V}^p_{0h}, \quad (6)$$

$$\boldsymbol{\sigma} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u})\right), \quad (7)$$

$$p = G_{\lambda_C}(u_n + \lambda_C\ p)\ \text{ on } \Gamma_C, \quad (8)$$

for all $t \in (0,T]$, *and the initial conditions given in (3). Here* λ_C *denotes a positive real number,* u_n *is the normal component of* $\mathbf{u}$ *and*

$$G_{\lambda_C}(\varphi) = \frac{1}{\lambda_C}\left(I - \Pi_{Q_h}\right)(\varphi),\ \varphi \in E_h,$$

Π_{Q_h} *being the projection of* E_h *onto the closed convex* Q_h.

Sketch of the proof. We summarize here the main lines of the proof, which is detailed in [4].

Let us denote by $\boldsymbol{\varepsilon}^*$ the adjoint operator of $\boldsymbol{\varepsilon}$ and by $\mathbf{U}^p_{ad_h}$ the discretized space of admissible displacements. Then, the discretized problem in space associated with (5) is equivalent to

$$\int_{\Omega(t)} \boldsymbol{\varepsilon}^*(\boldsymbol{\sigma}) \cdot (\mathbf{v} - \mathbf{u})\, \mathrm{d}x - I_{\mathbf{U}^p_{ad_h}}(\mathbf{u}) + I_{\mathbf{U}^p_{ad_h}}(\mathbf{v}) \geq L(\mathbf{v} - \mathbf{u}), \forall \mathbf{v} \in \mathbf{V}^p_{0h} \quad (9)$$

where

$$L\mathbf{v} = \int_\Omega \mathbf{f} \cdot \mathbf{v}\, \mathrm{d}x + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v}\, \mathrm{d}\gamma.$$

Using subdifferential notation we can write the previous inequality as

$$-\boldsymbol{\varepsilon}^*(\boldsymbol{\sigma}) + L \in \partial I_{\mathbf{U}^p_{ad_h}}(\mathbf{u}). \quad (10)$$

Let B be the operator

$$\begin{aligned} B : \mathbf{V}^p_{0h} &\longrightarrow E_h \\ \mathbf{v} &\longmapsto B(\mathbf{v}) = p, \text{ defined by: } p|_C = \mathbf{v}|_C \cdot \mathbf{n}|_C, \ \forall C \in \mathcal{S}_h. \end{aligned} \quad (11)$$

It is easy to prove that (see [4])

$$\partial I_{\mathbf{U}^p_{ad_h}}(\mathbf{v}) = B^*(\partial I_{Q_h})(B(\mathbf{v})), \ \forall \mathbf{v} \in \mathbf{V}^p_{0h},$$

so the equation (10) is equivalent to

$$-\boldsymbol{\varepsilon}^*(\boldsymbol{\sigma}) + L \in B^*(\partial I_{Q_h})(B(\mathbf{u})).$$

Then, there exists $p \in \partial I_{Q_h}(B(\mathbf{u}))$ such that

$$-\boldsymbol{\varepsilon}^*(\boldsymbol{\sigma}) + L \in B^*(p),$$

which is equivalent to the weak equation (6). Finally, the expression (8) of p is obtained by introducing the Moreau-Yosida approximation of ∂I_{Q_h} and applying the lemma 2. □

Remark 4 *It is easy to prove that* G_{λ_C} *is contractive for* $\lambda_C > 1$. *So, given* $u_n \in E_h$, p *is the unique fixed point of (8) (see [6,8,24]).*

4.2.1 Approximating contact multiplier

In [3,4] the nonlinear equation $p = G_{\lambda_C}(u_n + \lambda_C\, p)$ was solved by using a fixed point method; this algorithm is robust and converges well for academic tests. Nevertheless, the greater the problem's magnitude, the slower its convergence; in the casting process, in which we must join two nonlinearities -contact and viscoplasticity-, this difficulty becomes more apparent. In this work we will use a generalized Newton method based on the Lipschitzian properties of G_{λ_C} (see [11,18,20]).

Remark 5 *Notice that at each* $\varphi \in E_h$,

$$G_{\lambda_C}(\varphi)_{|C} = \hat{G}_{\lambda_C}(\varphi_{|C}),\ \forall C \in \mathcal{S}_h,$$

where

$$\hat{G}_{\lambda_C}(s) = \begin{cases} 0, & \text{if } s \leq 0, \\ s/\lambda_C, & \text{if } s > 0, \end{cases} \tag{12}$$

at each point $s \in \mathbb{R}$.

Lemma 6 *At each point* $s_0 \in \mathbb{R}$, *the following approximation for* $\hat{G}_{\lambda_C}(s_0)$ *holds true:*

$$\hat{G}_{\lambda_C}(s_0) \cong \hat{G}_{\lambda_C}(s_1) + \begin{cases} 0, & \text{if } s_0 < 0, \\ -V s_1, & \text{if } s_0 = 0, \\ (s_0 - s_1)/\lambda_C, & \text{if } s_0 > 0, \end{cases} \tag{13}$$

for $s_1 \in \mathbb{R}$ *close enough, where* V *belongs to the subdifferential* $\partial \hat{G}_{\lambda_C}(s_0)$. *Furthermore, the error of this approximation is the order* $O\left(|s_0 - s_1|^2\right)$.

Proof. The Lipschitzian function $\hat{G}_{\lambda_C}$, defined in (12), is differentiable at every point $s \neq 0$ and strongly semismooth. So, we can write for all $s_0, s_1 \in \mathbb{R}$ (see [22])

$$\hat{G}_{\lambda_C}(s_0) = \hat{G}_{\lambda_C}(s_1) + V\,(s_0 - s_1) + O\left(|s_0 - s_1|^2\right),$$

where $V \in \partial \hat{G}_{\lambda_C}(s_0)$. □

Remark 7 *It is easy to prove that*

$$\partial \hat{G}_{\lambda_C}(0) = [0, 1/\lambda_C].$$

In order to introduce an iterative algorithm, we discretize the problem (6)-(8) in time using the following notation: the time interval of interest is divided

into N steps, $t^0 = 0$, $t^{j+1} = t^j + \Delta t$, $j = 0, ..., N-1$, with $\Delta t = T/N$; from now on we denote by g^j an approximation of a given function $g(t)$ at time t^j.

Then, from theorem 3 and lemmas 6 and 7, at each time step t^{j+1}, given the starting values $(\mathbf{u}_0^{j+1}, \boldsymbol{\sigma}_0^{j+1}, p_0^{j+1})$, successive approximations $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}, p_k^{j+1})$, $k \geq 1$, of the solution $(\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1}, p^{j+1})$ could be computed by using the recurrence formulas:

$$\int_\Omega \boldsymbol{\sigma}_k^{j+1} : \boldsymbol{\varepsilon}(\mathbf{v})\, \mathrm{d}x + \int_{\Gamma_C} p_k^{j+1}\, v_n\, \mathrm{d}\gamma = \int_\Omega \mathbf{f}^{j+1} \cdot \mathbf{v}\, \mathrm{d}x + \int_{\Gamma_N} \mathbf{g}^{j+1} \cdot \mathbf{v}\, \mathrm{d}\gamma, \quad \forall \mathbf{v} \in \mathbf{V}_{0h}^p, \tag{14}$$

$$\boldsymbol{\sigma}_k^{j+1} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1})\right), \tag{15}$$

$$p_k^{j+1} = p_{k-\frac{1}{2}}^{j+1} + \begin{cases} 0, & \text{if } s_{k-1}^{j+1} \leq 0, \\ \left(s_k^{j+1} - s_{k-1}^{j+1}\right)/\lambda_C, & \text{if } s_{k-1}^{j+1} > 0, \end{cases} \tag{16}$$

where

$$s_k^{j+1} = \left(u_k^{j+1}\right)_n + \lambda_C\, p_k^{j+1}, \tag{17}$$

$$p_{k-\frac{1}{2}}^{j+1} = G_{\lambda_C}(s_{k-1}^{j+1}). \tag{18}$$

Notice that approximation (16) corresponds to the choice $s_0 = s_{k-1}^{j+1}$, $s_1 = s_k^{j+1}$ and $V = 0$ for $s_0 = 0$ in (13). It is important to remark that this fact has no relevant numerical influence because the region $[s_{k-1} = 0]$ tends to the free surface, which has null measure.

Remark 8 *Note that at each iteration the variational equality corresponding to (14) is equivalent to the linear problem*

$$-\mathrm{Div}\left(\boldsymbol{\sigma}_k^{j+1}\right) = \mathbf{f}^{j+1} \text{ in } \Omega, \tag{19}$$

$$\boldsymbol{\sigma}_k^{j+1}\mathbf{n} = \mathbf{g}^{j+1} \text{ on } \Gamma_N, \tag{20}$$

$$(\sigma_k^{j+1})_n = -p_k^{j+1},\ (\boldsymbol{\sigma}_k^{j+1})_\tau = \mathbf{0} \text{ on } \Gamma_C, \tag{21}$$

$$\mathbf{u}_k^{j+1} = \mathbf{0} \text{ on } \Gamma_D, \tag{22}$$

in the distributional sense; so the contact multiplier p_k^{j+1} represents the obstacle reaction.

If we analyze formulas (14)-(18), there is a coupling between the displacements and the contact multiplier on the boundary Γ_C; furthermore, from (21), normal stresses only depend on the contact multiplier on this boundary. In what

follows, our objective is to split Γ_C at each iteration into two parts: on the first part we will know the contact multiplier –and, in turn, the normal stresses–, on the other part the normal displacements will be null. This procedure will first allow us to compute displacements and stresses at each iteration, and then to compute the contact multiplier.

Lemma 9 *Let us consider the following sets:*

$$(\Gamma_{C,k}^{-})^{j+1} = \{C \in S_h;\ s_k^{j+1} \leq 0\}, \tag{23}$$

$$(\Gamma_{C,k}^{+})^{j+1} = \{C \in S_h;\ s_k^{j+1} > 0\}. \tag{24}$$

Then, $p_k^{j+1} = 0$ *on* $(\Gamma_{C,k-1}^{-})^{j+1}$ *and* $\left(u_k^{j+1}\right)_n = 0$ *on* $(\Gamma_{C,k-1}^{+})^{j+1}$.

Proof. With the notation introduced in (23)-(24), relation (16) can be rewritten as

$$p_k^{j+1} = \begin{cases} 0, & \text{on } (\Gamma_{C,k-1}^{-})^{j+1}, \\ \\ p_{k-\frac{1}{2}}^{j+1} + \left(s_k^{j+1} - s_{k-1}^{j+1}\right)/\lambda_C, & \text{on } (\Gamma_{C,k-1}^{+})^{j+1}, \end{cases}$$

since $\hat{G}_{\lambda_C}$ is null on each $C \in (\Gamma_{C,k-1}^{-})^{j+1}$. So, we can easily update the multiplier p_k^{j+1} by zero on $(\Gamma_{C,k-1}^{-})^{j+1}$. Nevertheless, on each face of $(\Gamma_{C,k-1}^{+})^{j+1}$, taking into account (17) and (18), we must compute p_k^{j+1} verifying

$$p_k^{j+1} = \frac{s_{k-1}^{j+1}}{\lambda_C} + \frac{\left(s_k^{j+1} - s_{k-1}^{j+1}\right)}{\lambda_C} = \frac{1}{\lambda_C}\left(u_k^{j+1}\right)_n + p_k^{j+1},$$

and this equality can only be verified if

$$\left(u_k^{j+1}\right)_n = 0 \text{ on } (\Gamma_{C,k-1}^{+})^{j+1}. \qquad \square \tag{25}$$

Thus, at each iteration the boundary Γ_C is split into two parts, $(\Gamma_{C,k-1}^{+})^{j+1}$ (where the normal displacements and the tangential component of stresses are null) and $(\Gamma_{C,k-1}^{-})^{j+1}$(where the normal and tangential stresses are null). It is very important to note that this is the only data which is modified at each iteration.

There are several ways to achieve the solution using formulas (14)-(18), but since our objective is to couple the contact effects with the viscoplastic ones, in the next subsection we propose an algorithm based on two steps. In the first one, we compute $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1})$ using lemma 9 and remark 8 to remove the contact multiplier in equation (14). Afterwards, we compute p_k^{j+1} and update $(\Gamma_{C,k}^{+})^{j+1}$.

4.2.2 Algorithm to contact problem

Step 1: Given $(\mathbf{u}_{k-1}^{j+1}, \boldsymbol{\sigma}_{k-1}^{j+1}, p_{k-1}^{j+1})$, we compute $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}) \in \mathbf{V}_{0h}^{p*} \times \mathbf{H}_h^q$ by solving the variational equality

$$\int_\Omega \boldsymbol{\sigma}_k^{j+1} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x = \int_\Omega \mathbf{f}^{j+1} \cdot \mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}^{j+1} \cdot \mathbf{v}\,\mathrm{d}\gamma,\ \forall \mathbf{v} \in \mathbf{V}_{0h}^{p*},$$

together with the constitutive law (15), where

$$\mathbf{V}_{0h}^{p*} = \{\mathbf{v} \in \mathbf{V}_{0h}^{p};\ v_n = 0 \text{ on } (\Gamma_{C,k-1}^+)^{j+1}\}.$$

In this first step the tangential component of $\boldsymbol{\sigma}_k^{j+1}$ is assumed to be null on Γ_C, but normal stresses due to the contact are unknown on $(\Gamma_{C,k-1}^+)^{j+1}$. For the practical computation we impose the Dirichlet condition (25) on $(\Gamma_{C,k-1}^+)^{j+1}$ by using a penalty method in the variational equation

$$\int_\Omega \boldsymbol{\sigma}_k^{j+1} : \boldsymbol{\varepsilon}\,(\mathbf{v})\,\mathrm{d}x + \frac{1}{\epsilon}\int_{(\Gamma_{C,k-1}^+)^{j+1}} \left(u_k^{j+1}\right)_n v_n\,\mathrm{d}\gamma = \int_\Omega \mathbf{f}^{j+1} \cdot \mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}^{j+1} \cdot \mathbf{v}\,\mathrm{d}\gamma, \tag{26}$$

for all $\mathbf{v} \in \mathbf{V}_{0h}^p$, where ϵ is a small parameter.

Step 2: Known $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1})$ and given that $p_k^{j+1} = 0$ on $(\Gamma_{C,k-1}^-)^{j+1}$, we update p_k^{j+1} on $(\Gamma_{C,k-1}^+)^{j+1}$ thanks to the variational equality

$$\int_{(\Gamma_{C,k-1}^+)^{j+1}} p_k^{j+1} v_n\,\mathrm{d}\gamma = -\int_\Omega \boldsymbol{\sigma}_k^{j+1} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x + \int_\Omega \mathbf{f}^{j+1} \cdot \mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}^{j+1} \cdot \mathbf{v}\,\mathrm{d}\gamma, \tag{27}$$

for all $\mathbf{v} \in \mathbf{V}_{0h}^p$ and we update $(\Gamma_{C,k}^+)^{j+1}$ using (24).

Remark 10 *From a numerical point of view, the variational equation (27) is equivalent to solve the linear system*

$$(\mathbf{B}\mathbf{B}^*)\,\mathbf{p}_k^{j+1} = \mathbf{B}\left(\mathbf{b}_k^{j+1} - \mathbf{A}_k^{j+1}\mathbf{u}_k^{j+1}\right),$$

where $\mathbf{A}_k^{j+1}$ is the stiffness matrix, $\mathbf{b}_k^{j+1}$ the external load vector (corresponding to the second member in (26)), $\mathbf{B}$ is the rectangular matrix associated with the operator $\mathbf{v} \in \mathbf{V}_{0h}^p \to v_n \in L^p\left((\Gamma_{C,k-1}^+)^{j+1}\right)$ and $\mathbf{B}^*$ its transpose. We notice that the matrix $\mathbf{B}\mathbf{B}^*$ of this system has a very low dimension. In that equation $\mathbf{p}_k^{j+1}$ denotes the vector containing the value of the contact multiplier at each $C \in \mathcal{S}_h$.

As we will see in section 6, numerical experiments show that this combination between Newton's techniques and penalty methods works well.

On the other hand, we would like to emphasize that this algorithm modifies the stiffness matrix at each iteration, but only at the rows corresponding to the contact nodes. Because the matrix is symmetric, we can store it in a one-dimensional array using a sparsity method and Cholesky's factorization is used at each iteration. In order to reduce the computing time, we propose numbering the nodes in such a way that the higher numbers correspond to contact nodes. This will permit the conservation of Cholesky's factorization for a good deal of the matrix.

4.3 ***Algorithm to the (VP) problem***

In this section we deduce an algorithm to the (VP) discretized weak problem defined by equations:

$$\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x = \int_\Omega \mathbf{f}\cdot\mathbf{v}\,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g}\cdot\mathbf{v}\,\mathrm{d}\gamma,\ \forall \mathbf{v}\in\mathbf{V}_{0h}^p, \tag{28}$$

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \Lambda\dot{\boldsymbol{\sigma}} + \theta_0 \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D, \tag{29}$$

$$\mathbf{u}(0) = \mathbf{u}_0,\ \boldsymbol{\sigma}(0) = \boldsymbol{\sigma}_0, \tag{30}$$

for all $t\in(0,T]$, where $\mathbf{u}_0, \boldsymbol{\sigma}_0$ are the discretized initial conditions in space. For that purpose, the elastic-viscoplastic law (29) is discretized in time by using an implicit Euler scheme. Let $(0,T)$ be the time interval of interest discretized into N steps: $t^0 = 0$, $t^{j+1} = t^j + \Delta t$, $j = 0,\ldots,N-1$, with $\Delta t = T/N$. We denote by g^j an approximation of a given function $g(t)$ at time t^j. In the next result we obtain an expression for the stress tensor at each time step. To avoid the nonlinearity due to the viscoplastic law we introduce a new multiplier called the viscoplastic multiplier.

Proposition 11 *At each time step t^{j+1}, $j = 0,\ldots,N-1$, the stress tensor is given by the relation*

$$\boldsymbol{\sigma}^{j+1} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}) - \Delta t\mathbf{q}^{j+1} + \boldsymbol{\zeta}^j\right), \tag{31}$$

where $\mathbf{q}^{j+1} = \nabla\Phi_q\left((\boldsymbol{\sigma}^{j+1})^D\right)$ is the viscoplastic multiplier, and

$$\boldsymbol{\zeta}^j = \Lambda\boldsymbol{\sigma}^j - \boldsymbol{\varepsilon}(\mathbf{u}^j), \tag{32}$$

is a function of the material's history.

Proof. We discretize the elastic-viscoplastic law at the time interval $[t^j, t^{j+1}]$ by using the implicit Euler scheme; then, we have

$$\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}) - \boldsymbol{\varepsilon}(\mathbf{u}^j) = \Lambda\left(\boldsymbol{\sigma}^{j+1} - \boldsymbol{\sigma}^j\right) + \Delta t\ \nabla\Phi_q\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right). \tag{33}$$

Because the trace of the deviatoric tensor is null, using (32) and the Hooke's law (1) we deduce from (33)

$$\sigma_{kk}^{j+1} = \frac{E}{1-2\nu}\left(\varepsilon_{kk}(\mathbf{u}^{j+1}) + \zeta_{kk}^j\right), \tag{34}$$

so the spheric part of $\boldsymbol{\sigma}^{j+1}$ can be computed from $\mathbf{u}^{j+1}$ by an explicit relation. Moreover, the deviatoric part $\left(\boldsymbol{\sigma}^{j+1}\right)^D$ must be obtained as a solution of a nonlinear equation. In effect,

$$\boldsymbol{\sigma}^{j+1} = \left(\boldsymbol{\sigma}^{j+1}\right)^D + \frac{E}{3\,(1-2\nu)}\left(\varepsilon_{kk}(\mathbf{u}^{j+1}) + \zeta_{kk}^j\right)\mathbf{I},$$

and then, replacing this expression in (33), we deduce

$$\begin{aligned}\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}) - \Lambda\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D + \frac{E}{3\,(1-2\nu)}\varepsilon_{kk}(\mathbf{u}^{j+1})\mathbf{I}\right)\\ -\Delta t\ \nabla\Phi_q\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right) = -\left(\boldsymbol{\zeta}^j\right)^D.\end{aligned} \tag{35}$$

Simplifying equation (35), we obtain

$$\boldsymbol{\varepsilon}^D(\mathbf{u}^{j+1}) - \Lambda\left(\boldsymbol{\sigma}^{j+1}\right)^D - \Delta t\ \nabla\Phi_q\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right) = -\left(\boldsymbol{\zeta}^j\right)^D. \tag{36}$$

Introducing the new multiplier $\mathbf{q}^{j+1} = \nabla\Phi_q\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right)$, equation (36) can be rewritten as

$$\boldsymbol{\varepsilon}^D(\mathbf{u}^{j+1}) - \Lambda\left(\boldsymbol{\sigma}^{j+1}\right)^D = \Delta t\mathbf{q}^{j+1} - \left(\boldsymbol{\zeta}^j\right)^D,$$

and consequently

$$\left(\boldsymbol{\sigma}^{j+1}\right)^D = \Lambda^{-1}\left(\boldsymbol{\varepsilon}^D(\mathbf{u}^{j+1}) - \Delta t\mathbf{q}^{j+1} + \left(\boldsymbol{\zeta}^j\right)^D\right). \tag{37}$$

Summing up, from (34) and (37), we obtain the formula (31). □

To compute the multiplier $\mathbf{q}^{j+1}$ we give the following lemma, which is based on the maximal monotone operator techniques developed in [6].

Lemma 12 *At each time step t^{j+1}, $j = 0, \ldots, N-1$, the viscoplastic multiplier $\mathbf{q}^{j+1}$ is a fixed point of the equation*

$$\mathbf{q}^{j+1} = \left(\nabla\Phi_q\right)_{\lambda_P}\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D + \lambda_P\mathbf{q}^{j+1}\right), \tag{38}$$

for every $\lambda_P > 0$. In this equation, $(\nabla\Phi_q)_{\lambda_P}$ denotes the Moreau-Yosida approximation of $\nabla\Phi_q$ given by the expression

$$(\nabla\Phi_q)_{\lambda_P}(\boldsymbol{\zeta}) = \frac{1}{\lambda_P}\left(1-\frac{1}{\mu}\right)\boldsymbol{\zeta}, \tag{39}$$

where $\mu = \mu(\boldsymbol{\zeta})$ is the unique root of the equation

$$\mu^{q-1} - \mu^{q-2} - \lambda_P\theta_0|\boldsymbol{\zeta}|^{q-2} = 0,$$

in the interval $[1, +\infty)$.

Sketch of the proof. This proof is based on the lemma 2. Indeed, by definition, $\mathbf{q}^{j+1} = \nabla\Phi_q\left((\boldsymbol{\sigma}^{j+1})^D\right)$, so, applying that lemma it results (38). The remainder of the proof consists of calculating the Moreau-Yosida approximation of $\nabla\Phi_q$. These calculus can be found in [4]. □

Remark 13 *Since $\nabla\Phi_q$ is a maximal monotone operator, $(\nabla\Phi_q)_{\lambda_P}$ is contractive for $\lambda_P > 1$ (see [6,8,24]). Then, the fixed point of (38) is unique.*

4.3.1 Approximating viscoplasticity multiplier

As noted above, a key component of our procedure must be that the stiffness matrix is preserved and we do not need to recalculate it at each iteration. At the same time we wish to achieve the efficiency of the Newton techniques. To this end, assuming that the stress tensor is known, we linearize the second term in equation (38) and so we obtain the next result.

Lemma 14 *Given any $\boldsymbol{\sigma} \in \mathbb{R}^9_s$ and $\lambda_P > 1$, the fixed point $\mathbf{q} \in \mathbb{R}^9_s$ of the equation*

$$\mathbf{q} = (\nabla\Phi_q)_{\lambda_P}\left(\boldsymbol{\sigma}^D + \lambda_P\mathbf{q}\right), \tag{40}$$

can be approximated by

$$\begin{aligned}\mathbf{q} \cong\ & (\nabla\Phi_q)_{\lambda_P}\left(\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}}\right) + \left(1-\frac{1}{\mu}\right)(\mathbf{q}-\hat{\mathbf{q}}) \\ & + \gamma\left(\left(\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}}\right) : (\mathbf{q}-\hat{\mathbf{q}})\right)\left(\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}}\right),\end{aligned} \tag{41}$$

for $\hat{\mathbf{q}} \in \mathbb{R}^9_s$ close enough to $\mathbf{q}$ and provided that $\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}} \neq \mathbf{0}$. Here $\mu = \mu\left(\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}}\right)$, and γ is given by the expression

$$\gamma = \frac{(q-2)(\mu-1)}{\mu\left((q-1)\mu - (q-2)\right)\left|\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}}\right|^2}. \tag{42}$$

The order of this approximation is $O\left(|\mathbf{q}-\hat{\mathbf{q}}|^2\right)$.

Proof. Thanks to the implicit function theorem, $(\nabla\Phi_q)_{\lambda_P}(\boldsymbol{\xi})$ is twice-differentiable for $\boldsymbol{\xi} \neq \mathbf{0}$; so we can apply Taylor's formula to approximate $\mathbf{q}$. Then, for $\hat{\mathbf{q}} \in \mathbb{R}^9_s$ close enough, if $\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}} \neq \mathbf{0}$, we obtain

$$\mathbf{q} = (\nabla\Phi_q)_{\lambda_P}(\boldsymbol{\sigma}^D+\lambda_P\hat{\mathbf{q}})+D(\nabla\Phi_q)_{\lambda_P}(\boldsymbol{\sigma}^D+\lambda_P\hat{\mathbf{q}})(\lambda_P(\mathbf{q}-\hat{\mathbf{q}}))+O\left(|\mathbf{q}-\hat{\mathbf{q}}|^2\right).$$

It is easy to prove that if we compute $D(\nabla\Phi_q)_{\lambda_P}$ from the expression (39), we obtain (41). □

Remark 15 *The case $\boldsymbol{\sigma}^D + \lambda_P\hat{\mathbf{q}} = \mathbf{0}$ was excluded from lemma 14 since from equation (38), if $\boldsymbol{\sigma}^D + \lambda_P\mathbf{q} = \mathbf{0}$, then $\mathbf{q} = \mathbf{0}$ is the fixed point.*

Summing up, if we replace the expression (31) in the equality (28), we obtain a displacements formulation of our problem. So, we propose the following iterative algorithm: At each time step t^{j+1}, given the starting values $(\mathbf{u}_0^{j+1}, \boldsymbol{\sigma}_0^{j+1}, \mathbf{q}_0^{j+1})$, successive approximations $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}, \mathbf{q}_k^{j+1})$, $k \geq 1$, of the solution $(\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1}, \mathbf{q}^{j+1})$ could be computed by using the recurrence formulas:

$$\begin{aligned}\int_\Omega \Lambda^{-1}\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x = \int_\Omega \Lambda^{-1}\left(\Delta t\mathbf{q}_{k-1}^{j+1} - \boldsymbol{\zeta}^j\right) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x \\ + \int_\Omega \mathbf{f}^{j+1}\cdot\mathbf{v}\;\mathrm{d}x + \int_{\Gamma_N}\mathbf{g}^{j+1}\cdot\mathbf{v}\;\mathrm{d}\gamma,\ \forall\mathbf{v}\in\mathbf{V}_{0h}^p,\end{aligned} \tag{43}$$

$$\boldsymbol{\sigma}_k^{j+1} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) - \Delta t\mathbf{q}_{k-1}^{j+1} + \boldsymbol{\zeta}^j\right), \tag{44}$$

$$\mathbf{q}_k^{j+1} = \begin{cases} \mathbf{0}, & \text{if } \boldsymbol{\kappa}_k^{j+1} = \mathbf{0}, \\ (\nabla\Phi_q)_{\lambda_P}(\boldsymbol{\kappa}_k^{j+1}) + \left(1 - \frac{1}{\mu_k^{j+1}}\right)\left(\mathbf{q}_k^{j+1} - \mathbf{q}_{k-1}^{j+1}\right) & \\ \qquad +\gamma\left(\boldsymbol{\kappa}_k^{j+1} : (\mathbf{q}_k^{j+1} - \mathbf{q}_{k-1}^{j+1})\right)\boldsymbol{\kappa}_k^{j+1}, & \text{if } \boldsymbol{\kappa}_k^{j+1} \neq \mathbf{0}, \end{cases} \tag{45}$$

where

$$\boldsymbol{\zeta}^j = \Lambda\boldsymbol{\sigma}^j - \boldsymbol{\varepsilon}(\mathbf{u}^j), \tag{46}$$

$$\boldsymbol{\kappa}_k^{j+1} = \left(\boldsymbol{\sigma}_k^{j+1}\right)^D + \lambda_P\,\mathbf{q}_{k-1}^{j+1}, \tag{47}$$

and γ is given by (42) for $\mu = \mu_k^{j+1} = \mu(\boldsymbol{\kappa}_k^{j+1})$, $\hat{\mathbf{q}} = \mathbf{q}_{k-1}^{j+1}$ and $\boldsymbol{\sigma} = \boldsymbol{\sigma}_k^{j+1}$.

The fixed point of (45) can be explicitly computed thanks to the following lemma.

Lemma 16 *At each time step t^{j+1}, $j = 0, \ldots, N-1$, and at each iteration $k > 0$, the fixed point $\mathbf{q}_k^{j+1}$ of the equation (45) can be computed using the*

recurrence formula

$$\mathbf{q}_k^{j+1} = \begin{cases} \mathbf{0}, & \text{if } \boldsymbol{\kappa}_k^{j+1} = \mathbf{0}, \\ \dfrac{\left(\mu_k^{j+1}-1\right)}{\lambda_P}\left(\left(\boldsymbol{\sigma}_k^{j+1}\right)^D + \left(\mu_k^{j+1}-1\right)\dfrac{q-2}{\mu_k^{j+1}}\boldsymbol{\kappa}_k^{j+1}\right) & \\ \quad -\left(\mu_k^{j+1}-1\right)\dfrac{q-2}{|\boldsymbol{\kappa}_k^{j+1}|^2}\left(\mathbf{q}_{k-1}^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)\boldsymbol{\kappa}_k^{j+1}, & \text{if } \boldsymbol{\kappa}_k^{j+1} \neq \mathbf{0}, \end{cases} \tag{48}$$

where $\boldsymbol{\kappa}_k^{j+1}$ *is given by (47) and* $\mu_k^{j+1} = \mu\left(\boldsymbol{\kappa}_k^{j+1}\right)$.

Proof. Let us consider $\boldsymbol{\kappa}_k^{j+1} \neq \mathbf{0}$. Taking into account the expression of $(\nabla\Phi_q)_{\lambda_P}$ given by (39), we obtain from (45) the equality

$$\begin{aligned} \mathbf{q}_k^{j+1} = &\left(1-\frac{1}{\mu_k^{j+1}}\right)\left(\frac{1}{\lambda_P}\boldsymbol{\kappa}_k^{j+1} + \left(\mathbf{q}_k^{j+1}-\mathbf{q}_{k-1}^{j+1}\right)\right) \\ &+\gamma_k^{j+1}\left(\boldsymbol{\kappa}_k^{j+1}:\left(\mathbf{q}_k^{j+1}-\mathbf{q}_{k-1}^{j+1}\right)\right)\boldsymbol{\kappa}_k^{j+1}, \end{aligned} \tag{49}$$

where

$$\gamma_k^{j+1} = \frac{(q-2)\left(\mu_k^{j+1}-1\right)}{\mu_k^{j+1}\left((q-1)\,\mu_k^{j+1}-(q-2)\right)|\boldsymbol{\kappa}_k^{j+1}|^2}. \tag{50}$$

In order to obtain an explicit expression for $\mathbf{q}_k^{j+1}$, we multiply equation (49) by $\boldsymbol{\kappa}_k^{j+1}$ which allows us to rewrite $\left(\mathbf{q}_k^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)$ as

$$\left(\mathbf{q}_k^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right) = \chi_k^{j+1}\left(1-\frac{1}{\mu_k^{j+1}}\right)\frac{|\boldsymbol{\kappa}_k^{j+1}|^2}{\lambda_P} + \left(\mathbf{q}_{k-1}^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)\left(1-\chi_k^{j+1}\right),$$

where

$$\chi_k^{j+1} = \frac{\mu_k^{j+1}}{1-\mu_k^{j+1}\gamma_k^{j+1}|\boldsymbol{\kappa}_k^{j+1}|^2} = \mu_k^{j+1}(q-1)-(q-2). \tag{51}$$

Then, replacing this expression in equation (49) we deduce that

$$\begin{aligned} \mathbf{q}_k^{j+1} = &\left(1-\frac{1}{\mu_k^{j+1}}\right)\left(\frac{1}{\lambda_P}\boldsymbol{\kappa}_k^{j+1} + \left(\mathbf{q}_k^{j+1}-\mathbf{q}_{k-1}^{j+1}\right)\right) \\ &-\gamma_k^{j+1}\left(\boldsymbol{\kappa}_k^{j+1}:\mathbf{q}_{k-1}^{j+1}\right)\boldsymbol{\kappa}_k^{j+1} + \gamma_k^{j+1}\{\chi_k^{j+1}\left(1-\frac{1}{\mu_k^{j+1}}\right)\frac{|\boldsymbol{\kappa}_k^{j+1}|^2}{\lambda_P} \\ &+\left(\mathbf{q}_{k-1}^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)\left(1-\chi_k^{j+1}\right)\}\boldsymbol{\kappa}_k^{j+1}. \end{aligned}$$

So, the updated viscoplasticity multiplier is characterized by the explicit expression

$$\begin{aligned}\mathbf{q}_k^{j+1} &= \frac{\left(\mu_k^{j+1}-1\right)}{\lambda_P}\left(\boldsymbol{\sigma}_k^{j+1}\right)^D - \mu_k^{j+1}\gamma_k^{j+1}\left(\boldsymbol{\kappa}_k^{j+1}\colon \mathbf{q}_{k-1}^{j+1}\right)\boldsymbol{\kappa}_k^{j+1} \\ &\quad + \gamma_k^{j+1}\left\{\chi_k^{j+1}\left(\mu_k^{j+1}-1\right)\frac{|\boldsymbol{\kappa}_k^{j+1}|^2}{\lambda_P} + \left(\mathbf{q}_{k-1}^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)\mu_k^{j+1}\left(1-\chi_k^{j+1}\right)\right\}\boldsymbol{\kappa}_k^{j+1} \\ &= \frac{\left(\mu_k^{j+1}-1\right)}{\lambda_P}\left(\left(\boldsymbol{\sigma}_k^{j+1}\right)^D + \gamma_k^{j+1}\chi_k^{j+1}|\boldsymbol{\kappa}_k^{j+1}|^2\boldsymbol{\kappa}_k^{j+1}\right) \\ &\quad - \mu_k^{j+1}\gamma_k^{j+1}\chi_k^{j+1}\left(\mathbf{q}_{k-1}^{j+1}:\boldsymbol{\kappa}_k^{j+1}\right)\boldsymbol{\kappa}_k^{j+1}.\end{aligned}$$

Finally, taking into account expressions (50) and (51) we obtain formula (48) to update the viscoplasticity multiplier. □

4.3.2 Algorithm to viscoplastic problem

Summing up, the proposed algorithm to the (VP) sub-model is:

- Let $(\mathbf{u}_0, \boldsymbol{\sigma}_0)$ be given; we compute $\mathbf{q}^0 = \nabla\Phi_q(\boldsymbol{\sigma}_0^D)$.
- Then, for $j \geq 0$, $(\mathbf{u}^j,\ \boldsymbol{\sigma}^j,\ \mathbf{q}^j)$ known at time t^j, we determine $(\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1},\ \mathbf{q}^{j+1})$ at time t^{j+1}, an approximated weak solution of the (VP) problem. To do so, we propose the following iterative algorithm:

i) $\mathbf{q}_0^{j+1} \in \mathbb{R}^9_s$ specified by $\mathbf{q}_0^{j+1} = \mathbf{q}^j$,

ii) with $\mathbf{q}_{k-1}^{j+1}$ known, calculate $\left(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}\right)$ by solving

$$\begin{aligned}\int_\Omega \Lambda^{-1}\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x &= \int_\Omega \Lambda^{-1}\left(\Delta t\mathbf{q}_{k-1}^{j+1} - \boldsymbol{\zeta}^j\right) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x \\ &\quad + \int_\Omega \mathbf{f}^{j+1}\cdot\mathbf{v}\ \mathrm{d}x + \int_{\Gamma_N}\mathbf{g}^{j+1}\cdot\mathbf{v}\ \mathrm{d}\gamma,\ \forall\mathbf{v}\in\mathbf{V}_{0h}^p,\end{aligned}$$

where

$$\boldsymbol{\zeta}^j = \Lambda\boldsymbol{\sigma}^j - \boldsymbol{\varepsilon}(\mathbf{u}^j),$$

and

$$\boldsymbol{\sigma}_k^{j+1} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) - \Delta t\mathbf{q}_{k-1}^{j+1} + \boldsymbol{\zeta}^j\right);$$

iii) the updated viscoplasticity multiplier is given by relation (48).

5 Algorithm to the (CVP) problem

Let us now summarize the iterative procedure for the resolution of the (CVP) complete model in the following scheme. For this, we must couple the algorithm

to the contact problem given in section 4.2.2 with the (VP) one. Then, if we replace in (26) the expression (31) for the stress tensor, we obtain the following algorithm:

- Let $(\mathbf{u}_0, \boldsymbol{\sigma}_0)$ be given; so, using remark 8, we initialize $p^0 = -(\sigma_0)_n$ and $\mathbf{q}^0 = \nabla\Phi_q\left(\boldsymbol{\sigma}_0^D\right)$.
- Then, for $j \geq 0$, ($\mathbf{u}^j$, $\boldsymbol{\sigma}^j$, $\mathbf{q}^j$, p^j) known at time t^j, we determine ($\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1}$, $\mathbf{q}^{j+1}$, p^{j+1}) at time t^{j+1}, an approximated weak solution of problem (3). To do so, we propose the following iterative algorithm:

i) $\mathbf{q}_0^{j+1} \in \mathbb{R}_s^9$ and $p_0^{j+1} \in \mathbb{R}$ are given by $\mathbf{q}_0^{j+1} = \mathbf{q}^j$, $p_0^{j+1} = p^j$.

ii) With $\mathbf{q}_{k-1}^{j+1}$, p_{k-1}^{j+1} known, calculate $\left(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}\right)$ by solving

Step 1:

$$\begin{aligned}\int_\Omega \Lambda^{-1}\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x + \frac{1}{\epsilon}\int_{(\Gamma^+_{C,k-1})^{j+1}} \left(u_k^{j+1}\right)_n v_n\,\mathrm{d}\gamma \\ = \int_\Omega \Lambda^{-1}\left(\Delta t\mathbf{q}_{k-1}^{j+1} - \boldsymbol{\zeta}^j\right) : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x \\ + \int_\Omega \mathbf{f}^{j+1}\cdot\mathbf{v}\;\mathrm{d}x + \int_{\Gamma_N}\mathbf{g}^{j+1}\cdot\mathbf{v}\;\mathrm{d}\gamma,\ \forall\mathbf{v}\in\mathbf{V}_{0h}^p; \qquad (52)\end{aligned}$$

where

$$\boldsymbol{\zeta}^j = \Lambda\boldsymbol{\sigma}^j - \boldsymbol{\varepsilon}(\mathbf{u}^j).$$

Step 2: The updated contact multiplier is given by solving the linear system

$$\left(\mathbf{B}\mathbf{B}^*\right)\mathbf{p}_k^{j+1} = \mathbf{B}\left(\mathbf{b}_k^{j+1} - \mathbf{A}_k^{j+1}\mathbf{u}_k^{j+1}\right),$$

where $\mathbf{A}_k^{j+1}$ is the stiffness matrix without the penalty term, $\mathbf{B}$ the constraint operator restricting the values of the normal displacements to the contact surface, $\mathbf{B}^*$ its transpose and $\mathbf{b}_k^{j+1}$ the vector associated with the linear form corresponding to the second member of (52). In this equation, $\mathbf{p}_k^{j+1}$ denotes the vector containing the contact multiplier at each $C \in \mathcal{S}_h$.

iii) The updated stress tensor is defined by

$$\boldsymbol{\sigma}_k^{j+1} = \Lambda^{-1}\left(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) - \Delta t\mathbf{q}_{k-1}^{j+1} + \boldsymbol{\zeta}^j\right).$$

iv) The updated viscoplasticity multiplier is given by the relation

$$\mathbf{q}_k^{j+1} = \begin{cases} \mathbf{0}, & \text{if } \boldsymbol{\kappa}_k^{j+1} = \mathbf{0}, \\ \dfrac{\left(\mu_k^{j+1}-1\right)}{\lambda_P}\left(\left(\boldsymbol{\sigma}_k^{j+1}\right)^D + \left(\mu_k^{j+1}-1\right)\dfrac{q-2}{\mu_k^{j+1}}\boldsymbol{\kappa}_k^{j+1}\right) & \\ \quad -\left(\mu_k^{j+1}-1\right)\dfrac{q-2}{|\boldsymbol{\kappa}_k^{j+1}|^2}\left(\mathbf{q}_{k-1}^{j+1} : \boldsymbol{\kappa}_k^{j+1}\right)\boldsymbol{\kappa}_k^{j+1}, & \text{if } \boldsymbol{\kappa}_k^{j+1} \neq \mathbf{0}, \end{cases}$$

where $\boldsymbol{\kappa}_k^{j+1} = \left(\boldsymbol{\sigma}_k^{j+1}\right)^D + \lambda_P \mathbf{q}_{k-1}^{j+1}$, $\lambda_P > 1$ and μ_k^{j+1} is the unique root of the equation

$$\mu^{q-1} - \mu^{q-2} - \lambda_P \theta_0 |\boldsymbol{\kappa}_k^{j+1}|^{q-2} = 0,$$

in the interval $[1, +\infty)$.

Remark 17 *In practice, updating the viscoplasticity and contact multipliers is performed with a relaxation parameter ϑ. For example, for the (VP) algorithm the updated viscoplasticity multiplier is obtained by the formula*

$$\mathbf{q}_k^{j+1} = \vartheta \mathbf{q}_k^{j+1} + (1-\vartheta)\mathbf{q}_{k-1}^{j+1}, \quad 0 < \vartheta \leq 1.$$

6 Numerical results

In this section, we will check the algorithms proposed herein by applying the numerical code on two examples of plane strains; these tests have been designed to try to reproduce a behaviour analogous to semicontinuous real casting behaviour. To show the efficiency of the new algorithms, we compare the cpu-time [2], the number of iterations, and the displacement and stress errors between the new algorithm (NM) and the old one (FPM), explained in [3,4].

6.1 Testing problems with large gradients

With this first test we will try to check the algorithm for a (CVP) problem with displacements and stresses whose gradients have a magnitude similar to the magnitude of real aluminium casting gradients. We will show that for the NM algorithm it is necessary to introduce an adimensionalization of stresses in order to obtain a good approximation and improve the convergence of the algorithm. Although the NM algorithm needs a small time step for convergence, we will see that an optimization of the time step and an Armijo rule for the viscoplastic algorithm will allow us to increase Δt.

Let $[0, 0.5]$ be the time interval and Ω be the cylindrical body whose axis is parallel to x_1-direction and its section the square of dimensions 0.5 m×0.5 m in the plane x_2x_3 . On $\partial\Omega$ we distinguish four parts: $\partial\Omega = \bar{\Gamma}_D \cup \bar{\Gamma}_N \cup \bar{\Gamma}_C \cup \bar{\Gamma}_{\pm}$,

[2] The numerical solution was computed on a PC with AMD Athlon XP 1700+ processor running on LINUX.

where

$$\bar{\Gamma}_D = \partial\Omega \cap [x_2 = 0],\ \bar{\Gamma}_N = \partial\Omega \cap ([x_2 = 0.5] \cup [x_3 = 0.5]),$$
$$\bar{\Gamma}_C = \partial\Omega \cap [x_3 = 0],\ \bar{\Gamma}_\pm = \partial\Omega \cap [x_1 = \pm 1].$$

The material parameters corresponding to the elastic-viscoplastic law (2) are

$$E = 10^9 \mathrm{N/m^2},\ \nu = 0.35,\ \theta_0 = 1.953125 \times 10^{-39} \mathrm{m^2/(sN)},\ q = 6.$$

The (CVP) problem to solve is

$$\left.\begin{aligned}
\mathrm{Div}(\boldsymbol{\sigma}) = \mathbf{0} \text{ in } \Omega,\\
\mathbf{u} = h(t)\,(0, x_2, -x_3) \text{ on } \Gamma_D,\\
\boldsymbol{\sigma}\mathbf{n} = 10^8 \mathbf{g} \text{ on } \Gamma_N,\\
\boldsymbol{\sigma}_\tau = \mathbf{0},\ u_n = 0 \text{ on } \Gamma_\pm,\\
\boldsymbol{\sigma}_\tau = \mathbf{0},\ \sigma_n \le 0,\ u_n \le 0,\ \sigma_n u_n = 0 \text{ on } \Gamma_C,\\
\dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \Lambda\dot{\boldsymbol{\sigma}} + \theta_0 \mid \boldsymbol{\sigma}^D \mid^4 \boldsymbol{\sigma}^D \text{ in } \Omega,\\
\mathbf{u}(x,0) = \mathbf{0},\ \ \boldsymbol{\sigma}(x,0) = \mathbf{0} \text{ in } \Omega,
\end{aligned}\right\} \tag{53}$$

with $T = 0.5$ s, $h(t) = 10^8 \dfrac{1+\nu}{E} t + \dfrac{2}{3} 10^{40} \theta_0 t^6$ and

$$\mathbf{g}(x,t) = \begin{cases} t\mathbf{n}, & \text{on } \Gamma_N \cap [x_2 = 0.5], \\ -t\mathbf{n}, & \text{on } \Gamma_N \cap [x_3 = 0.5]. \end{cases}$$

Its solution is readily verifiable

$$\mathbf{u}(x,t) = h(t)\,(0, x_2, -x_3),\ \boldsymbol{\sigma}(x,t) = 10^8 \begin{pmatrix} 0 & 0 & 0 \\ 0 & t & 0 \\ 0 & 0 & -t \end{pmatrix}.$$

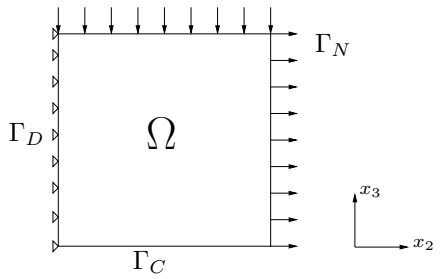

Fig. 2. Computational domain of the first test.

To reduce the size of the problem we consider the corresponding plane strain problem over the symmetry plane $[x_1 = 0]$ (see Figure 2). To solve this problem we use a uniform spatial mesh with 800 elements and 441 nodes, corresponding to a discretization parameter $\Delta x = 0.025$ m. For time discretization we use a small time step ($\Delta t = 10^{-5}$) since the solution is nonlinear in time.

We initialize the multipliers in this way

$$\mathbf{q}^0 = \mathbf{0},\ p^0 = 10^{-5},$$

and consider

$$\lambda_C = \lambda_P = 1/\Delta t,\ \epsilon = 10^{-50},\ \rho = 0.9.$$

The test for convergence is performed on both multipliers. The stopping test on the contact multiplier p is

$$\begin{aligned} |p_k - p_{k-1}| &< \delta \max\{\delta, |\, p_k|\},\ \forall C \in \mathcal{S}_h, \\ |p_k - G_{\lambda_C}(s_k)| &< \delta \max\{\delta, |\, G_{\lambda_C}(s_k)|\},\ \forall C \in \Gamma_{C,k}^+, \end{aligned} \tag{54}$$

and on each component of the viscoplastic multiplier $\mathbf{q}$ is

$$\begin{aligned} |(q_k)_{il} - (q_{k-1})_{il}| &< \delta \max\{\delta, |\, (q_k)_{il}|\},\ \forall K \in \mathcal{T}_h, \\ |(q_k)_{il} - \left(\nabla \Phi_q(\boldsymbol{\sigma}_k^D)\right)_{il}| &< \delta \max\{\delta, |\, \left(\nabla \Phi_q(\boldsymbol{\sigma}_k^D)\right)_{il}|\},\ \forall K \in \mathcal{T}_h, \end{aligned} \tag{55}$$

where δ is a small parameter and $s_k = (u_k)_n + \lambda_C p_k$.

Remark 18 *Notice that, with stopping tests (54) and (55), if the multiplier is close to zero we perform an absolute test with δ^2 and, in the other case, a relative test with δ.*

Firstly, we have tested the NM algorithm for the elastic-viscoplastic law, the (VP) sub-model, considering a Dirichlet condition on Γ_C. Nevertheless, convergence is not always achieved due to large gradient stresses. To overcome this unexpected difficulty, we propose an *adimensionalization* technique on the stresses, which is explained in the next subsection.

6.1.1 Adimensionalization technique

The adimensionalization technique consists of choosing a reference stress η^α and introducing new nondimensional unknowns

$$\boldsymbol{\sigma}_\alpha = \eta^\alpha \boldsymbol{\sigma},\ \mathbf{u}_\alpha = \mathbf{u}$$

and new mechanical coefficients

$$\Lambda_\gamma = \eta^\gamma \Lambda,\ (\theta_0)_\beta = \eta^\beta \theta_0,$$

to solve a similar problem for these new unknowns. Taking into account the constitutive law we can write

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{u}) = \eta^{-(\gamma+\alpha)} \Lambda_\gamma \dot{\boldsymbol{\sigma}}_\alpha + \eta^{-\beta} (\theta_0)_\beta \, \eta^{-5\alpha} \mid \boldsymbol{\sigma}_\alpha^D \mid^4 \boldsymbol{\sigma}_\alpha^D;$$

so, if $\gamma = -\alpha$ and $\beta = -5\alpha$, then $\mathbf{u}_\alpha$ and $\boldsymbol{\sigma}_\alpha$ verify both

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{u}_\alpha) = \Lambda_\gamma \dot{\boldsymbol{\sigma}}_\alpha + (\theta_0)_\beta \mid \boldsymbol{\sigma}_\alpha^D \mid^4 \boldsymbol{\sigma}_\alpha^D,$$

and the equilibrium equations with the corresponding scaling down of the applied forces. Because the typical magnitude of the stresses is of order 10^8 we consider

$$\eta = 10 \text{ and } \alpha = -8.$$

Consequently, the adimensionalized (VP) system to solve is

$$\left.\begin{aligned} \mathrm{Div}(\boldsymbol{\sigma}_\alpha) &= \mathbf{0} \text{ in } \Omega, \\ \mathbf{u}_\alpha &= h\,(t)\,(0, x_2, -x_3) \text{ on } \Gamma_D \cup \Gamma_C, \\ \boldsymbol{\sigma}_\alpha \mathbf{n} &= \mathbf{g} \text{ on } \Gamma_N, \\ (\boldsymbol{\sigma}_\alpha)_\tau &= \mathbf{0},\ (u_\alpha)_n = 0 \text{ on } \Gamma_\pm, \\ \dot{\boldsymbol{\varepsilon}}(\mathbf{u}_\alpha) &= \Lambda_{-\alpha} \dot{\boldsymbol{\sigma}}_\alpha + (\theta_0)_{-5\alpha} \mid \boldsymbol{\sigma}_\alpha^D \mid^4 \boldsymbol{\sigma}_\alpha^D \text{ in } \Omega, \\ \mathbf{u}_\alpha(x,0) &= \mathbf{0},\ \ \boldsymbol{\sigma}_\alpha(x,0) = \mathbf{0} \text{ in } \Omega. \end{aligned}\right\} \tag{56}$$

With this adimensionalization technique convergence is achieved for the NM method (see successive iterants calculated with and without adimensionalization in Figure 3). From now on, we will denote by ANM the NM method together with the adimensionalization technique. The cpu-time, the L^2 relative error in the last time step and the mean of the number of iterations are summarized in table 1, where we can compare the ANM method with the FPM one. Notice that with the ANM algorithm the number of iterations decreases approximately 95% and the cpu-time 85%.

Table 1
The **(VP)** sub-model with $\Delta t = \delta = 10^{-5}$.

Method	cpu-time (s)	$\mathbf{u}$ (m)-error	$\boldsymbol{\sigma}$ (N/m^2)-error	iter.
ANM	2744.0	2.621×10^{-5}	1.244×10^{-5}	3
FPM	18790.2	2.622×10^{-5}	1.245×10^{-5}	57

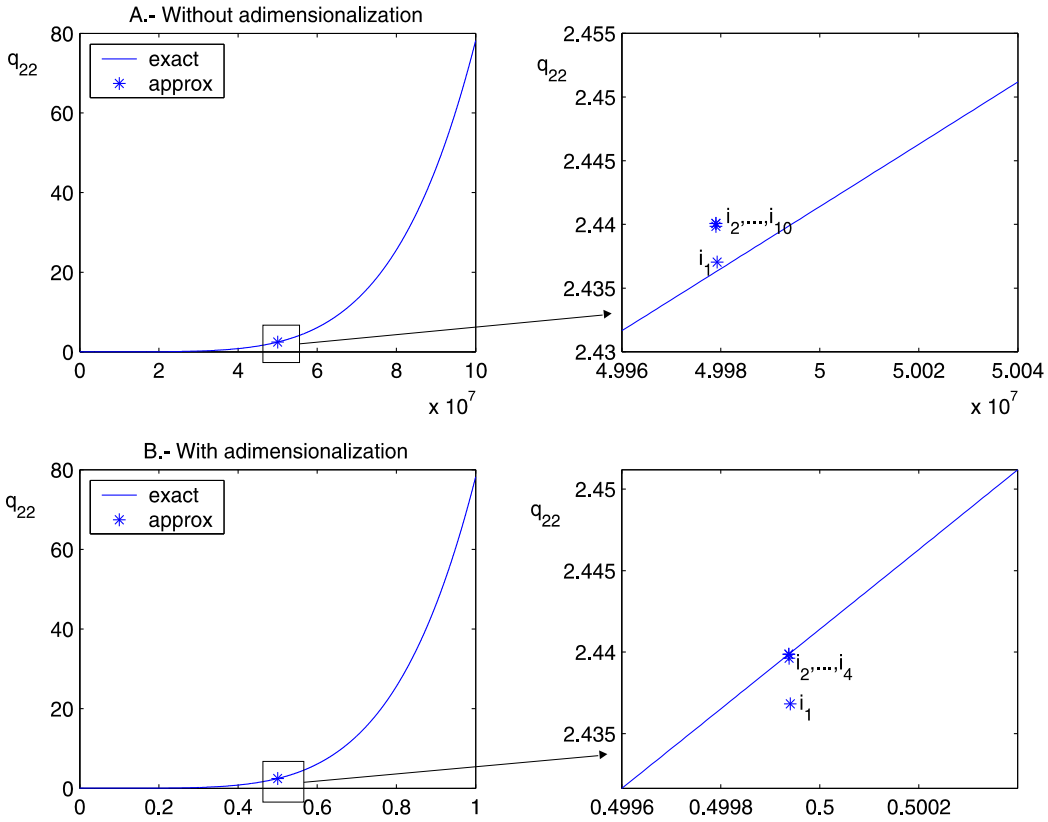

Fig. 3. Numerical approximation of viscoplastic multiplier. The top row of figures without adimensionalization and the second one with the proposed adimensionalization technique. The $*$ shows the successive iterants with the NM algorithm.

Finally, to consider the corresponding (CVP) sub-model, we add the contact condition on Γ_C to the system (56); the results with both algorithms are analogous to those obtained for the (VP) sub-model (see table 2).

Table 2
The **(CVP)** sub-model with $\Delta t = \delta = 10^{-5}$.

Method	cpu-time (s)	$\mathbf{u}$ (m)-error	$\boldsymbol{\sigma}$ (N/m^2)-error	iter.
ANM	5365.2	2.917×10^{-5}	1.345×10^{-5}	3
FPM	25256.8	2.918×10^{-5}	1.345×10^{-5}	69

We notice that the ANM algorithm needs less iterations for convergence, thus less cpu-time, than the FPM algorithm. Nevertheless, the FPM algorithm is more robust and adimensionalization is not necessary.

6.1.2 Increasing the time step

Since the time interval of interest in the aluminium casting is $(0, 1000]$, if we want to employ the NM algorithm in casting simulation we must be able to

increase the time step used in the previous simulations. If we do so, the FPM algorithm works well, but the ANM algorithm does not always converge due to the large gradients. To solve this problem, we use an *optimization technique on the time step* in such a way that, given Δt, if convergence is not achieved, we reduce the time step until the algorithm converges.

Furthermore, to stabilize the ANM algorithm we employ an *Armijo rule* on the viscoplastic algorithm (see [7]). Since the viscoplastic multiplier $\mathbf{q}$ is the fixed point of the equation (38) we define the error function

$$E(\mathbf{q}) = |\mathbf{q} - (\nabla\Phi_q)_{\lambda_P}\left(\boldsymbol{\sigma}^D + \lambda_P\mathbf{q}\right)|^2, \quad \mathbf{q} \in \mathbb{R}^9_s.$$

So, given a time step t^{j+1} and two successive iterants $\mathbf{q}_k^{j+1}, \mathbf{q}_{k+1}^{j+1}$, we have two possibilities:

- If $E(\mathbf{q}_{k+1}^{j+1}) \leq E(\mathbf{q}_k^{j+1})$, the algorithm works well.

- If $E(\mathbf{q}_{k+1}^{j+1}) > E(\mathbf{q}_k^{j+1})$, $\mathbf{q}_{k+1}^{j+1}$ moves far away from the fixed point. To avoid this problem, by using an Armijo rule, we look for a constant ρ such that

$$E(\mathbf{q}_k^{j+1} + \rho(\mathbf{q}_{k+1}^{j+1} - \mathbf{q}_k^{j+1})) \leq E(\mathbf{q}_k^{j+1}). \tag{57}$$

  For that purpose, let us define

$$\phi(\rho) = E(\mathbf{q}_k^{j+1} + \rho(\mathbf{q}_{k+1}^{j+1} - \mathbf{q}_k^{j+1})), \quad \rho \in [0,1].$$

  Then, from (57), ρ must verify

$$\phi(\rho) < \phi(0).$$

  Given $s > 0$, $\beta \in (0,1)$ and $m \in (0,1)$, the Armijo rule consists of choosing

$$\rho_n = \beta^{r_n} s,$$

  where r_n is the first nonnegative integer such that

$$\phi(\rho_n) < \phi(0) + m\,\rho_n\,\phi'(0).$$

  For this test we have considered the parameters

$$s = 0.99,\ \beta = 0.91,\ m = 10^{-5}.$$

Table 3 shows the results obtained with the ANM method together with these two new strategies compared with the FPM method for the (VP) sub-model. Starting with $\Delta t = 0.125$, the ANM method reduces the number of iterations 94% and the cpu-time 66%. Moreover, displacement and stress errors obtained with this method are smaller since the time step is reduced down to 1.5625×10^{-2}.

Table 3
The **(VP)** sub-model with $\Delta t = 0.125$ and $\delta = 10^{-3}$.

Method	cpu-time (s)	$\mathbf{u}$ (m)-error	$\boldsymbol{\sigma}$ (N/m^2)-error	iter.
ANM	2.39	9.325×10^{-2}	2.923×10^{-2}	23
FPM	7.06	5.318×10^{-1}	3.868×10^{-1}	393

Table 4 corresponds to the (CVP) sub-model. If we compare the results, the ANM algorithm obtains smaller displacement and stress errors than the FPM algorithm; in this case, the time step is also reduced down to 1.5625×10^{-2}. Furthermore, the number of iterations with these techniques decreases 95%. Nevertheless, if we analyze the cpu-time it only decreases 26%. This inconveniently small time reduction is due to two factors: on the one hand, the ANM method with the optimization technique on the time step uses 11 time steps, whereas the FPM method only uses 4 steps; on the other hand, although the contact algorithm needs fewer iterations than FPM algorithm, it is necessary to recalculate the stiffness matrix at each iteration (see subsection 4.2.2). So, to reduce the cpu-time we propose numbering the nodes of the mesh in such a way that the higher numbers correspond to contact nodes; this would permit a recalculation of only a portion of the matrix at each iteration.

Table 4
The **(CVP)** sub-model with $\Delta t = 0.125$ and $\delta = 10^{-3}$.

Method	cpu-time (s)	$\mathbf{u}$ (m)-error	$\boldsymbol{\sigma}$ (N/m^2)-error	iter.
ANM	5.79	8.823×10^{-2}	3.008×10^{-2}	20
FPM	7.81	5.935×10^{-1}	3.389×10^{-1}	430

6.2 A growing butt curl

The aim of this test is to check the contact algorithm, the (CP) sub-model, when there is a gap which grows with time between the slab and the rigid foundation (see Figure 5). This gap is similar to the butt curl deformation in a real aluminium casting process.

We consider the same domain as in the previous test. On $\partial\Omega$ we distinguish four parts: $\partial\Omega = \bar{\Gamma}_D \cup \bar{\Gamma}_N \cup \bar{\Gamma}_C \cup \bar{\Gamma}_{\pm}$, where

$$\bar{\Gamma}_D = \partial\Omega \cap ([x_2 = 0] \cup [x_3 = 0.5]), \bar{\Gamma}_N = \partial\Omega \cap [x_2 = 0.5],$$
$$\bar{\Gamma}_C = \partial\Omega \cap [x_3 = 0],\ \bar{\Gamma}_{\pm} = \partial\Omega \cap [x_1 = \pm 1].$$

We assume that the material is elastic with parameters

$$E = 1\text{N/m}^2,\ \nu = 0.35.$$

To abbreviate, let λ, μ be the corresponding Lame coefficients of the material. The (CP) problem to solve is

$$\left.\begin{aligned} -\mathrm{Div}(\boldsymbol{\sigma}) &= \mathbf{f} \text{ in } \Omega, \\ \mathbf{u} &= \widehat{\mathbf{u}} \text{ on } \Gamma_D, \\ \boldsymbol{\sigma}\mathbf{n} &= (0,0,3\mu h^2) \text{ on } \Gamma_N, \\ \boldsymbol{\sigma}_\tau &= \mathbf{0},\ u_n = 0 \text{ on } \Gamma_\pm, \\ \boldsymbol{\sigma}_\tau &= \mathbf{m},\ \sigma_n \le 0,\ u_n \le 0,\ \sigma_n u_n = 0 \text{ on } \Gamma_C, \\ \dot{\boldsymbol{\varepsilon}}(\mathbf{u}) &= \Lambda\dot{\boldsymbol{\sigma}} \text{ in } \Omega, \\ \mathbf{u}(x,0) = \widehat{\mathbf{u}}(x,0),&\quad \boldsymbol{\sigma}(x,0) = \widehat{\boldsymbol{\sigma}}(x,0) \text{ in } \Omega, \end{aligned}\right\}$$

with

$$h(x_2,t) = x_2 - 0.4 + t,$$

$$\mathbf{f}(x,t) = \begin{cases} (0,-3(\lambda+\mu)h^2,-6\mu h x_3), & \text{if } x_2 \le 0.4-t, \\ (0,0,-6\mu h), & \text{if } x_2 > 0.4-t, \end{cases}$$

$$\mathbf{m}(x,t) = \begin{cases} 3h^2\mu(0,-x_3,0), & \text{if } x_2 \le 0.4-t, \\ 3h^2\mu(0,-1,0), & \text{if } x_2 > 0.4-t, \end{cases}$$

$$\widehat{\mathbf{u}}\,(x,t) = \begin{cases} h^3(0,0,x_3), & \text{if } x_2 \le 0.4-t, \\ h^3(0,0,1), & \text{if } x_2 > 0.4-t, \end{cases}$$

$$\widehat{\boldsymbol{\sigma}}(x,t) = \begin{cases} h^3 \begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & \dfrac{3\mu x_3}{h} \\ 0 & \dfrac{3\mu x_3}{h} & \lambda+2\mu \end{pmatrix}, & \text{if } x_2 \le 0.4-t, \\ 3\mu h^2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, & \text{if } x_2 > 0.4-t, \end{cases}$$

and $T = 0.2$ s. We notice that in this test the frictionless condition on Γ_C has been replaced by a nonhomogeneous condition for the tangential stress when there is a gap. Although this condition has not physical meaning it is necessary in order to obtain the analytical solution for a test problem with an

effective and nontrivial contact. The solution of this problem is

$$\mathbf{u} = \widehat{\mathbf{u}},\ \boldsymbol{\sigma} = \widehat{\boldsymbol{\sigma}}.$$

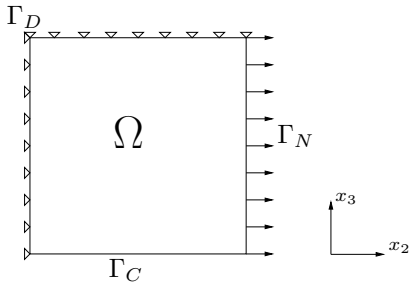

Fig. 4. Computational domain of second test.

As in the previous example, we consider the plane strain problem over the plane $[x_1 = 0]$ (see Figure 4). Since this solution is strongly nonlinear in space, we use a thinner mesh than in the previous test (with 12800 triangles and 6561 vertices). We consider a time step equal to the spatial discretization parameter Δx (6.25×10^{-3}), $\epsilon = 10^{-50}$, $\rho = 0.9$, $\delta = 10^{-3}$, and we initialize

$$p^0 = \begin{cases} 1, \text{ on } (\Gamma_C^+)^0, \\ 0, \text{ on } (\Gamma_C^-)^0. \end{cases}$$

We notice that although the numerical results correspond to $\lambda_C = 1/\Delta t$, the convergence of the NM algorithm is independent of the parameter λ_C.

Figure 5 represents the normal displacement at five different time steps. The displacements on the deformed configuration at four different time steps are represented in Figure 6. Figure 7 shows the Von Mises norm of stresses at the last time step.

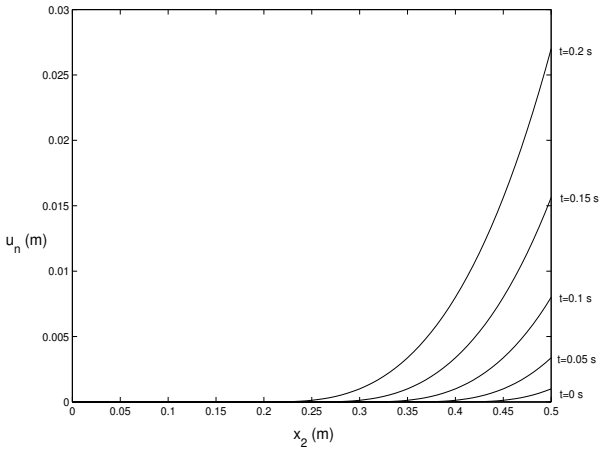

Fig. 5. Butt curl at different time steps.

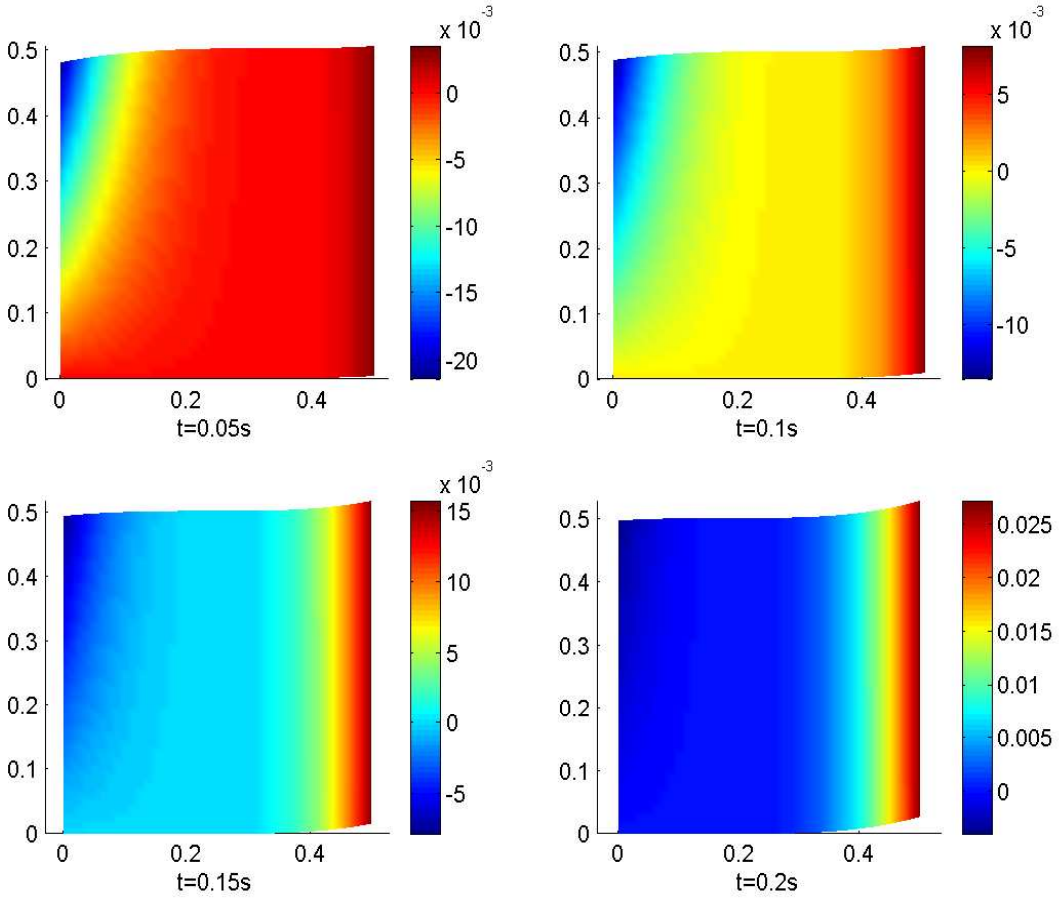

Fig. 6. Displacements on the deformed configuration at four time steps.

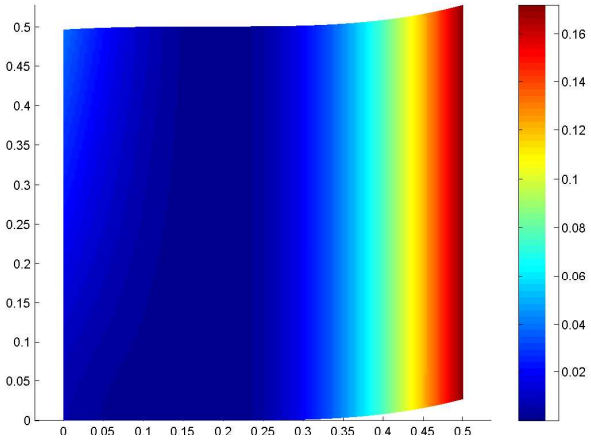

Fig. 7. Von Mises norm of stresses at the last time step.

Table 5 shows the results obtained with the NM method and the FPM method for the (CP) sub-model. We observe that, like in the previous test, the NM algorithm needs less cpu-time than the FPM one. We also notice that the number of iterations decreases approximately 98% whereas the cpu-time decreases only 37% due to the recalculation of the stiffness matrix at each itera-

tion. Furthermore, the NM algorithm improves considerably the fixed point approximation for the contact multiplier.

Table 5
The **(CP)** sub-model with $\Delta t = 6.25 \times 10^{-3}$ and $\delta = 10^{-3}$.

Method	cpu-time (s)	$\mathbf{u}$ (m)-error	$\boldsymbol{\sigma}$ (N/m^2)-error	p error	iter.
NM	435.90	9.051×10^{-4}	9.612×10^{-3}	8.674×10^{-19}	4
FPM	698.76	9.050×10^{-4}	9.613×10^{-3}	2.151×10^{-7}	191

7 Conclusions

In this work we have introduced an algorithm to solve a Signorini contact problem in Maxwell-Norton materials arising from an aluminium casting simulation. To avoid the nonlinearities due to the contact condition and the viscoplastic law, the numerical solution is based on maximal monotone operator techniques involving two multipliers. Following [6] these multipliers can be expressed as fixed points of two nonlinear equations. To approximate these multipliers we propose an iterative algorithm based on Newton's techniques. The efficiency of this algorithm has been compared with the fixed point method algorithm, which was used by the authors in [3,4]. Numerical results show that:

- The fixed point method is slow but very robust.
- The Newton method combined with the adimensionalization technique is fast and accurate. It needs fewer iterations and less cpu-time, and the errors are smaller.
- The main difficulty of the fixed point method is that it is necessary to obtain good parameters to achieve convergence (see [4]). In [19] the authors give an explicit form of the optimal parameters as a function of some constants related to the variational inequality. The calculation of the optimal parameters for the (CVP) problem will be the aim of a next work. Nevertheless, the convergence of the contact algorithm with the Newton method is independent of the algorithm's parameters' values.
- The cpu-time to solve the (CP) problem could be reduced by a modification of the stiffness matrix factorization; this will be the aim of a future work.
- Test examples have shown that if we want to simulate an aluminium casting process with the new algorithm it is necessary to introduce an adimensionalization technique of stresses and a small time step. However, if we employ the new algorithm together with the strategies described in the subsection 6.1.2, we can increase the time step, and, in turn, we can simulate aluminium casting processes using a reasonable cpu-time and obtaining better approximations.

References

[1] P. Barral, Análisis matemático y simulación numérica del comportamiento termomecánico de una colada de aluminio, Ph. D. Thesis, Department of Applied Mathematics, University of Santiago de Compostela, 2001.

[2] P. Barral, A. Bermúdez, M.C. Muñiz, M.V. Otero, P. Quintela and P. Salgado, Numerical simulation of some problems related to aluminium casting, J. Mat. Proc. Tech. 142 (2003) 383-399.

[3] P. Barral and P. Quintela, A numerical algorithm for prediction of thermomechanical deformation during the casting of aluminium alloy ingots, Finite Elem. Anal. Des. 34 (2000) 125-143.

[4] P. Barral and P. Quintela, A numerical method for simulation of thermal stresses during casting of aluminium slabs, Comput. Methods Appl. Mech. Engrg. 178 (1999) 69-88.

[5] P. Barral and P. Quintela, Existence of solution for a contact problem of Signorini type in Maxwell-Norton materials, IMA J. Appl. Math. 67 (2002) 525-549.

[6] A. Bermúdez and C. Moreno, Duality methods for solving variational inequalities, Comput. Math. Appl. 7 (1981) 43-58.

[7] D.P. Bertsekas, Nonlinear Programming (Athena Scientific, 1995).

[8] H. Brezis, Analyse Fonctionnelle. Theorie and Applications (Masson, Paris, 1983).

[9] M. Djaoua and P. Suquet, Évolution quasi-statique des milieux visco-plastiques de Maxwell-Norton, Math. Methods Appl. Sci. 6, 2 (1984) 192-205.

[10] M. Burguera and J. M. Viaño, Numerical solving of frictionless contact problems in perfectly plastic bodies, Comput. Methods Appl. Mech. Engrg. 121 (1995) 303-322.

[11] F. Facchinei and J.S. Pang, Finite-Dimensional Variational Inequalities and complementarity Problems (Springer Verlag, 2003).

[12] A. Friaâ, Le matériau de Norton-Hoff généralisé et ses applications en analyse limite, C.R. Acad. Sci. Paris Ser. A 286 (1978) 953-956.

[13] G. Geymonat and P. Suquet, Functional spaces for Norton-Hoff materials, Math. Methods Appl. Sci. 8 (1986) 206-222.

[14] R. Glowinski and P. Le Tallec, Augmented lagrangian and operator-splitting methods in nonlinear mechanics (SIAM Studies in Applied Mathematics, 1989).

[15] I. R. Ionescu and M. Sofonea, Quasistatic processes for elastic-viscoplastic materials, Quart. Appl. Math. 46 (1988) 229-243.

[16] N. Kikuchi and J.T. Oden, Contact problems in elasticity: a study of variational inequalities and finite element methods (SIAM Studies in Applied Mathematics, 1988).

[17] P.D. Panagiotopoulos, Inequality problems in mechanics and applications: convex and nonconvex energy functions (Birkhuser, 1985).

[18] J.S. Pang, A B-differentiable equation based, globally and locally quadratically convergent algorithm for nonlinear programs, complementary and variational inequality problems, Math. Programming 51 (1991) 101-132.

[19] C. Parés, M. Castro and J. Macías, On the convergence of the Bermúdez-Moreno algorithm with constant parameters, Numer. Math. 92 (2002) 113-128.

[20] S.M. Robinson, Newton's method for a class of nonsmooth equations, Set-Valued Anal. 2 (1994) 291-305.

[21] M. Sofonea, On a contact problem for elastic-viscoplastic bodies, Nonlinear Anal. 29 (1997) 1037-1050.

[22] D. Sun, A further result on an implicit function theorem for locally Lipschitz functions, Oper. Res. Lett. 28 (2001) 193-198.

[23] P. Wriggers, Finite element algorithms for contact problems, Arch. Comput. Methods Engrg. 2, 4 (1995) 1-49.

[24] E. Zeidler, Nonlinear functional analysis and its applications (Springer, 1990).