ESAIM: M2AN 48 (2013) 87–106
DOI: 10.1051/m2an/2013095

**ESAIM: Mathematical Modelling and Numerical Analysis**
www.esaim-m2an.org

# A BERMÚDEZ–MORENO ALGORITHM ADAPTED TO SOLVE A VISCOPLASTIC PROBLEM IN ALLOY SOLIDIFICATION PROCESSES

P. Barral[1], P. Quintela[1] and M.T. Sánchez[2]

**Abstract.** The aim of this work is to present a computationally efficient algorithm to simulate the deformations suffered by a viscoplastic body in a solidification process. This type of problems involves a nonlinearity due to the considered thermo-elastic-viscoplastic law. In our previous papers, this difficulty has been solved by means of a duality method, known as Bermúdez–Moreno algorithm, involving a multiplier which was computed with a fixed point algorithm or a Newton method. In this paper, we will improve the former algorithms by means of a generalized duality method with variable parameters and we will present numerical results showing the applicability of the resultant algorithm to solidification processes. Furthermore, we will describe a numerical procedure to choose a constant parameter for the Bermúdez–Moreno algorithm which gives good results when it is applied to solidification processes.

**Mathematics Subject Classification.** 74C10, 74D10, 65N30.

Received October 2, 2012. Revised May 22, 2013.
Published online November 15, 2013.

## 1. Introduction

In this paper, we present an efficient numerical algorithm to calculate the deformations suffered by a body submitted to a solidification process. A behaviour law frequently used in alloy solidification processes was described in [9,15], where Maxwell–Norton materials with mechanical coefficients strongly dependent on the temperature were considered. Their numerical simulation presents several difficulties related with the nonlinearity of the behaviour law in the regions close to the recently solidified material (see [4]).

In our previous work [3], we have studied and solved a particular solidification problem arising from the mathematical modelling of aluminium casting processes. In this case, two nonlinearities were considered: the previously mentioned one due to the behaviour law and also a Signorini boundary condition due to the contact with the mold. To avoid these nonlinearities, the numerical solution was based on the Bermúdez–Moreno algorithm and the nonlinearities were solved firstly, by using a fixed point method and secondly, by using a generalized Newton method. Both procedures applied to compute the viscoplastic multiplier presented convergence problems in the regions close to the solidification front when they were used to solve a real casting process.

*Keywords and phrases.* Viscoplastic materials, duality methods, solidification process.

[1] Department of Applied Mathematics. Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain. patricia.barral@usc.es; peregrina.quintela@usc.es

[2] Centro Universitario de la Defensa Zaragoza, Academia General Militar, Ctra. Huesca, s/n, 50090 Zaragoza, Spain. tererua@unizar.es

Article published by EDP Sciences

© EDP Sciences, SMAI 2013

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

In this paper we will focus our interest in finding an efficient numerical method to deal with the nonlinear behaviour law. For this, we will propose to consider the Bermúdez–Moreno algorithm presented in [3], taking advantage of the method introduced in [13]. In the latter, the authors introduced a generalized Yosida regularization and they presented a generalization of the Bermúdez–Moreno algorithm that allows the use of very general operators as parameters. Moreover, as a particular case, they analyzed the use of scalar and matrix-valued parameters, proving that the optimal choice of the parameter would be a matrix-valued function. Later, in [1], an application of this generalized algorithm with scalar parameters in one-dimensional problems was presented. Here, we will extend those techniques when the regularized function is tensorial as it occurs with operators associated to viscoplastic behaviour laws. The procedure presented here could be generalized to other solidification problems with a strong dependence on the temperature gradient.

This work is organized as follows. First, in Section 2, we will describe a thermo-elastic-viscoplastic behaviour law frequently used in solidification processes. In this law, the viscoplastic part is given by a nonlinear Maxwell–Norton law and the thermal part is a generalization of Arrhenius law. Section 3 describes the numerical procedure to approximate the viscoplastic multiplier and, so, the solution of the problem. Firstly, in Section 3.1, we will summarize the former results relative to the application of the Bermúdez–Moreno algorithm to the Maxwell–Norton law. In Section 3.2 we will propose to approximate the viscoplastic multiplier by using a fixed point method with variable parameters which are computed automatically. In Section 3.3 the resulting algorithm is presented. Finally, in Section 4 we will apply this numerical algorithm to solve a particular solidification process: an aluminium casting. We will validate the algorithm by using an academic test in order to prove the efficiency of the method and, secondly, we will present the numerical results obtained in the simulation of the considered casting process. Furthermore, we will propose a numerical procedure to compute an optimal constant parameter in an automatic way in order to improve the cpu-time in casting simulations.

## 2. Maxwell–Norton law in solidification processes

Due to the great computational demands of the three-dimensional simulations in solidification processes, in this work we assume that the problem verifies the plane strain assumption and our aim is to solve the corresponding two-dimensional problem in the plane $x_2x_3$. Let $[0,t_f]$ be the time interval to carry out the mechanical simulation and we denote by $\Omega(t)$ the interest domain at the instant $t \in [0,t_f]$. The temperature field, $T(\mathbf{x},t)$, at each point $\mathbf{x} \in \Omega(t)$, is assumed to be previously computed by using the methodology introduced in [2]. There, the thermal problem is modelled by using an enthalpy formulation for a two-phase Stefan problem without considering the convective heat transfer in the fluid phase of the alloy. To take into account the movement of the fluid, the reader is referred to [7].

The mechanical domain at each time instant $t$ corresponds to the solidified part of the domain, denoted by $\Omega_s(t)$, which is obtained from the solution of the thermal problem:

$$\Omega_s(t) = \{\mathbf{x} \in \Omega(t); T(\mathbf{x},t) < T_l\},$$

where $T_l$ is the liquidus temperature (see Fig. 1). From now on, we will suppose that $\Omega_s(t)$ is smooth enough. We denote by $\mathbf{u}(\mathbf{x},t)$ the displacement field and by $\boldsymbol{\sigma}(\mathbf{x},t)$ the stress tensor field at each point $\mathbf{x} \in \Omega_s(t)$ and at each instant $t \in (0,t_f]$.

One of the most used behaviour law in alloy solidification processes was introduced in [9,15]. In these works, the material is supposed to be a nonlinear Maxwell–Norton thermo-elastic-viscoplastic solid and so the strain rate tensor is the superposition of elastic and viscoplastic components together with the thermal strain rate due to the strong temperature gradients during the solidification:

$$\boldsymbol{\varepsilon}(\dot{\mathbf{u}}) = \dot{\boldsymbol{\varepsilon}}^e + \dot{\boldsymbol{\varepsilon}}^{vp} + \dot{\boldsymbol{\varepsilon}}^{th},$$

where the upper dot denotes the usual time derivative and the strain tensor is related to the displacement field by the usual formula

$$\varepsilon_{ij}(\dot{\mathbf{u}}) = \frac{1}{2}(\partial_j \dot{u}_i + \partial_i \dot{u}_j).$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

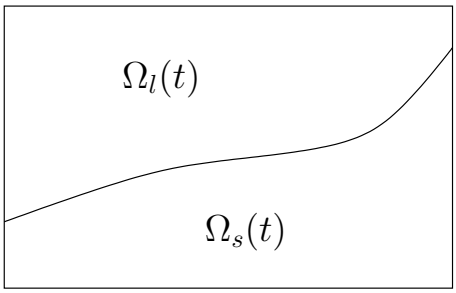



FIGURE 1. Computational domain.

The elastic deformations $\boldsymbol{\varepsilon}^e$ are related to the stress tensor $\boldsymbol{\sigma}$ through Hooke's law with material parameters depending on temperature

$$\boldsymbol{\varepsilon}^e = \Lambda(T)\boldsymbol{\sigma} = \frac{1+\nu(T)}{E(T)}\boldsymbol{\sigma} - \frac{\nu(T)}{E(T)}\mathrm{tr}(\boldsymbol{\sigma})\mathbf{I},$$

where $E$ and $\nu$ denote the Young's modulus and Poisson's ratio, respectively.

The thermal expansion is related to the temperature by a generalized Arrhenius law

$$\dot{\boldsymbol{\varepsilon}}^{th} = \alpha(T)\dot{T}\mathbf{I},$$

where $\alpha$ is the coefficient of thermal expansion, including volume changes due to possible phase transformations.

The viscoplastic part of the strain rate field, $\dot{\boldsymbol{\varepsilon}}^{vp}$, is governed by Norton–Hoff's law (see [12,16]). To describe this law, let $\mathcal{S}_3$ be the space of symmetric second order tensors and let us introduce the dissipation potential $\Phi_q$ defined for each $\boldsymbol{\sigma} \in \mathcal{S}_3$ and $T \in \mathbb{R}$ by

$$\Phi_q(\boldsymbol{\sigma}, T) = \frac{1}{q}\theta(T)|\boldsymbol{\sigma}|^q,$$

where $q \geq 2$ is a fixed real number and $\theta$ denotes a positive material parameter depending on temperature. Norton–Hoff's law may be written as (see [12])

$$\dot{\boldsymbol{\varepsilon}}^{vp} = \nabla_\sigma \Phi_q(\boldsymbol{\sigma}^D, T) = \theta(T) \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D, \tag{2.1}$$

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

$$\boldsymbol{\sigma}^D = \boldsymbol{\sigma} - \frac{1}{3}\mathrm{tr}(\boldsymbol{\sigma})\mathbf{I},$$

and $\nabla_\sigma \Phi_q$ denotes the gradient of $\Phi_q$ with respect to $\boldsymbol{\sigma}$, relative to the usual inner product in $\mathcal{S}_3$. From now on, unless otherwise stated, we will omit the dependence of $\Phi_q$ on the temperature and we will drop the subindex $\sigma$ in the gradient of $\Phi_q$.

Summing up, the constitutive law is formulated as

$$\boldsymbol{\varepsilon}(\dot{\mathbf{u}})(t) = \dot{\overline{(\Lambda(T)\boldsymbol{\sigma})}}(t) + (\nabla\Phi_q(\boldsymbol{\sigma}^D))(t) + \left(\alpha(T)\dot{T}\right)(t)\mathbf{I} \quad \text{in } \Omega_s(t), \quad \text{with } 0 < t \leq t_f. \tag{2.2}$$

In Section 4 we will present a complete mathematical model corresponding to a casting solidification process in which this nonlinearity due to the thermo-elastic-viscoplastic law plays a key role.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

## 3. Bermúdez–Moreno algorithm with variable parameters to solve the thermo-elastic-viscoplastic law

The constitutive law (2.2) is nonlinear due to the viscoplastic term and the dependence of the parameters on the temperature. One possibility to deal with this nonlinearity, used by the authoresses in [4], is based on the well-known Bermúdez–Moreno algorithm involving a viscoplastic multiplier, which is determined as the fixed point of a nonlinear equation (see [6]). A brief description of this method is summarized in Section 3.1, and an optimal choice of the parameters involved in the previous algorithm is deduced in Section 3.2. Finally, in Section 3.3 the resulting algorithm is described.

### 3.1. Former results

Let us consider a mesh of the computational domain $\Omega_s(t)$, denoted by $\mathcal{T}_h(t)$, and a finite element approximation in the usual way: displacements are discretized in space using the Lagrange finite element method of degree one and stresses are assumed constant within each element. To simplify the notation, from now on we will omit the subindex $h$ to denote spatial discretization.

The thermo-elastic-viscoplastic law (2.2) is discretized in time by using an implicit Euler scheme. For that purpose, we discretize the time interval of interest $[0, t_f]$ into $N$ steps: $t^0 = 0$, $t^{j+1} = t^j + \Delta t$, $j = 0, \ldots, N-1$; with $\Delta t = t_f/N$. We denote by $g^j$ an approximation of a given function $g(t)$ at time $t^j$ over $\mathcal{T}_h(t^j)$. Then, the discretized behaviour law is

$$\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}) - \boldsymbol{\varepsilon}(\mathbf{u}^j) = \left(\Lambda(T^{j+1})\boldsymbol{\sigma}^{j+1} - \Lambda(T^j)\boldsymbol{\sigma}^j\right) + \alpha(T^{j+1})(T^{j+1} - T^j)\mathbf{I} + \Delta t\ \nabla\Phi_q\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right) \text{ in } \Omega_s^{j+1}. \quad (3.1)$$

The necessary thermal approach is computed following the methodology described in [2], which consists of an implicit semidiscretization in time, a finite element method for space approximation and a Bermúdez–Moreno algorithm to solve the nonlinearity due to the phase change.

Next, we state two Lemmas which let us rewrite (3.1) as an explicit expression of the stress tensor by introducing a viscoplastic multiplier. The proofs of these Lemmas can be found in [4].

**Lemma 3.1.** *Let $\gamma_p$ be a real positive number. At each time step $t^{j+1}$, $j = 0, \ldots, N-1$, the stress tensor in $\Omega_s^{j+1}$ is given by the relation*

$$\boldsymbol{\sigma}^{j+1} = \mathcal{V}(T^{j+1})\left(\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}) - \Delta t\mathbf{q}^{j+1} + \frac{\Delta t\gamma_p s^{j+1}}{3}\mathrm{tr}(\boldsymbol{\varepsilon}(\mathbf{u}^{j+1}))\mathbf{I} + \mathbf{F}^j\right), \quad (3.2)$$

*where $\mathbf{q}^{j+1}$ is the viscoplastic multiplier at the time step $t^{j+1}$*

$$\mathbf{q}^{j+1} = (\nabla\Phi_q)^{\gamma_p}\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right) = (\nabla\Phi_q - \gamma_p\mathbf{I})\left(\left(\boldsymbol{\sigma}^{j+1}\right)^D\right), \quad (3.3)$$

*and $\mathbf{F}^j$ is the history of the solidified metal up to time $t^j$ together with the expansional effects of the temperature changes at the time interval $[t^j, t^{j+1}]$*

$$\mathbf{F}^j = -\boldsymbol{\varepsilon}(\mathbf{u}^j) + \Lambda(T^j)\boldsymbol{\sigma}^j - \alpha(T^{j+1})(T^{j+1} - T^j)\left(1 + \Delta t\gamma_p s^{j+1}\right)\mathbf{I} + \frac{\Delta t\gamma_p s^{j+1}}{3}\left(\frac{1}{s^j}\mathrm{tr}(\boldsymbol{\sigma}^j) - \mathrm{tr}(\boldsymbol{\varepsilon}(\mathbf{u}^j))\right)\mathbf{I},$$

*where*

$$s^j = \frac{E(T^j)}{1 - 2\nu(T^j)}.$$

*In (3.2), the automorphism $\mathcal{V}(T)$ is defined by*

$$\mathcal{V}(T)\boldsymbol{\tau} = \widetilde{\lambda}(T)\mathrm{tr}(\boldsymbol{\tau})\mathbf{I} + 2\widetilde{\mu}(T)\boldsymbol{\tau},$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

*being*

$$\widetilde{\lambda}(T) = \frac{E(T)\nu(T)}{(L(T)-2\nu(T))(L(T)+\nu(T))}, \quad \widetilde{\mu}(T) = \frac{E(T)}{2(L(T)+\nu(T))},$$

*and* $L(T) = 1 + \gamma_p E(T)\Delta t$.

Notice that in equation (3.3) of Lemma 3.1 the viscoplastic multiplier is obtained by perturbing the viscoplastic law with a positive parameter $\gamma_p$. In the following Lemma the Bermúdez–Moreno method is applied to compute this multiplier.

**Lemma 3.2.** *Let $\gamma_p$ be a real positive number. 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*

$$\mathbf{q}^{j+1} = (\nabla \Phi_q)_{\lambda_p}^{\gamma_p} \left( \left(\boldsymbol{\sigma}^{j+1}\right)^D + \lambda_p \mathbf{q}^{j+1} \right), \tag{3.4}$$

*for $\lambda_p$ a real positive number such that $\lambda_p \gamma_p < 1$. In this equation, $(\nabla \Phi_q)_{\lambda_p}^{\gamma_p}$ denotes the Moreau–Yosida approximation of $(\nabla \Phi_q)^{\gamma_p}$ given by the expression*

$$(\nabla \Phi_q)_{\lambda_p}^{\gamma_p}(\boldsymbol{\zeta}) = \frac{\boldsymbol{\zeta}}{\lambda_p} \left( 1 - \frac{1}{\eta^{j+1}(1-\lambda_p\gamma_p)} \right), \ \boldsymbol{\zeta} \in \mathcal{S}_3, \tag{3.5}$$

*where $\eta^{j+1} = \eta(\boldsymbol{\zeta})$ is the unique root of the equation*

$$\eta^{q-1} - \eta^{q-2} - \frac{\lambda_p \theta(T^{j+1})}{(1-\lambda_p\gamma_p)^{q-1}} |\boldsymbol{\zeta}|^{q-2} = 0, \tag{3.6}$$

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

Notice that equation (3.4) defines $\mathbf{q}$ in an implicit way. To compute it, in [4] fixed point method was proposed. Although the algorithm published in that work is very robust, its convergence turns to be very slow in alloy solidification simulations. During these processes, the large thermal gradients produce great variations in the viscoplastic multiplier close to the liquidus-solidus interphase in such a way that the latter algorithm do not achieve a good convergence.

In order to improve the convergence, in [3] the viscoplastic multiplier was approximated by using Newton techniques together with an adimensionalization of the stress tensor, an Armijo rule and an optimization of the time step. Numerical results in academic tests showed that the previous combination works well in academic examples; however, this method also presented convergence difficulties in alloy solidification simulations.

To overcome this limitation when the algorithm is applied to problems with strong thermal gradients, in the next section we propose to return to the fixed point algorithm to solve equation (3.4), improving it with an optimized computation of the parameters involved therein.

### 3.2. Optimal choice of parameters

Our guess is that reproducing the great variations of the viscoplastic multiplier implicitly defined in (3.4) is very complicated when parameters $\gamma_p, \lambda_p$ are assumed constant in the entire domain and for all time step. So, following [13], in this paper we propose to replace the constant parameters by scalar functions depending not only on each time step but also on each element of the mesh. Although in [13] this choice did not give good results in academic cases and the authors proved that the optimal choice of the parameter should be a matrix-valued function, we will show that in the alloy solidification problems, the convergence improvement is considerable. This fact is due to the viscoplastic behaviour, which varies greatly from one element to another and from one time step to another, explaining the poor convergence results when $\gamma_p$ was constant.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

Therefore, from here on, we consider $\gamma_p$, $\lambda_p$ scalar functions constant on each element $K$ of the mesh and at each time step $t^j$ and we replace (3.4) by

$$\mathbf{q}^j|_K = (\nabla\Phi_q)^{\gamma_p^j|_K}_{\lambda_p^j|_K}\left(\left(\boldsymbol{\sigma}^j|_K\right)^D + \lambda_p^j|_K\mathbf{q}^j|_K\right), \tag{3.7}$$

being

$$\gamma_p^j|_K = \gamma_p(K,t^j), \quad \lambda_p^j|_K = \lambda_p(K,t^j).$$

The main question is to determine a suitable technique to update the value of parameters $\gamma_p^j|_K$, $\lambda_p^j|_K$ from the solution obtained at time step $t^j$. Taking into account that the optimal parameters to convergence verify $\lambda_p^j|_K\gamma_p^j|_K = 1/2$ (see [6,17]), the problem is reduced to calculate, for example, $\gamma_p^j|_K$.

From now on, we focus on how to update $\gamma_p^j|_K$ at each time step $t^j$ and at each element $K$ of the mesh. Taking into account the plane strain assumption in $x_2x_3$, at each element $K \in \mathcal{T}_h(t^j)$, any stress tensor $\boldsymbol{\tau}$ can be written in the matrix form

$$\boldsymbol{\tau}_{|K} = \begin{pmatrix} \tau_{11} & 0 & 0 \\ 0 & \tau_{22} & \tau_{23} \\ 0 & \tau_{23} & \tau_{33} \end{pmatrix} \in \mathcal{S}_3.$$

Since the Norton–Hoff's law (2.2) only involves the deviatoric part of the stress tensor, in this study we consider for each element $K \in \mathcal{T}_h(t^j)$ the space of matrices with null trace under the plane strain assumption:

$$\mathcal{S}^D = \{\boldsymbol{\tau} = (\tau_{ij}) \in \mathcal{S}_3; \tau_{12} = \tau_{13} = 0, \tau_{11} = -(\tau_{22}+\tau_{33})\}.$$

In order to carry out different computations, we introduce the following notation:

- $H$ denotes the application of the gradient of the dissipation potential

$$\begin{aligned} H : \mathcal{S}^D &\longrightarrow \mathcal{S}^D \\ \boldsymbol{\tau} &\longrightarrow H(\boldsymbol{\tau}) = (\nabla\Phi_q)(\boldsymbol{\tau}) = \theta|\boldsymbol{\tau}|^{q-2}\boldsymbol{\tau}. \end{aligned} \tag{3.8}$$

  Here, the dependence of $\theta$ on the temperature is omitted since, in practice, the temperature is constant at each element of the mesh.
- Given $\boldsymbol{\zeta} \in \mathcal{S}^D$ and $\mathbf{q} = H^{\gamma_p}(\boldsymbol{\zeta}) \in \mathcal{S}^D$, we denote $\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}$ as

$$\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q}) = H^{\gamma_p}_{1/(2\gamma_p)}\left(\boldsymbol{\zeta} + \frac{1}{2\gamma_p}\mathbf{q}\right), \tag{3.9}$$

where $H^{\gamma_p}_{1/(2\gamma_p)}$ is the regularized Yosida of the perturbed operator $H^{\gamma_p} = H - \gamma_p\mathbf{I}$.

So, with this notation, the viscoplastic multiplier $\mathbf{q}^j|_K$, given by expression (3.7), can be rewritten at each time step $t^j$ and over each element $K$ as:

$$\mathbf{q}^j|_K = H^{\gamma_p^j|_K}_{1/(2\gamma_p^j|_K)}\left(\left(\boldsymbol{\sigma}^j|_K\right)^D + \frac{1}{2\gamma_p^j|_K}\mathbf{q}^j|_K\right) = \tilde{H}^{\gamma_p^j|_K}_{1/(2\gamma_p^j|_K)}\left(\mathbf{q}^j|_K\right), \tag{3.10}$$

thanks to definition (3.3). Hence, $\mathbf{q}^j|_K$ is obtained as the fixed point of $\tilde{H}^{\gamma_p^j|_K}_{1/(2\gamma_p^j|_K)}$ at each time step and over each element of the mesh.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

Then, the strategy proposed in [13] consists of finding at each element $K$ and at each time step $t^j$ a parameter $\gamma_p^j|_K$ such that

$$D\tilde{H}^{\gamma_p^j|_K}_{1/(2\gamma_p^j|_K)}(\mathbf{q}^j|_K) = \mathbf{0}. \tag{3.11}$$

In order to give a scalar expression for $\gamma_p^j|_K$, we try to minimize $\rho_{\gamma_p^j|_K}$, the spectral radius of $D\tilde{H}^{\gamma_p^j|_K}_{1/(2\gamma_p^j|_K)}(\mathbf{q}^j|_K)$.

To do that, we give the following Lemmas:

**Lemma 3.3.** *Let $\gamma_p$ be a real positive number and $\boldsymbol{\zeta} \in \mathcal{S}^D$. The differential $D\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q})$ is given by the expression*

$$D\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q}) = \left[\mathbf{I} - \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p} DH\left(\boldsymbol{\zeta}\right)\right)^{-1}\right],$$

*with $\mathbf{q} = H^{\gamma_p}\left(\boldsymbol{\zeta}\right)$.*

*Proof.* In [6] it is proved that

$$H^{\gamma_p}_{\lambda_p}(\boldsymbol{\tau}) = \frac{1}{1-\lambda_p\gamma_p} H_{\frac{\lambda_p}{1-\lambda_p\gamma_p}}\left(\frac{1}{1-\lambda_p\gamma_p}\boldsymbol{\tau}\right) - \frac{\gamma_p}{1-\lambda_p\gamma_p}\boldsymbol{\tau}, \quad \boldsymbol{\tau} \in \mathcal{S}^D,$$

for all $\lambda_p, \gamma_p > 0$, $\lambda_p\gamma_p < 1$. In particular, for $\gamma_p\lambda_p = 1/2$ we have that

$$H^{\gamma_p}_{1/(2\gamma_p)}(\boldsymbol{\tau}) = 2H_{\frac{1}{\gamma_p}}\left(2\boldsymbol{\tau}\right) - 2\gamma_p\boldsymbol{\tau}.$$

By using the definition of the Yosida regularization, we obtain that

$$H^{\gamma_p}_{1/(2\gamma_p)}(\boldsymbol{\tau}) = 2\gamma_p\left[\mathbf{I} - \left(\mathbf{I} + \frac{1}{\gamma_p}H\right)^{-1}\right](2\boldsymbol{\tau}) - 2\gamma_p\boldsymbol{\tau} = 2\gamma_p\left[\mathbf{I} - \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p}H\right)^{-1}\right](\boldsymbol{\tau}).$$

Then, applying the chain rule it results

$$DH^{\gamma_p}_{1/(2\gamma_p)}(\boldsymbol{\tau}) = 2\gamma_p\left[\mathbf{I} - \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p}DH(\boldsymbol{\eta})\right)^{-1}\right], \tag{3.12}$$

for

$$\boldsymbol{\eta} = \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p}H\right)^{-1}(\boldsymbol{\tau}), \tag{3.13}$$

or equivalently,

$$\boldsymbol{\tau} = \frac{1}{2}\boldsymbol{\eta} + \frac{1}{2\gamma_p}H(\boldsymbol{\eta}). \tag{3.14}$$

In particular, if we consider in (3.13)

$$\boldsymbol{\tau} = \boldsymbol{\zeta} + \frac{1}{2\gamma_p}\mathbf{q}, \tag{3.15}$$

and taking into account that $\mathbf{q}$ is defined as the perturbed operator of $H$ with parameter $\gamma_p$, $\boldsymbol{\tau}$ can be written as

$$\boldsymbol{\tau} = \frac{1}{2}\boldsymbol{\zeta} + \frac{1}{2\gamma_p}H\left(\boldsymbol{\zeta}\right). \tag{3.16}$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

Then, from (3.14) and (3.16), we deduce that for $\boldsymbol{\tau}$ defined by (3.15) the corresponding $\boldsymbol{\eta}$ is $\boldsymbol{\eta} = \boldsymbol{\zeta}$. Consequently, from (3.9) and (3.12), we infer that

$$D\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q}) = \frac{1}{2\gamma_p} DH^{\gamma_p}_{1/(2\gamma_p)}\left(\boldsymbol{\zeta} + \frac{1}{2\gamma_p}\mathbf{q}\right) = \left[\mathbf{I} - \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p} DH\left(\boldsymbol{\zeta}\right)\right)^{-1}\right],$$

and the Lemma is proved. □

**Lemma 3.4.** *Let $\boldsymbol{\zeta} \in \mathcal{S}^D$. Let us assume that $DH\left(\boldsymbol{\zeta}\right)$ is positive definite with eigenvalues $0 < \omega_1 \leq \omega_2 \leq \omega_3$. Then,*

$$\frac{\omega_i - \gamma_p}{\omega_i + \gamma_p}$$

*are the eigenvalues of $D\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q})$ for $\mathbf{q} = H^{\gamma_p}\left(\boldsymbol{\zeta}\right)$ and its spectral radius is*

$$\rho_{\gamma_p} = \max_{i=1,2,3} \left|\frac{\omega_i - \gamma_p}{\omega_i + \gamma_p}\right|. \tag{3.17}$$

*Proof.* Let $\mu_i$ be an eigenvalue of $D\tilde{H}^{\gamma_p}_{1/(2\gamma_p)}(\mathbf{q})$, then, using Lemma 3.3, there exists $\boldsymbol{\tau} \in \mathcal{S}^D$ such that

$$\left[\mathbf{I} - \left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p} DH\left(\boldsymbol{\zeta}\right)\right)^{-1}\right](\boldsymbol{\tau}) = \mu_i \boldsymbol{\tau}.$$

By doing some simple calculations, we obtain

$$(1 - \mu_i)\left(\frac{1}{2}\mathbf{I} + \frac{1}{2\gamma_p} DH\left(\boldsymbol{\zeta}\right)\right)(\boldsymbol{\tau}) = \boldsymbol{\tau},$$

from where we deduce that

$$DH\left(\boldsymbol{\zeta}\right)(\boldsymbol{\tau}) = \gamma_p \frac{1 + \mu_i}{1 - \mu_i}\boldsymbol{\tau}.$$

Let us denote

$$\omega_i = \gamma_p \frac{1 + \mu_i}{1 - \mu_i}. \tag{3.18}$$

Then, $\omega_i$ is an eigenvalue of $DH\left(\boldsymbol{\zeta}\right)$ and we can write the eigenvalue $\mu_i$ in terms of $\omega_i$ in the following manner:

$$\mu_i = \frac{\omega_i - \gamma_p}{\omega_i + \gamma_p}.$$

□

In [13], the following Lemma is proved:

**Lemma 3.5.** *Under the assumptions of Lemma 3.4, the parameter $\gamma_p^\star$ minimizing $\rho_{\gamma_p}$ is given by:*

$$\gamma_p^\star = \sqrt{\omega_1 \omega_3}. \tag{3.19}$$

In the following, we are going to detail the computation of the eigenvalues of $DH(\boldsymbol{\zeta})$, $\boldsymbol{\zeta} \in \mathcal{S}^D$. It is easy to prove that a basis of the space $\mathcal{S}^D$ is $B_{\mathcal{S}^D} = \left\{\boldsymbol{\tau}^1, \boldsymbol{\tau}^2, \boldsymbol{\tau}^3\right\}$ with

$$\boldsymbol{\tau}^1 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \boldsymbol{\tau}^2 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \boldsymbol{\tau}^3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

**Lemma 3.6.** *Let $\boldsymbol{\zeta} \in \mathcal{S}^D$. The matrix associated to $DH(\boldsymbol{\zeta})$ in terms of the basis $B_{\mathcal{S}^D}$ is:*

$$DH(\boldsymbol{\zeta}) = \begin{pmatrix} AC_{21}\zeta_{22} + B & AC_{31}\zeta_{22} & 2A\zeta_{23}\zeta_{22} \\ AC_{21}\zeta_{33} & AC_{31}\zeta_{33} + B & 2A\zeta_{23}\zeta_{33} \\ AC_{21}\zeta_{23} & AC_{31}\zeta_{23} & 2A\zeta_{23}\zeta_{23} + B \end{pmatrix},$$

*where*

$$A = \theta(q-2)\,|\boldsymbol{\zeta}|^{q-4}, \tag{3.20}$$

$$B = \theta\,|\boldsymbol{\zeta}|^{q-2}, \tag{3.21}$$

$$C_{21} = \zeta_{22} - \zeta_{11},\ C_{31} = \zeta_{33} - \zeta_{11}.$$

*The eigenvalues of $DH(\boldsymbol{\zeta})$ are given by the expressions*

$$\omega_1 = \omega_2 = B, \quad \omega_3 = B + \frac{2}{3}A\zeta_{eq}^2, \tag{3.22}$$

*where $\zeta_{eq}$ is the Von Mises equivalent stress defined by*

$$\zeta_{eq} = \sqrt{\frac{3}{2}\boldsymbol{\zeta} : \boldsymbol{\zeta}}.$$

*Moreover, the eigenvalues verify that $\omega_i \geq 0$, $i = 1, 2, 3$.*

*Proof.* From definition (3.8), it is easy to prove that $DH(\boldsymbol{\zeta})$ is given by the expression

$$DH(\boldsymbol{\zeta})\,\boldsymbol{\tau} = \theta(q-2)\,|\boldsymbol{\zeta}|^{q-4}\,(\boldsymbol{\zeta} : \boldsymbol{\tau})\,\boldsymbol{\zeta} + \theta\,|\boldsymbol{\zeta}|^{q-2}\,\boldsymbol{\tau} = A\,(\boldsymbol{\zeta} : \boldsymbol{\tau})\,\boldsymbol{\zeta} + B\boldsymbol{\tau},$$

for all $\boldsymbol{\tau} \in \mathcal{S}^D$. Then, in order to obtain its associated matrix in terms of the basis $B_{\mathcal{S}^D}$, we calculate the image of each element $\boldsymbol{\tau}^i \in B_{\mathcal{S}^D}$, $i = 1, 2, 3$, and we write them in terms of the basis $B_{\mathcal{S}^D}$:

$$DH(\boldsymbol{\zeta})(\boldsymbol{\tau}^1) = A\,(\zeta_{22} - \zeta_{11})\,\boldsymbol{\zeta} + B\boldsymbol{\tau}^1 = AC_{21}\boldsymbol{\zeta} + B\boldsymbol{\tau}^1 = \begin{pmatrix} AC_{21}\zeta_{11} - B & 0 & 0 \\ 0 & AC_{21}\zeta_{22} + B & AC_{21}\zeta_{23} \\ 0 & AC_{21}\zeta_{23} & AC_{21}\zeta_{33} \end{pmatrix}$$

$$= (AC_{21}\zeta_{22} + B)\,\boldsymbol{\tau}^1 + AC_{21}\zeta_{33}\boldsymbol{\tau}^2 + AC_{21}\zeta_{23}\boldsymbol{\tau}^3,$$

and, analogously,

$$DH(\boldsymbol{\zeta})(\boldsymbol{\tau}^2) = AC_{31}\zeta_{22}\boldsymbol{\tau}^1 + (AC_{31}\zeta_{33} + B)\,\boldsymbol{\tau}^2 + AC_{31}\zeta_{23}\boldsymbol{\tau}^3,$$
$$DH(\boldsymbol{\zeta})(\boldsymbol{\tau}^3) = 2A\zeta_{23}\zeta_{22}\boldsymbol{\tau}^1 + 2A\zeta_{23}\zeta_{33}\boldsymbol{\tau}^2 + (2A\zeta_{23}\zeta_{23} + B)\,\boldsymbol{\tau}^3.$$

By using symbolic calculus, we obtain the eigenvalues of the associated matrix, that are given by expressions (3.22). □

Then, at each time step $t^j$ and at each element $K$ of the mesh, the parameter $\gamma_p^j|_K$ is that minimizing the spectral radius obtained in (3.17) for $\boldsymbol{\zeta} = \left(\boldsymbol{\sigma}^j|_K\right)^D$. Then, from Lemma 3.5, we can conclude the following result:

**Proposition 3.7.** *The optimal choice of parameters for the viscoplastic multiplier $\mathbf{q}^j|_K$ at each time step $t^j$ and over each element $K$ of the mesh is*

$$\gamma_p^j|_K = \sqrt{\omega_1^j|_K\,\omega_3^j|_K}, \quad \lambda_p^j|_K = \frac{1}{2\gamma_p^j|_K},$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

*where* $\omega_1^j|_K$ *and* $\omega_3^j|_K$ *are the eigenvalues of* $DH\left(\left(\boldsymbol{\sigma}^j|_K\right)^D\right)$ *and they are given by*

$$\omega_1^j|_K = B^j|_K, \quad \omega_3^j|_K = B^j|_K + \frac{2}{3}A^j|_K \left(\left(\sigma^j|_K\right)^D\right)_{eq}^2, \tag{3.23}$$

*with*

$$A^j|_K = \theta(T^j|_K)(q-2)\left|\left(\boldsymbol{\sigma}^j|_K\right)^D\right|^{q-4},$$
$$B^j|_K = \theta(T^j|_K)\left|\left(\boldsymbol{\sigma}^j|_K\right)^D\right|^{q-2}.$$

*Proof.* It is easy to prove that $DH\left(\left(\boldsymbol{\sigma}^j|_K\right)^D\right)$ is positive definite, so we can apply Lemma 3.5 and we obtain expression (3.23) for its eigenvalues thanks to Lemma 3.6. □

### 3.3. Bermúdez–Moreno method with variable parameters (VBM)

We propose the following algorithm to solve the behaviour law (3.1):

(1) Given the initial conditions of the problem $T^0$, $\mathbf{u}^0$, $\boldsymbol{\sigma}^0$, we initialize $\mathbf{q}^0 = H\left(\left(\boldsymbol{\sigma}^0\right)^D\right)$, with $H$ defined by (3.8).
(2) At the time step $t^j$, $j \geq 0$, given $(\mathbf{u}^j, \boldsymbol{\sigma}^j, \mathbf{q}^j)$ at each element $K \in \mathcal{T}_h(t^j)$, we compute:
  (a) the parameter $\gamma_p^j|_K$ by using the expressions

$$\gamma_p^j|_K = \sqrt{\omega_1^j|_K\, \omega_3^j|_K}, \tag{3.24}$$
$$\omega_1^j|_K = B^j|_K, \quad \omega_3^j|_K = B^j|_K + \frac{2}{3}A^j|_K \left(\left(\sigma^j|_K\right)^D\right)_{eq}^2, \tag{3.25}$$

  where

$$A^j|_K = \theta(T^j|_K)(q-2)\left|\left(\boldsymbol{\sigma}^j\right)^D|_K\right|^{q-4},$$
$$B^j|_K = \theta(T^j|_K)\left|\left(\boldsymbol{\sigma}^j\right)^D|_K\right|^{q-2},$$
$$\left(\left(\sigma^j|_K\right)^D\right)_{eq} = \sqrt{\frac{3}{2}\left(\boldsymbol{\sigma}^j|_K\right)^D : \left(\boldsymbol{\sigma}^j|_K\right)^D};$$

  (b) the history of the solidified metal $\mathbf{F}^j|_K$ up to time $t^j$ given by Lemma 3.1:

$$\mathbf{F}^j|_K = -\boldsymbol{\varepsilon}(\mathbf{u}^j) + \Lambda(T^j)\boldsymbol{\sigma}^j - \alpha(T^{j+1})(T^{j+1} - T^j)\left(1 + \Delta t \gamma_p^j s^{j+1}\right)\mathbf{I}$$
$$+ \frac{\Delta t \gamma_p^j s^{j+1}}{3}\left(\frac{1}{s^j}\mathrm{tr}(\boldsymbol{\sigma}^j) - \mathrm{tr}(\boldsymbol{\varepsilon}(\mathbf{u}^j))\right)\mathbf{I}, \tag{3.26}$$

  where

$$s^j = \frac{E(T^j)}{1 - 2\nu(T^j)},$$

  and, to simplify the notation, we drop the restriction to $K$ in the right-hand side.
(3) At the time step $t^{j+1}$, with starting values $(\mathbf{u}_0^{j+1}, \boldsymbol{\sigma}_0^{j+1}, \mathbf{q}_0^{j+1}) = (\mathbf{u}^j, \boldsymbol{\sigma}^j, \mathbf{q}^j)$, successive approximations $(\mathbf{u}_k^{j+1}, \boldsymbol{\sigma}_k^{j+1}, \mathbf{q}_k^{j+1})$, $k \geq 1$, of the solution $(\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1}, \mathbf{q}^{j+1})$, can be computed in the following way:
  (a) At each iteration, the displacement field, $\mathbf{u}_k^{j+1}$, is computed by discretizing the corresponding equilibrium equations.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

(b) The updated stress tensor, $\boldsymbol{\sigma}_k^{j+1}$, is defined from Lemma 3.1:

$$\boldsymbol{\sigma}_k^{j+1}|_K = \mathcal{V}(T^{j+1}) \left( \boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) - \Delta t \mathbf{q}_{k-1}^{j+1} + \frac{\Delta t \gamma_p^j s^{j+1}}{3} \mathrm{tr}(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1})) \mathbf{I} + \mathbf{F}^j \right), \tag{3.27}$$

for all $K \in \mathcal{T}_h(t^{j+1})$, where

$$s^{j+1} = \frac{E(T^{j+1})}{1 - 2\nu(T^{j+1})}.$$

(c) The updated viscoplastic multiplier is given by expressions (3.4)–(3.6) for $\lambda_p^j = 1/(2\gamma_p^j)$:

$$\mathbf{q}_k^{j+1}|_K = 2\gamma_p^j \left[ \left( \boldsymbol{\sigma}_k^{j+1} \right)^D + \frac{1}{2\gamma_p^j} \mathbf{q}_{k-1}^{j+1} \right] \left( 1 - \frac{2}{\eta_k^{j+1}} \right), \forall K \in \mathcal{T}_h(t^{j+1}), \tag{3.28}$$

where $\eta_k^{j+1} = \eta \left( \left( \boldsymbol{\sigma}_k^{j+1} \right)^D + \frac{1}{2\gamma_p^j} \mathbf{q}_{k-1}^{j+1} \right)$ is the unique root of the equation

$$\eta^{q-1} - \eta^{q-2} - \frac{2^{q-2}\theta(T^{j+1})}{\gamma_p^j} \left| \left( \boldsymbol{\sigma}_k^{j+1} \right)^D + \frac{1}{2\gamma_p^j} \mathbf{q}_{k-1}^{j+1} \right|^{q-2} = 0,$$

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

Here, all the formulae are also restricted to each element $K$ of the mesh.

## 4. Application to solidification processes in aluminium casting: Academic test and real simulation

In this section, we are going to solve a particular solidification process, the direct chill aluminium casting. Firstly, we describe the mathematical model arising from this process and we propose a numerical algorithm to solve it. This model let us validate the VBM algorithm presented in this work to deal with the thermo-elastic-viscoplastic law. Secondly, in order to prove the efficiency of the method, we compare the results obtained with this algorithm in an academic test with those obtained using the standard Bermúdez–Moreno method (SBM). Finally, we present the numerical results obtained in the simulation of a real casting process.

During casting processes, the liquid aluminium is poured onto a water-cooled mold which is called bottom block; when the aluminium begins to solidify the bottom block starts to descend leaving room for more liquid metal. A detailed description of the complete process can be found in [8, 11, 18].

Let $[0, t_f]$ be the time interval to carry out the mechanical simulation. Due to casting symmetry, $\Omega(t)$ represents a half of the middle section of the slab at the instant $t \in [0, t_f]$. The temperature field $T(\mathbf{x}, t)$ at each point $\mathbf{x} \in \Omega(t)$ is previously computed by using the mathematical model developed in [2]. The mechanical domain at each time instant $t$ corresponds to the solidified part of the slab, denoted by $\Omega_s(t)$, which is obtained from the solution of the thermal problem

$$\Omega_s(t) = \{\mathbf{x} \in \Omega(t); T(\mathbf{x}, t) < T_l\},$$

where $T_l$ is the liquidus temperature (see Fig. 2). We assume that at the initial instant $t = 0$ there exists a solidified part of the slab, that is, $\Omega_s(0) \neq \emptyset$.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

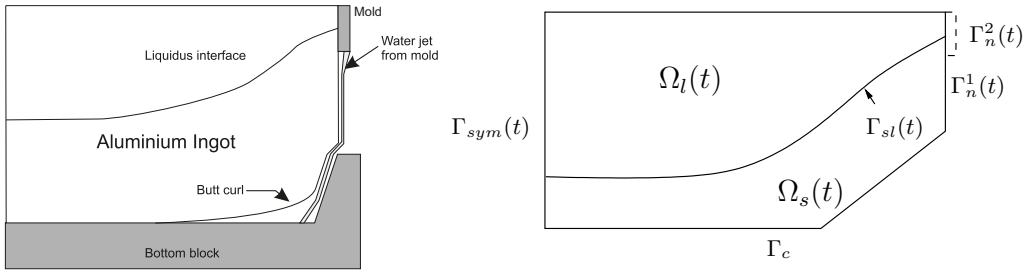



FIGURE 2. Scheme of the casting process.

### 4.1. Mathematical model

The mechanical problem consists of determining the displacement field $\mathbf{u}(\mathbf{x},t)$ and the stress tensor field $\boldsymbol{\sigma}(\mathbf{x},t)$ at each point $\mathbf{x} \in \Omega_s(t)$ and at each instant $t \in (0,t_f]$, such that:

$$-\operatorname{Div}(\boldsymbol{\sigma}) = \mathbf{f} \quad \text{in} \quad \Omega_s(t), \tag{4.1}$$

$$\boldsymbol{\sigma}\mathbf{n} = p_r\mathbf{n} \quad \text{on} \quad \Gamma_{sl}(t), \tag{4.2}$$

$$\boldsymbol{\sigma}\mathbf{n} = \mathbf{0} \quad \text{on} \quad \Gamma_n^1(t), \tag{4.3}$$

$$\boldsymbol{\sigma}_t = \mathbf{0},\ u_n = 0 \quad \text{on} \quad \Gamma_n^2(t) \cup \Gamma_{\text{sym}}(t), \tag{4.4}$$

$$\boldsymbol{\sigma}_t = \mathbf{0},\ \sigma_n \le 0,\ u_n \le 0,\ \sigma_n u_n = 0 \quad \text{on} \quad \Gamma_C, \tag{4.5}$$

$$\boldsymbol{\varepsilon}(\dot{\mathbf{u}}) = \overbrace{(A(T)\boldsymbol{\sigma})}^{\cdot} + \nabla\Phi_q(\boldsymbol{\sigma}^D) + \alpha(T)\dot{T}\,\mathbf{I} \quad \text{in} \quad \Omega_s(t), \tag{4.6}$$

$$\mathbf{u}(0) = \mathbf{u}_0, \quad \boldsymbol{\sigma}(0) = \boldsymbol{\sigma}_0 \quad \text{in} \quad \Omega_s(0). \tag{4.7}$$

Here, we have considered the usual equilibrium equations under the small strains assumption in the quasi static case governed by equation (4.1), where $\mathbf{f}$ denotes the volume forces due to the gravity effect. Equation (4.6) gives the behaviour law of the aluminium which is the considered one in the first part of this work. Equation (4.2) describes the metallostatic pressure exerted by the overlying liquid metal over the upper boundary $\Gamma_{sl}(t)$, which is the isotherm corresponding to the liquidus temperature $T_l$. $\Gamma_C$ is the part of the slab susceptible to be in contact with the bottom block in which we impose the Signorini unilateral frictionless contact condition (4.5). Equation (4.7) gives the initial conditions. Finally, the rest of equations corresponds to the usual symmetry conditions (on $\Gamma_{\text{sym}}(t)$), to the absence of applied forces (on $\Gamma_n^1(t)$) and to the region confined by the mold (on $\Gamma_n^2(t)$).

### 4.2. Weak formulation

Let $p$ be the conjugate of the viscoplastic exponent $q$. We consider the space of displacement fields as

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

and the subset of kinematically admissible displacements at each instant $t$ as

$$\mathbf{U}^p_{\text{ad}}(t) = \{\mathbf{v} \in \mathbf{U}^p(t);\ v_n \le 0 \text{ on } \Gamma_C\},$$

where

$$\mathbf{U}^p(t) = \{\mathbf{v} \in \mathbf{V}^p(t); v_n = 0 \text{ on } \Gamma_n^2(t) \cup \Gamma_{\text{sym}}(t)\}.$$

The corresponding spaces of stress fields are defined as:

$$\mathbf{X}^q(t) = \{\boldsymbol{\tau} = (\tau_{ij});\ \tau_{ij} = \tau_{ji},\ \boldsymbol{\tau}^D \in [L^q(\Omega_s(t))]^9,\ \operatorname{tr}(\boldsymbol{\tau}) \in L^2(\Omega_s(t))\},$$
$$\mathbf{H}^q(t) = \{\boldsymbol{\tau} \in \mathbf{X}^q(t);\ \operatorname{Div}(\boldsymbol{\tau}) \in [L^q(\Omega_s(t))]^3\}.$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

The subset of kinematically admissible stresses at each instant $t$ is

$$\mathbf{H}^q_{\rm ad}(t) = \big\{\boldsymbol{\tau} \in \mathbf{H}^q(t); -{\rm Div}(\boldsymbol{\tau}) = \mathbf{f}(t) \text{ in } \Omega_s(t),\ \boldsymbol{\tau}\mathbf{n} = \mathbf{0} \text{ on } \Gamma_n^1(t),\ \boldsymbol{\tau}\mathbf{n} = p_r(t)\mathbf{n} \text{ on } \Gamma_{sl}(t),$$
$$\boldsymbol{\tau}_t = \mathbf{0} \text{ on } \Gamma_C \cup \Gamma_{\rm sym}(t) \cup \Gamma_n^2(t),\ \tau_n \le 0 \text{ on } \Gamma_C\big\}.$$

In [3], the following variational formulation of Problem $(P)$ is proposed:

**Problem** $(VP)$:
Find $\mathbf{u} \in W^{1,\infty}(0,t_f;\mathbf{U}^p_{\rm ad}(t))$ and $\boldsymbol{\sigma} \in W^{1,\infty}(0,t_f;\mathbf{H}^q(t))$ such that *a.e.* $t \in (0,t_f]$

$$\int_{\Omega_s(t)} \boldsymbol{\sigma}(t) : \boldsymbol{\varepsilon}(\mathbf{v} - \mathbf{u}(t))\,{\rm d}x \ge \int_{\Omega_s(t)} \mathbf{f}(t)\cdot(\mathbf{v}-\mathbf{u}(t))\,{\rm d}x + \int_{\Gamma_{sl}(t)} p_r(t)\mathbf{n}\cdot(\mathbf{v}-\mathbf{u}(t))\,{\rm d}\Gamma, \quad \forall \mathbf{v} \in \mathbf{U}^p_{\rm ad}(t), \tag{4.8}$$

$$\boldsymbol{\varepsilon}(\dot{\mathbf{u}})(t) = \dot{\widetilde{(A(T)\boldsymbol{\sigma})}}(t) + (\nabla\Phi_q(\boldsymbol{\sigma}^D))(t) + \Big(\alpha(T)\dot{T}\Big)(t)\mathbf{I}, \text{ in } \Omega_s(t), \tag{4.9}$$

$$\mathbf{u}(0) = \mathbf{u}_0, \quad \boldsymbol{\sigma}(0) = \boldsymbol{\sigma}_0, \text{ in } \Omega_s(0). \tag{4.10}$$

## 4.3. Numerical solution

The main difficulties we must overcome in the numerical solution of Problem $(VP)$ are the following:

- The solidified part of the slab, which is the computational domain of the mechanical simulation, grows with time. Furthermore, on the upper boundary, which is the isotherm of the liquidus temperature, we must impose the metallostatic pressure due to the weight of the liquid metal.
- To model the butt curl we must solve a contact condition between the slab and the bottom block.
- The aluminium behaviour law is non linear and depends strongly on the temperature field. To solve this difficulty we use the algorithm proposed in Section 3.3.

### *4.3.1. Imposing the metallostatic pressure*

To impose the metallostatic pressure on the interphase we use a fictitious domain method: we model the entire slab, assuming the liquid metal to be a "very weak" elastic material under the action of gravity forces, which therefore does not offer resistance to solid deformations. This methodology has the advantage that remeshing is not needed at each time step to adjust the free boundary, and therefore the numerical implementation is easier. So, we assume that the behaviour law in the liquid domain is given by a Hooke's law

$$\boldsymbol{\sigma}^\epsilon(\mathbf{u}^\epsilon) = A_l^{-1}\boldsymbol{\varepsilon}(\mathbf{u}^\epsilon) = \lambda_l^\epsilon {\rm Div}(\mathbf{u}^\epsilon)\mathbf{I} + 2\mu_l^\epsilon\boldsymbol{\varepsilon}(\mathbf{u}^\epsilon),$$

where $\epsilon$ is assumed to be small enough and the Lamé coefficients of the liquid domain, $\lambda_l^\epsilon$, $\mu_l^\epsilon$ change with different length scale:

$$\lambda_l^\epsilon = \epsilon^\beta\overline{\lambda},\ \mu_l^\epsilon = \epsilon^\alpha\overline{\mu},$$

with $\overline{\lambda}$, $\overline{\mu}$, $\alpha$ and $\beta$ real positive numbers independent of $\epsilon$.

Assuming that $\alpha > \beta$, this has the effect of applying the metallostatic pressure to the solidification front, as it was proved in [5] by using asymptotic expansion techniques.

With this technique the variational inequality (4.8) is defined over the complete slab and the integral over the thermal free boundary $\Gamma_{sl}(t)$ disappears as follows:

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

**Problem** $(EVP)$:
Find $\mathbf{u} \in W^{1,\infty}(0,t_f;\mathbf{U}^p_{\rm ad}(t))$ and $\boldsymbol{\sigma} \in W^{1,\infty}(0,t_f;\mathbf{H}^q(t))$ such that *a.e.* $t \in (0,t_f]$

$$\int_{\Omega(t)} \boldsymbol{\sigma}(t) : \boldsymbol{\varepsilon}(\mathbf{v}-\mathbf{u}(t))\,\mathrm{d}x \geq \int_{\Omega(t)} \mathbf{f}(t)\cdot(\mathbf{v}-\mathbf{u}(t))\,\mathrm{d}x, \forall \mathbf{v} \in \mathbf{U}^p_{\rm ad}(t), \tag{4.11}$$

$$\boldsymbol{\varepsilon}(\dot{\mathbf{u}})(t) = \begin{cases} \dot{\overbrace{(\Lambda_s(T)\boldsymbol{\sigma})}}(t) + (\nabla\Phi_q(\boldsymbol{\sigma}))(t) + \left(\alpha_s(T)\dot{T}\right)(t)\mathbf{I}, \text{ in } \Omega_s(t), \\ \dot{\overbrace{(\Lambda_l(T)\boldsymbol{\sigma})}}(t), \text{ in } \Omega(t)\backslash\Omega_s(t), \end{cases} \tag{4.12}$$

$$\mathbf{u}(0) = \mathbf{u}_0, \quad \boldsymbol{\sigma}(0) = \boldsymbol{\sigma}_0, \text{ in } \Omega(0), \tag{4.13}$$

where the functional spaces are extended in the natural way to the entire domain $\Omega(t)$. From now on, we will consider this new weak formulation.

*4.3.2. Contact condition*

To solve the variational inequality (4.11) due to the contact condition, the authoresses proposed in [4] to use the Bermúdez–Moreno algorithm. Its numerical solution involves a contact multiplier $p$, which is a fixed point of a nonlinear equation, and it is computed by using a fixed point method. The algorithm presented in [4] is robust and converges well for academic tests; nevertheless, the greater the problem's magnitude, the slower its convergence. In solidification processes, where we must join two nonlinearities (contact and viscoplasticity) this difficulty becomes more apparent. In order to improve the accurate and the rate of convergence of the algorithm, in [3] we proposed to compute the contact multiplier by using a generalized Newton method together with a penalization technique to conserve the matrix symmetry; this methodology works well, in particular, when it is combined with the algorithm proposed in Section 3.3 to approach the viscoplastic multiplier.

So, let us apply this method to Problem $(EVP)$. By considering the same spatial and temporal discretization as in Section 3, at the time step $t^{j+1}$, the necessary approximation of the displacement field, $\mathbf{u}_k^{j+1}$, in point (3)(a) of the VBM algorithm is computed as follows:

- Given $\mathbf{u}_{k-1}^{j+1}$ we compute $\mathbf{u}_k^{j+1}$ by solving the equation

$$\int_{\Omega^{j+1}} \boldsymbol{\sigma}_k^{j+1} : \boldsymbol{\varepsilon}(\mathbf{v})\,\mathrm{d}x + \frac{1}{\epsilon_c}\int_{\Gamma^+_{C,k-1}} (u_k^{j+1})_n v_n = \int_{\Omega^{j+1}} \mathbf{f}^{j+1}\cdot\mathbf{v}\,\mathrm{d}x,\ \forall \mathbf{v} \in \mathbf{U}_h(t^{j+1}), \tag{4.14}$$

  where $\boldsymbol{\sigma}_k^{j+1}$ is defined by (3.27). In (4.14) the contact condition at each iteration is imposed by the penalty term with small parameter $\epsilon_c$ on the faces with effective contact defined by:

$$\Gamma^+_{C,k-1} = \left\{ c \in \mathcal{S}_h; (u_{k-1}^{j+1})_n + \lambda_c p_{k-1} > 0 \right\}, \tag{4.15}$$

  $\mathcal{S}_h$ being the triangulation induced by the mesh on the boundary $\Gamma_C$ and $\lambda_c \geq 1$.
- Once the displacement field is calculated, the contact multiplier is updated by

$$p_k = \begin{cases} \dfrac{1}{\epsilon_c}(u_k^{j+1})_n & \text{on } \Gamma^+_{C,k-1}, \\ 0 & \text{on } \Gamma_C \setminus \Gamma^+_{C,k-1}. \end{cases} \tag{4.16}$$

## 4.4. Numerical algorithm to solve the complete casting problem (VBM)

Summing up, assuming again that the temperature field $T$ was previously computed following [7], the proposed algorithm to solve the casting problem remains:

(1) Let $(\mathbf{u}_0,\boldsymbol{\sigma}_0)$ be given and let consider $p^0 = -(\sigma_0)_n$ and $\mathbf{q}^0 = H\left((\boldsymbol{\sigma}_0)^D\right)$.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

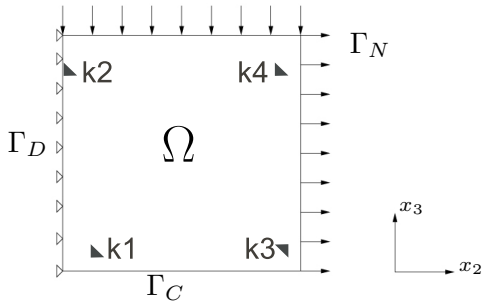



FIGURE 3. Computational domain for the test case.

(2) At the time step $t^j$, $j \geq 0$, given $(\mathbf{u}^j,\ \boldsymbol{\sigma}^j,\ \mathbf{q}^j,\ p^j)$ known at each element $K \in \mathcal{T}_h(t^j)$, we compute:
- The variable parameter $\gamma_p^j|_K$ by using expressions (3.24) and (3.25).
- The history of the solidified metal $\mathbf{F}^j$ up to time $t^j$ by using expression (3.26).

(3) Then, we determine $(\mathbf{u}^{j+1}, \boldsymbol{\sigma}^{j+1}, \mathbf{q}^{j+1}, \gamma_p^{j+1}, p^{j+1})$ at time $t^{j+1}$, an approximated weak solution of Problem $(EVP)$, following the iterative algorithm:

(a) Initialize $\mathbf{u}_0^{j+1} = \mathbf{u}^j$, $\mathbf{q}_0^{j+1} = \mathbf{q}^j$, $p_0^{j+1} = p^j$.

(b) With $\mathbf{q}_{k-1}^{j+1}$, $p_{k-1}^{j+1}$ known, we calculate $(\mathbf{u}_k^{j+1}, p_k^{j+1})$ in two steps:
- Solving the variational equality

$$\int_{\Omega_s^{j+1}} \mathcal{V}(T^{j+1}) \left[ \boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) + \frac{\Delta t \gamma_p^j s^{j+1}}{3} \mathrm{tr}(\boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}))\mathbf{I} \right] : \boldsymbol{\varepsilon}(\mathbf{v}) \mathrm{d}x$$
$$+ \int_{\Omega_l^{j+1}} \Lambda_l^{-1} \boldsymbol{\varepsilon}(\mathbf{u}_k^{j+1}) : \boldsymbol{\varepsilon}(\mathbf{v}) \mathrm{d}x + \frac{1}{\epsilon_c} \int_{(\Gamma_{C,k-1}^+)^{j+1}} \left( u_k^{j+1} \right)_n v_n \mathrm{d}\gamma$$
$$= \int_{\Omega_s^{j+1}} \left[ \mathcal{V}(T^{j+1}) \Delta t \mathbf{q}_{k-1}^{j+1} \right] : \boldsymbol{\varepsilon}(\mathbf{v}) \mathrm{d}x - \int_{\Omega_s^{j+1}} \left[ \mathcal{V}(T^{j+1}) \mathbf{F}^j \right] : \boldsymbol{\varepsilon}(\mathbf{v}) \mathrm{d}x + \int_{\Omega^{j+1}} \mathbf{f}^{j+1} \cdot \mathbf{v} \mathrm{d}x,$$

for all $\mathbf{v} \in \mathbf{U}_h(t^{j+1})$.
- Updating the contact multiplier $p_k^{j+1}$ by using expression (4.16) and the effective contact boundary $(\Gamma_{C,k}^+)^{j+1}$ by expression (4.15).

(c) Finally, the stress tensor $\boldsymbol{\sigma}_k^{j+1}$ and the viscoplastic multiplier $\mathbf{q}_k^{j+1}$ are updated by using expressions (3.27) and (3.28), respectively.

## 4.5. Numerical simulation of casting processes

In this section we are going to show the efficiency of the algorithm introduced in Section 4.4 by carrying out numerical simulations of a test case and a real casting process.

### *4.5.1. Numerical solution of a test case: A problem with large gradients*

In this section we are going to compare and validate the algorithm proposed in this work by applying the numerical code on an academic example. This test has been designed to reproduce a behaviour analogous to the real casting behaviour.

Let $(0, 0.5\mathrm{s}]$ be the time interval and $\Omega$ be the cylindrical body whose axis is parallel to $x_1$-direction and its section the square of dimensions $0.5\mathrm{m}\times 0.5\mathrm{m}$ in the plane $x_2x_3$. We consider the corresponding plane strain problem over the plane $x_2x_3$ and we present the numerical simulation on the two-dimensional domain (see Fig. 3). For the sake of simplicity, we consider $\Omega$ independent of time. On $\partial\Omega$ we distinguish four parts:

$$\partial\Omega = \bar{\Gamma}_D \cup \bar{\Gamma}_N \cup \bar{\Gamma}_C \cup \bar{\Gamma}_{\pm},$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

TABLE 1. Test problem with $\Delta t = 0.0625$.

| Method | SBM | VBM |
|---|---|---|
| cpu-time (s) | 4.20 | 1.15 |
| iterations | 280 | 32 |
| $\mathbf{u}$ (m)-error | $1.88 \times 10^{-1}$ | $2.03 \times 10^{-1}$ |
| $\boldsymbol{\sigma}$ (N/m$^2$)-error | $8.25 \times 10^{-2}$ | $8.37 \times 10^{-2}$ |

where

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

We consider that $\Omega$ is already a solidified block with constant temperature. Then, there are not thermal stresses and the material parameters corresponding to the constitutive law, independent of the temperature $T$, are taken as:

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

Then, the problem to solve is

$$\left.\begin{aligned} \text{Div}(\boldsymbol{\sigma}) = \mathbf{0} \ \text{ in } \ \Omega, \\ \mathbf{u} = h\,(t)\,(0, x_2, -x_3) \ \text{ on } \ \Gamma_D, \\ \boldsymbol{\sigma}\mathbf{n} = 10^8\,\mathbf{g} \ \text{ on } \ \Gamma_N, \\ \boldsymbol{\sigma}_\tau = \mathbf{0},\ u_n = 0 \ \text{ on } \ \Gamma_\pm, \\ \boldsymbol{\sigma}_\tau = \mathbf{0},\ \sigma_n \le 0,\ u_n \le 0,\ \sigma_n u_n = 0 \ \text{ on } \ \Gamma_C, \\ \boldsymbol{\varepsilon}(\dot{\mathbf{u}}) = A\dot{\boldsymbol{\sigma}} + \theta_0 \mid \boldsymbol{\sigma}^D \mid^{q-2} \boldsymbol{\sigma}^D \ \text{ in } \ \Omega, \\ \mathbf{u}(x,0) = \mathbf{0},\ \ \boldsymbol{\sigma}(x,0) = \mathbf{0} \ \text{ in } \ \Omega, \end{aligned}\right\} \tag{4.17}$$

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

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

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 and for time discretization we use a time step $\Delta t = 0.0625$ s.

We compute the viscoplastic multiplier $\mathbf{q}^0 = H\left(\left(\boldsymbol{\sigma}^0\right)^D\right) = \mathbf{0}$ and we consider the following parameters:

- For the standard SBM method $\gamma_p = 0.1 \times 10^{-8}$, $\lambda_p = 5 \times 10^8$;
- For the variable VBM method $\gamma_p^0 = 0.1 \times 10^{-8}$.

Table 1 shows a comparison between the two mentioned algorithms. The reduction in cpu-time (72%) and iterations (88%) of the VBM algorithm with respect to the SBM one is considerable.

In order to take advantage of the computed optimal parameter, we run the academic test with a constant value for $\gamma_p$. This value is chosen as the mean in time of the parameters for $I$ different elements, distributed over the domain, that is:

$$\gamma_p = \frac{1}{I} \sum_{i=1}^{I} \frac{1}{t_f} \left( \sum_{j=1}^{N} \gamma_p^j |_{K_i} \Delta t \right). \tag{4.18}$$

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

TABLE 2. Elastic law parameters.

| $T$ (°C) | $E$ ($10^9$ N/m$^2$) | $\nu$ |
|---|---|---|
| 126.9 | 67 | 0.3134 |
| 326.9 | 58 | 0.3448 |
| 526.9 | 40 | 0.45 |

TABLE 3. Viscoplastic law parameters.

| $\phi$ (N/sm$^2$) | $k$ ($10^6$ N/m$^2$) | $G$ (Kcal/mol) | $q$ |
|---|---|---|---|
| $4.0 \times 10^{12}$ | 15.72 | 37.3 | 6 |

In particular, if we consider $I = 4$ and elements $K_i$ shown in Figure 3, the computed value is $\gamma_p = 0.8 \times 10^{-9}$. With this value for $\gamma_p$, the cpu-time was reduced to 1.12 s and the number of iterations was 35, results analogous to those obtained with the VBM method. These results are reasonable since we take advantage of an optimal value computed for $\gamma_p$ by means of the VBM method and we avoid to recompute $\gamma_p$ at each time step and at each element of the mesh by using the SBM method.

*4.5.2. Numerical simulation of a real casting process*

In this section we are going to present the results obtained in the numerical simulation of a real casting process. We compare the efficiency between the standard Bermúdez–Moreno algorithm (SBM method) introduced in [4] and the algorithm proposed in this work, the variable Bermúdez–Moreno algorithm (VBM method).

The parameters to characterize the thermo-elastic-viscoplastic law of aluminium (see (4.6)) have been introduced in the previous work [4] after carrying out a complete bibliographic search in the engineering literature (see [10, 14, 19]):

- Volume forces: We consider the density function introduced in [10], given in Kg/m$^3$:

$$\rho(T) = \begin{cases} 2700 - 0.23T & \text{if } T \leq T_s, \\ 2360 & \text{if } T = T_l, \end{cases}$$

  where $T_s = 620$ °C and $T_l = 649$ °C. The density values in the interval $(T_s, T_l)$ are computed by linear interpolation.
- Elastic law: Values for Young's modulus $E$ and Poisson's coefficient $\nu$ depending on temperature are obtained from Table 2 by linear interpolation.
- Viscoplastic law: The viscoplastic parameter $\theta(T)$ is computed from the data of Table 3 as

$$\theta(T) = \theta_0 e^{\frac{-G}{R(T+273)}}, \quad \theta_0 = \frac{\phi}{(\sqrt{2}k)^q}.$$

- Thermal law: The coefficient of thermal expansion $\alpha(T)$, which includes volume changes due to phase transformations, is computed from the density function as follows:

$$\alpha(T) = -\frac{1}{3} \frac{\rho_l^{1/3}}{\rho(T)^{4/3}} \frac{\mathrm{d}\rho(T)}{\mathrm{d}T},$$

  where $\rho_l = \rho(T_l)$ is the liquidus density.
- Algorithm parameters: The parameters associated to the algorithms used in this simulation are:
  - For the SBM algorithm, the constant parameter is $\gamma_p = 1.25 \times 10^{-12}$, obtained after carrying out many numerical experiments.
  - For the VBM algorithm, the initial value for the parameter function is $\gamma_p^0 = 1.25 \times 10^{-12}$.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

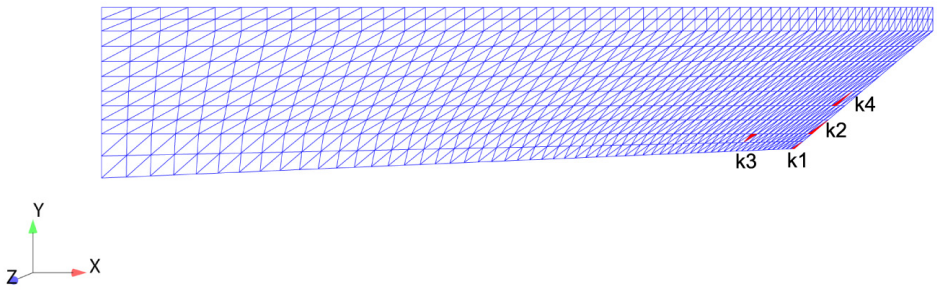



FIGURE 4. Initial mesh.

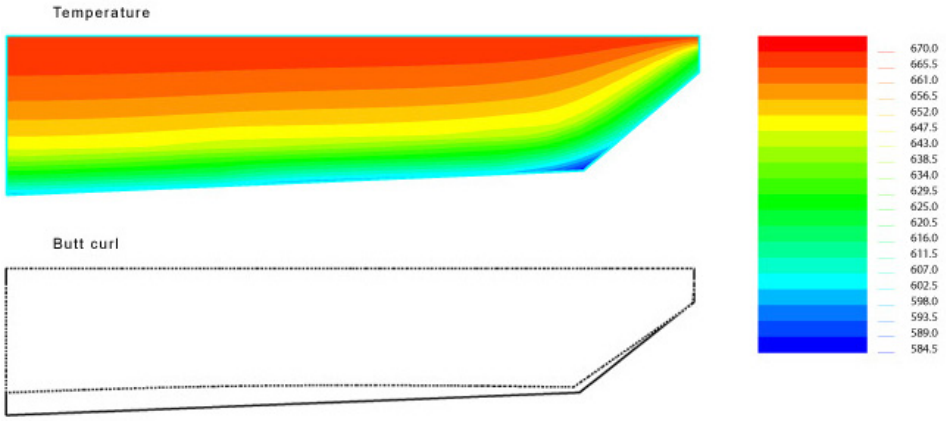



FIGURE 5. Isotherms and butt curl deformation after 145 s.

- Mesh construction: We take into account the peculiarities of casting processes. The contact zone corresponds with the part of the slab rested on the mold. So, the mesh is constructed in such a way that the first nodes correspond to the contact nodes. The slab deformation depends strongly on the thermal gradients, which are larger in the recently solidified zone, that varies with the time. So, the mesh is finest-grained where the thermal gradients are larger. Figure 4 shows the initial mesh of the computational domain. This mesh has 1320 elements and 732 nodes.
- Time discretization: The considered time interval was $(0, 145\ \text{s}]$ and in the numerical solution a time step $\Delta t = 0.01$ s was used.

Figure 5 shows the isotherms and the butt curl obtained at the last time step of the mechanical simulation. The numerical simulation of the problem has been carried out by using the two algorithms: The SBM algorithm has a good but slow convergence which depends strongly on the chosen parameters whereas the VBM algorithm improves the results of the fixed point algorithm in viscoplasticity, computing automatically the parameters.

The cpu-time and the mean of the number of iterations obtained with the SBM and VBM algorithms are summarized in Table 4. With the VBM algorithm the cpu-time decreases approximately 46.73% with respect to the SNM algorithm and the average number of iterations decreases 61.64%.

Figure 6 shows the behaviour of the parameter $\gamma_p$ with respect to time for 4 different elements. These elements were chosen close to the solidification front, two of them on the contact boundary and the other two in the internal part (see Fig. 4). Notice that, when the element is recently solidified, the updated value for $\gamma_p$ presents a sharp change with respect to the initial value. Nevertheless, in the following time steps, this behaviour becomes smoother and the parameter $\gamma_p$ presents an oscillatory variation around the initial value for these 4 elements.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

TABLE 4. Cpu-time and number of iterations after 145 s of casting.

| Method | SBM | VBM |
|---|---|---|
| cpu-time (min) | 435.43 | 231.95 |
| iterations | 451 | 173 |

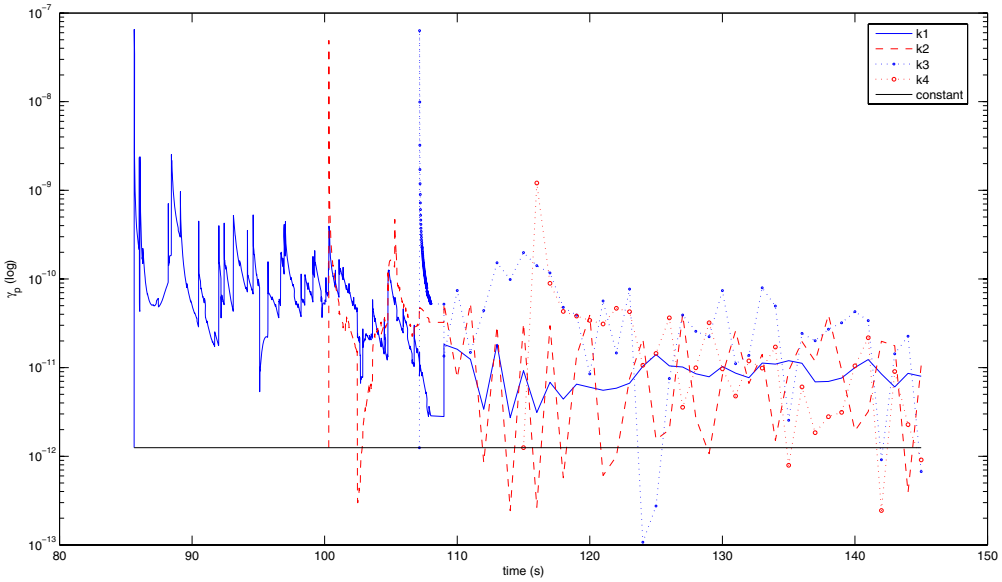



FIGURE 6. Time evolution of parameter $\gamma_p$ for 4 different elements.

During the numerical simulation of this solidification process, we detected that the results obtained from the SBM and VBM methods are significantly different if the time step used during the start stage is not small enough (approximately, $\Delta t = 1$ s). In this case, we obtain results with false convergence in the iterative method. In order to avoid this, we need to reduce the time step in the start stage to $\Delta t = 0.01$ s, in such a way that the results do not show these false convergences.

Furthermore, in order to take advantage of the computed optimal parameter, we run the casting simulation with a constant value for $\gamma_p$, chosen as the mean in time of the parameters following expression (4.18). In particular, if we consider $I = 4$ and elements $K_i$ shown in Figure 4, the computed parameter is $\gamma_p = 5 \times 10^{-10}$. With this value for $\gamma_p$, the cpu-time was reduced to 147 minutes and the average number of iterations was 64, improving the results obtained for the VBM method. Notice that for the manual computation of this parameter it is necessary to carry out many simulations since the parameter magnitude varies greatly from a solidification problem to another. So, this methodology represents an important improvement with respect to the standard algorithm.

## 5. CONCLUSIONS

In this work we have introduced an efficient algorithm (VBM method) to simulate the deformations suffered by a body in a solidification process. To avoid the nonlinearity due to the thermo-elastic-viscoplastic law, the numerical solution is based on maximal monotone operator techniques involving a viscoplastic multiplier. Following [13], we have improved the viscoplastic algorithm introduced in [4] by means of a generalized duality method with variable parameters. The efficiency of this algorithm has been compared with the standard

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press

Bermúdez–Moreno algorithm (SBM method). Numerical results show that:

- The SBM algorithm is strongly dependent on the parameters.
- The VBM algorithm overcomes this difficulty considering the parameters as scalar functions depending on time and space. We propose a methodology to compute the parameters automatically.
- Numerical results in the particular case of aluminium casting show that the VBM algorithm is adequate to compute the deformations suffered by the slab.
- We propose a methodology to compute an optimal constant parameter for this type of problems by means of the two algorithms: first, we carry out a numerical simulation by using the VBM algorithm and we compute a mean value for the parameter $\gamma_p$; secondly, we carry out a numerical simulation with this value for $\gamma_p$ by using the SBM method. This methodology gave good results when it was applied to the aluminium casting process.

*Acknowledgements.* This research was supported by Ministerio de Economía y Competitividad, Spain, under research project MTM2011-23976.

## References

[1] I. Arregui, J.J. Cendán, C. Parés and C. Vázquez, Numerical solution of a 1-d elastohydrodynamic problem in magnetic storage devices. *ESAIM: M2AN* **42** (2008) 645–665.

[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. Mater. Process. Technol.* **142** (2003) 383–399.

[3] P. Barral, C. Moreno, P. Quintela and M.T. Sánchez, A numerical algorithm for a Signorini problem associated with Maxwell-Norton materials by using generalized Newton's methods. *Comput. Methods Appl. Mech. Engrg.* **195** (2006) 880–904.

[4] 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.

[5] P. Barral and P. Quintela, Asymptotic justification of the treatment of a metallostatic pressure type boundary condition in an aluminium casting. *Math. Models Methods Appl. Sci.* **11** (2001) 951–977.

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

[7] A. Bermúdez and M.V. Otero, Numerical solution of a three-dimensional solidification problem in aluminium casting. *Finite Elem. Anal. Des.* **40** (2004) 1885–1906.

[8] J.M. Drezet and M. Plata, Thermomechanical effects during direct chill and electromagnetic casting of aluminum alloys. Part I: Experimental investigation. *Light Metals* (1995) 931–940.

[9] J.M. Drezet, M. Rappaz and Y. Krahenbuhl, Thermomechanical effects during direct chill and electromagnetic casting of aluminum alloys. Part II: numerical simulation. *Light Metals* (1995) 941–950.

[10] T.S. El-Raghy, H.A. El-Demerdash, H.A. Ahmed and A.M. El-Sheikh, Modelling of the transient and steady state periods during aluminium dc casting. *Light Models* (1995) 925–929.

[11] M.C. Flemings, Solidification processing. In McGraw-Hill Series in *Materials Science and Engineering*. McGraw-Hill, New York (1974).

[12] A. Friaâ, Le matériau de Norton-Hoff généralisé et ses applications en analyse limite. *C. R. Acad. Sci. Paris Sér. A-B* **286** (1978) A953–A956.

[13] J.M. Gallardo, C. Parés and M. Castro, A generalized duality method for solving variational inequalities. Applications to some nonlinear Dirichlet problems. *Numer. Math.* **100** (2005) 259–291.

[14] J. Lemaitre and J.L. Chaboche, *Mécanique des matériaux solides*. Dunod, Paris (1988).

[15] S. Mariaux, M. Rappaz, Y. Krahenbuhl and M. Plata, Modelling of Thermomechanical Effects During the Start-Up Phase of the Electromagnetic Casting Process. *Advances in Production and Fabrication of Light Metals and Metal Matrix Composites* (1992) 175–187.

[16] M.C. Naya and P. Quintela, Modelling of materials with long memory. *Int. J. Solids Struct.* **45** (2008) 6133–6156.

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

[18] W. Schneider, E.K. Jensen and B. Carrupt, Development of a new starting block shape for the dc casting of sheet ingots. Part I: Experimental results. *Light Metals* (1995) 961–967.

[19] W.A. Wong and J.J. Jonas, Aluminum extrusion as a thermally activated process. *Trans. Metall. Soc. AIME* **242** (1968) 2271–2280.

https://doi.org/10.1051/m2an/2013095 Published online by Cambridge University Press