

MODELLING OF PHASE CHANGE WITH NON-CONSTANT DENSITY USING XFEM AND A LAGRANGE MULTIPLIER

Dave Martin^{a,b,†}, Hicham Chaouki^{a,b}, Jean-Loup Robert^a, Donald Ziegler^c, Mario Fafard^{a,b}

^aDepartment of Civil and Water Engineering, Laval University, Quebec, QC, G1V 0A6, Canada

^bNSERC/Alcoa Industrial Research Chair MACE³ and Aluminium Research Centre - REGAL, Laval University, Quebec, QC, G1V 0A6, Canada

^cAlcoa Primary Metals, Alcoa Technical Center, 100 Technical Drive, Alcoa Center, PA, 15069-0001, USA

ABSTRACT

A two phase model for two-dimensional solidification problems with variable densities was developed by coupling the Stefan problem with the Stokes problem and applying a mass conserving velocity condition on the phase change interface. The extended finite element method (XFEM) was used to capture the strong discontinuity of the velocity and pressure as well as the jump in heat flux at the interface. The melting temperature and velocity condition were imposed on the interface using a Lagrange multiplier and the penalization method, respectively. The resulting formulations were then coupled using a fixed point iteration algorithm. Three examples were investigated and the results were compared to numerical results coming from a commercial software using ALE techniques to track the solid/liquid interface. The model was able to reproduce the benchmark simulations while maintaining a sharp phase change interface and conserving mass.

Keywords: Phase change, XFEM, Lagrange multiplier, Non-constant density

1. INTRODUCTION

The finite element method Reddy (2006) has been extensively studied and successfully used in a wide variety of scenarios involving continuous media but particular situations still offer a challenge, such as material and geometrical discontinuities. This makes the finite element method ill suited to solve problems involving discontinuities that are part of the solution or moving with time. The Stefan problem Nedjar (2002); Beckermann *et al.* (1999); Helenbrook (2013); Özişik (1993) for the isothermal solidification or melting of a material is one such situation because of the discontinuous heat flux at the phase change interface. The problem is further complicated by the fact that most materials undergo a change in density at the interface, effectively adding a mass flux boundary condition. Luckily, the density variation in most materials is small and can be neglected Morgan (1981); Postek *et al.* (2008). For certain materials however, the density variation may be quite important, up to 25%. Furthermore, many practical applications involve following the total volume of the material or mass flow. In these situations, neglecting the density change significantly hinders the use of numerical models.

To handle such problems involving discontinuities, the extended finite element method Belytschko *et al.* (2001); Dolbow *et al.* (2000); Belytschko *et al.* (2009) was developed, based on the partition of unity method Babuska and Melenk (1997); Dolbow *et al.* (2000); Melenk and Babuska (1996). Using carefully selected functions $\psi(\mathbf{x}, t)$, the technique adds degrees of freedom that will “enrich” the interpolation and allows the solution to adopt a non-linear (discontinuous) behavior. The particular type of behavior is determined by the enrichment function $\psi(\mathbf{x}, t)$, known *a priori*. Only nodes having support cut by the interface have a modified behavior must be enriched. Consequently, the additional com-

putational costs are local to the interface. The interface geometry is stored and transported in a computationally efficient manner, most commonly using the level set method Osher and Sethian (1988); Osher and Fedkiw (2001).

Numerous extended finite element models for the solutions of the classical (diffusive) Stefan problem are found in the literature Chessa *et al.* (2002); Bernauer and Herzog (2011); Merle and Dolbow (2002); Ji *et al.* (2002). More complex models involving convection with constant density have also been developed using different numerical techniques Zabaras *et al.* (2006); Vynnycky and Kimura (2007); Brent *et al.* (1988). Particularly, a fully XFEM Stefan/Navier-Stokes model was used by Martin (2016). Models including the density variation are more uncommon Yoo and Tack (1991). A straight-forward strategy to include the non-constant material densities is to use a moving-mesh algorithm such as the one found in the commercial code Comsol. This algorithm defines the phase change interface on a set of nodes, allowing the mass conservation boundary condition to be easily applied. However, the moving mesh adds considerable computational costs caused by the increase in degrees of freedom of the overall problem and the remeshing procedure required when the mesh becomes too distorted. These costs may hinder the use of a moving mesh algorithms in large scale multi-physical simulations which often have a large amount of degrees of freedom.

In the work presented here, a new coupled formulation using the extended finite element method for both the Stefan problem and Stokes equations based on Martin (2016) is developed for the case of variable phase densities. A fixed point iteration scheme is then used to obtain a converged solution for a given time step. The conservation of mass at

[†]Corresponding author. Email: dave.martin.1@ulaval.ca

the phase change interface is handled by applying a velocity boundary condition.

The paper is divided as follows. The governing equations for the Stefan and Stokes problems are described in section 2. The finite element formulation, level set problem and details concerning the interface movement and extended finite element method are described in section 3. Benchmark examples are then solved in section 4 to validate the algorithm. To this end, the commercial finite element simulation software Comsol was used with a moving mesh algorithm to capture the interface movement. Finally, the paper ends with some concluding remarks.

2. GOVERNING EQUATIONS

2.1. Stefan Problem Formulation

Consider a domain Ω with an initial temperature $T(x, t_0)$ and interface Γ separating solid (Ω_s) and liquid (Ω_l) phases with different thermal properties and densities. We suppose that the material has an isothermal phase change at some melting temperature T_m . Applying the conservation of energy in Ω results in the following equations Özişik (1993):

$$(\rho c_p)_s \frac{\partial T}{\partial t} - \nabla \cdot (k_s \nabla T) = 0 \quad \mathbf{x} \in \Omega_s \quad (1a)$$

$$(\rho c_p)_l \left(\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T \right) - \nabla \cdot (k_l \nabla T) = 0 \quad \mathbf{x} \in \Omega_l \quad (1b)$$

$$T - T_m = 0 \quad \mathbf{x} \in \Gamma \quad (1c)$$

$$T = \hat{T} \quad \mathbf{x} \in \Gamma_D \quad (1d)$$

$$-k \nabla T \cdot \mathbf{n} = \hat{q} \quad \mathbf{x} \in \Gamma_N \quad (1e)$$

where c_p is the specific heat, k the thermal conductivity, ρ the density, \mathbf{v} the liquid phase velocity. Subscripts l and s indicate liquid and solid phases, respectively. Additionally, the melting temperature is applied on the solid-liquid interface (1c). Dirichlet and Neumann boundary conditions away from the interface are applied on $\partial\Omega = \Gamma_N \cup \Gamma_D$ as usual (1d, 1e).

Conservation of energy at the interface requires that the jump in heat flux normal to the interface (caused by the imposition of the melting temperature) be related to the rate of solidification or melting of the material as described in Özişik (1993):

$$[-k \nabla T] \cdot \mathbf{n}_\Gamma = (k_l \nabla T_l - k_s \nabla T_s) \cdot \mathbf{n}_\Gamma = \rho_s L v_\Gamma \quad \mathbf{x} \in \Gamma \quad (2)$$

where L is the latent heat and v_Γ the normal interface velocity Özişik (1993). The normal vector \mathbf{n}_Γ points from the liquid to solid phase, meaning that the interface velocity is positive for melting and negative for solidification.

Tracking the moving interface is done using the level set method Osher and Fedkiw (2001); Osher and Sethian (1988); Osher and Fedkiw (2003). The principle behind this method is to introduce a signed distance function to the interface, $\phi(\mathbf{x}, t)$, defined as follows Osher and Fedkiw (2003):

$$\phi(\mathbf{x}, t) = \min_{\mathbf{x}_\Gamma \in \Gamma} |\mathbf{x} - \mathbf{x}_\Gamma(t)| \text{sign}(\mathbf{n}_\Gamma \cdot (\mathbf{x} - \mathbf{x}_\Gamma(t))) \quad \mathbf{x} \in \Omega \quad (3)$$

The interface is then easily identified as the set of points where $\phi(\mathbf{x}, t) = 0$. In this work, the level set field is constructed so that the liquid phase is on the positive side of the interface (i.e. $\mathbf{x} \in \Omega_l$ if $\phi(\mathbf{x}, t) > 0$).

2.2. Stokes Problem formulation

In the present study, the liquid phase velocity \mathbf{v} is governed by the Stokes problem for viscous incompressible fluids:

$$\rho_l \frac{\partial \mathbf{v}}{\partial t} = \nabla \cdot \boldsymbol{\sigma} \quad \mathbf{x} \in \Omega_l \quad (4a)$$

$$\nabla \cdot \mathbf{v} = 0 \quad \mathbf{x} \in \Omega_l \quad (4b)$$

$$\mathbf{v} = \frac{\rho_l - \rho_s}{\rho_l} v_\Gamma \mathbf{n}_\Gamma \quad \mathbf{x} \in \Gamma \quad (4c)$$

$$\mathbf{v} = \hat{\mathbf{v}} \quad \mathbf{x} \in \Gamma_D \quad (4d)$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \hat{\boldsymbol{\sigma}} \quad \mathbf{x} \in \Gamma_N \quad (4e)$$

$$\boldsymbol{\sigma} = -p \mathbf{I} + 2\mu D(\mathbf{v}) \quad (4f)$$

$$D(\mathbf{v}) = \frac{1}{2} (\nabla \mathbf{v} + \nabla \mathbf{v}^T) \quad (4g)$$

where p is the pressure, μ the viscosity and $D(\mathbf{v})$ the rate of deformation tensor. The convection term in the complete Navier-Stokes equations was neglected, leading to two linear systems of equations for the heat transfer and fluid flow problems. The only non-linearity is in the coupling terms between the two problems: the convective heat transfer and interface velocity.

The variation in density between the solid and liquid phases creates a mass flux at the interface, which is a function of the interface velocity and specific phase densities (equation (4c)). The other physical properties are assumed constant. The initial velocity field $\mathbf{v}(\mathbf{x}, t_0)$ is assumed divergence-free with a given initial pressure field $p(\mathbf{x}, t_0)$. Dirichlet and Neumann boundary conditions away from the interface are applied on $\partial\Omega = \Gamma_N \cup \Gamma_D$ as usual (4d, 4e).

2.3. Enriched Interpolation Scheme

The phase change problem is characterized by the jump in the heat flux which is caused by the temperature gradient discontinuity. However, the application of the interface boundary condition (1c) implies that the temperature is continuous at the interface. Such a weak discontinuity can be handled using the extended finite element method. To this end, the following interpolation scheme is used Chessa et al. (2002):

$$T(\mathbf{x}, t) = \sum_{i \in I} N_i^T(\mathbf{x}) T_i(t) + \sum_{j \in J} N_j^T(\mathbf{x}) \psi_j^T(\mathbf{x}, t) T_j^*(t) \quad (5a)$$

$$\psi_j^T(\mathbf{x}, t) = |\phi(\mathbf{x}, t)| - |\phi(\mathbf{x}_j, t)| \quad (5b)$$

where N^T are the standard interpolation functions, T_i and T_j^* the standard and enriched degrees of freedom, respectively, and $\psi_j^T(\mathbf{x}, t)$ the enrichment function, based on the absolute value of the level set field. A more compact way to write expression (5) is by using the standard matrix form

$$T(\mathbf{x}, t) = \langle N_T \rangle \{T\} \quad (6a)$$

$$\langle N_T \rangle = \langle N_1^T, \dots, N_{n_I}^T, N_1^T \psi_1^T, \dots, N_{n_J}^T \psi_{n_J}^T \rangle \quad (6b)$$

$$\{T\} = \langle T_1, \dots, T_{n_I}, T_1^*, \dots, T_{n_J}^* \rangle^T \quad (6c)$$

where n_I and n_J are the number of standard and enriched nodes, respectively. When using (5) special attention must be given to elements containing enriched nodes that are not cut by the interface, called blending elements. A modified interpolation scheme must be used in these elements to maintain an optimal convergence rate, as described in Fries (2008); Shibamura and Utsunomiya (2009).

In order to capture the jump in the heat flux at the interface, a Lagrange multiplier λ is used Gerstenberger (2010). The interpolation scheme

for the Lagrange multiplier is given by:

$$\lambda(\mathbf{x}, t) = \sum_{i \in I} N_i^\lambda(\mathbf{x}) \lambda_i(t) + \sum_{j \in J} N_j^\lambda(\mathbf{x}) \psi_j^\lambda(\mathbf{x}, t) \lambda_j^*(t) \quad (7a)$$

$$\psi_j^\lambda(\mathbf{x}, t) = H(\phi(\mathbf{x}, t)) - H(\phi(\mathbf{x}_j, t)) \quad (7b)$$

$$H(\mathbf{x}, t) = \begin{cases} 1 & \text{if } \phi(\mathbf{x}, t) < 0 \\ 0 & \text{if } \phi(\mathbf{x}, t) > 0 \end{cases} \quad (7c)$$

where H is the Heaviside function.

Following (6), the Lagrange multiplier may be rewritten in two dimensions as

$$\begin{aligned} \lambda(\mathbf{x}, t) &= [N_\lambda] \langle \lambda \rangle^T \\ [N_\lambda] &= \begin{bmatrix} N_1^\lambda & \dots & N_{n_I}^\lambda & N_1^\lambda \psi_1^\lambda & \dots & N_{n_J}^\lambda \psi_{n_J}^\lambda & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & N_1^\lambda & \dots & N_{n_I}^\lambda & N_1^\lambda \psi_1^\lambda & \dots & N_{n_J}^\lambda \psi_{n_J}^\lambda \end{bmatrix} \\ \langle \lambda \rangle &= \langle \lambda_1^x, \dots, \lambda_{n_I}^x, \lambda_1^*, \dots, \lambda_{n_J}^*, \lambda_1^y, \dots, \lambda_{n_I}^y, \lambda_1^{y*}, \dots, \lambda_{n_J}^{y*} \rangle \end{aligned}$$

where $[N_\lambda]$ is the matrix of interpolation functions.

The Navier-Stokes equations are valid (and solved) in the liquid phase only. For this purpose, the fluid-structure interaction approach, proposed in Gerstenberger and Wall (2008), is used. Therefore, the velocity and pressure fields can be interpolated using the following scheme:

$$\mathbf{v}(\mathbf{x}, t) = \sum_{i \in I} N_i^v(\mathbf{x}) \psi^v(\mathbf{x}, t) \mathbf{v}_i(t) \quad (9a)$$

$$p(\mathbf{x}, t) = \sum_{i \in I} N_i^p(\mathbf{x}) \psi^v(\mathbf{x}, t) p_i(t) \quad (9b)$$

$$\psi^v(\mathbf{x}, t) = \begin{cases} 1 & \text{if } \phi(\mathbf{x}, t) > 0 \\ 0 & \text{if } \phi(\mathbf{x}, t) < 0 \end{cases} \quad (9c)$$

Following (6) and (2.3), the velocity and pressure fields may be rewritten as:

$$\mathbf{v}(\mathbf{x}, t) = [N_v] \{ \mathbf{v} \} \quad (10a)$$

$$p(\mathbf{x}, t) = \langle N_p \rangle \{ p \} \quad (10b)$$

$$[N_v] = \begin{bmatrix} N_1^v \psi_1^v & \dots & N_{n_I}^v \psi_{n_I}^v & 0 & \dots & 0 \\ 0 & \dots & 0 & N_1^v \psi_1^v & \dots & N_{n_I}^v \psi_{n_I}^v \end{bmatrix} \quad (10c)$$

$$\{ \mathbf{v} \} = \langle v_1^x, \dots, v_{n_I}^x, v_1^y, \dots, v_{n_I}^y \rangle^T \quad (10d)$$

$$\langle N_p \rangle = \langle N_1^p \psi_1^v, \dots, N_{n_I}^p \psi_{n_I}^v \rangle \quad (10e)$$

$$\{ p \} = \langle p_1, \dots, p_{n_I} \rangle^T \quad (10f)$$

According to this interpolation scheme, the solid part of the domain is ignored. Also, enriched degrees of freedom are not required because no new information (behavior) is introduced. All velocity and pressure degrees of freedom whose support is completely inside the solid domain are removed from the system of equations.

3. NUMERICAL IMPLEMENTATION

3.1. Stefan Problem

The weak form of the energy conservation equations (1a,1b) is

$$\int_{\Omega} \delta T \rho c_p \frac{\partial T}{\partial t} d\Omega + \int_{\Omega_l} \delta T \rho_l c_p \mathbf{v} \cdot \nabla T d\Omega + \int_{\Omega} \nabla \delta T k \nabla T d\Omega - \int_{\Gamma} \delta T \lambda \cdot \mathbf{n}_{\Gamma} d\Gamma = 0 \quad (11a)$$

$$\int_{\Omega} \delta \lambda \cdot \left(\frac{1}{k} \lambda + \nabla T \right) d\Omega - \int_{\Gamma} \delta \lambda \cdot \mathbf{n}_{\Gamma} (T - T_m) d\Gamma = 0 \quad (11b)$$

where δT and $\delta \lambda$ are the test functions. The method used in this work to impose the melting temperature on the interface is the stable Lagrange

multiplier used in Martin *et al.* (2016) and originally developed in Gerstenberger and Wall (2010); Baiges *et al.* (2012). The Lagrange multiplier is defined as a vectorial flux and interpolated on the same mesh as the temperature field. The projection of this secondary variable on the interface is then used as a scalar Lagrange multiplier to impose the melting temperature. The Neumann boundary condition has been omitted for the sake of clarity.

Using a backward Euler scheme for the time derivative of T in (11) gives Fries and Zilian (2009):

$$\int_{\Omega} \delta T^{n+1} \frac{(\rho c_p T)^{n+1} - (\rho c_p T)^n}{\Delta t} d\Omega + \int_{\Omega_l} \delta T^{n+1} (\rho c_p)^{n+1} \mathbf{v}^{n+1} \cdot \nabla T^{n+1} d\Omega + \int_{\Omega} \nabla \delta T^{n+1} k^{n+1} \nabla T^{n+1} d\Omega - \int_{\Gamma} \delta T^{n+1} \lambda^{n+1} \cdot \mathbf{n}_{\Gamma} d\Gamma = 0 \quad (12a)$$

$$\int_{\Omega} \delta \lambda^{n+1} \cdot \left(\frac{1}{k} \lambda^{n+1} + \nabla T^{n+1} \right) d\Omega - \int_{\Gamma} \delta \lambda^{n+1} \cdot \mathbf{n}_{\Gamma} (T^{n+1} - T_m) d\Gamma = 0 \quad (12b)$$

where n indicates the previous time step.

After replacing T and λ with their approximations we obtain the system of equations

$$\begin{bmatrix} \frac{1}{\Delta t} [M] + [C] + [K] & -[L] \\ [Q] - [L]^T & [M_\lambda] \end{bmatrix} \begin{Bmatrix} \{T\}^{n+1} \\ \{\lambda\}^{n+1} \end{Bmatrix} = \begin{bmatrix} \frac{1}{\Delta t} [M]^* & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \{T\}^n \\ \{\lambda\}^n \end{Bmatrix} - \begin{Bmatrix} 0 \\ \{\mathbf{f}_\lambda\} \end{Bmatrix} \quad (13a)$$

$$[M] = \sum_e \int_{\Omega^e} \{N_T\}^{n+1} (\rho c_p)^{n+1} \langle N_T \rangle^{n+1} d\Omega \quad (13b)$$

$$[M]^* = \sum_e \int_{\Omega^e} \{N_T\}^{n+1} (\rho c_p)^n \langle N_T \rangle^n d\Omega \quad (13c)$$

$$[C] = \sum_e \int_{\Omega^e} \{N_T\}^{n+1} (\rho c_p)^{n+1} \mathbf{v}^{n+1} [B_T]^{n+1} d\Omega \quad (13d)$$

$$[K] = \sum_e \int_{\Omega^e} ([B_T]^T)^{n+1} k^{n+1} [B_T]^{n+1} d\Omega \quad (13e)$$

$$[M_\lambda] = \sum_e \int_{\Omega^e} \frac{1}{k^{n+1}} ([N_\lambda]^T)^{n+1} [N_\lambda]^{n+1} d\Omega \quad (13f)$$

$$[Q] = \sum_e \int_{\Omega^e} ([N_\lambda]^T)^{n+1} [B]^{n+1} d\Omega \quad (13g)$$

$$[L] = \sum_e \int_{\Gamma^e} \{N_T\}^{n+1} [N_\lambda]^{n+1} \mathbf{n}_{\Gamma} d\Gamma \quad (13h)$$

$$\{\mathbf{f}_\lambda\} = \sum_e \int_{\Gamma^e} ([N_\lambda]^T)^{n+1} \cdot \mathbf{n}_{\Gamma} T_m d\Gamma \quad (13i)$$

where $B_{ij}^{n+1} = \frac{\partial N_j^{n+1}}{\partial x_i}$ is the gradient matrix and the $\{\}$ brackets are used to indicate a vector transpose. In elements which are not cut by the interface, the boundary condition is removed and the system reduces to:

$$\begin{bmatrix} \frac{1}{\Delta t} [M] + [C] + [K] & 0 \\ [Q] & [M_\lambda] \end{bmatrix} \begin{Bmatrix} \{T\}^{n+1} \\ \{\lambda\}^{n+1} \end{Bmatrix} = \begin{bmatrix} \frac{1}{\Delta t} [M]^* & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \{T\}^n \\ \{\lambda\}^n \end{Bmatrix} \quad (14)$$

3.2. Stokes Problem

The weak form of the Stokes problem (4) is given as follows

$$\int_{\Omega_l} \delta \mathbf{v} \cdot \rho_l \frac{\partial \mathbf{v}}{\partial t} d\Omega + \int_{\Omega_l} 2\mu D(\delta \mathbf{v}) : D(\mathbf{v}) d\Omega - \int_{\Omega_l} (\nabla \cdot \delta \mathbf{v}) p d\Omega = 0$$

$$\int_{\Omega_l} \delta p \nabla \cdot \mathbf{v} d\Omega = 0 \quad (15a)$$

where $\delta \mathbf{v}$ and δp are the test functions for the velocity and pressure, respectively. The Neumann boundary condition has been omitted for the sake of clarity. Using a backward Euler scheme for the time derivative of \mathbf{v} in (15) gives the system of equations Fries and Zilian (2009):

$$\int_{\Omega_l} \delta \mathbf{v}^{n+1} \cdot \left(\frac{(\rho_l \mathbf{v})^{n+1} - (\rho_l \mathbf{v})^n}{\Delta t} \right) d\Omega + \int_{\Omega_l} 2\mu D(\delta \mathbf{v}^{n+1}) : D(\mathbf{v}^{n+1}) d\Omega - \int_{\Omega_l} \nabla \cdot \delta \mathbf{v}^{n+1} p^{n+1} d\Omega = 0 \quad (16a)$$

$$\int_{\Omega_l} \delta p^{n+1} \nabla \cdot \mathbf{v}^{n+1} d\Omega = 0 \quad (16b)$$

Substituting the approximation for the velocity and pressure fields into (16) leads to the system of equations:

$$\begin{bmatrix} [K] & -[D] \\ [D]^T & 0 \end{bmatrix} \begin{Bmatrix} \{\mathbf{v}\}^{n+1} \\ \{p\}^{n+1} \end{Bmatrix} = \begin{bmatrix} [M]^* & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \{\mathbf{v}\}^n \\ \{p\}^n \end{Bmatrix} \quad (17a)$$

$$[K] = [M] + \begin{bmatrix} [A_{11}] & [A_{12}] \\ [A_{12}]^T & [A_{22}] \end{bmatrix} \quad (17b)$$

$$[M] = \frac{1}{\Delta t} \sum_e \int_{\Omega^e} ([N_v]^T)^{n+1} \rho_l [N_v]^{n+1} d\Omega \quad (17c)$$

$$[M]^* = \frac{1}{\Delta t} \sum_e \int_{\Omega^e} ([N_v]^T)^{n+1} \rho_l [N_v]^n d\Omega \quad (17d)$$

$$[A_{11}] = \sum_e \int_{\Omega^e} 2\mu \left(\{B_x\}^{n+1} \langle B_x \rangle^{n+1} + \frac{1}{2} \{B_y\}^{n+1} \langle B_y \rangle^{n+1} \right) d\Omega \quad (17e)$$

$$[A_{22}] = \sum_e \int_{\Omega^e} 2\mu \left(\frac{1}{2} \{B_x\}^{n+1} \langle B_x \rangle^{n+1} + \{B_y\}^{n+1} \langle B_y \rangle^{n+1} \right) d\Omega \quad (17f)$$

$$[A_{12}] = \sum_e \int_{\Omega^e} 2\mu \left(\frac{1}{2} \{B_y\}^{n+1} \langle B_x \rangle^{n+1} \right) d\Omega \quad (17g)$$

$$[D] = \sum_e \int_{\Omega^e} \langle \langle B_x \rangle^{n+1} \langle B_y \rangle^{n+1} \rangle^T \langle N_p \rangle^{n+1} d\Omega \quad (17h)$$

$$\langle B_{x_i} \rangle^{n+1} = \frac{\partial \langle N_1^v \psi_1^v, \dots, N_{n_I}^v \psi_{n_I}^v \rangle^{n+1}}{\partial x_i} \quad (17i)$$

The mass flux interface boundary condition is imposed using the penalty method Chessa *et al.* (2002); Bernauer and Herzog (2011). This technique multiplies the residual form of equation (4c) by a very large penalization parameter β and introduces it in the finite element formulation of the momentum equation. This method is simple to implement and has proven to be robust for a variety of problems. The formulation for elements intersected by the interface becomes:

$$\begin{bmatrix} [K'] & -[D] \\ [D]^T & 0 \end{bmatrix} \begin{Bmatrix} \{\mathbf{v}\}^{n+1} \\ \{p\}^{n+1} \end{Bmatrix} = \begin{bmatrix} [M]^* & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \{\mathbf{v}\}^n \\ \{p\}^n \end{Bmatrix} + \begin{Bmatrix} \{\mathbf{f}_p^{n+1}\} \\ 0 \end{Bmatrix} \quad (18a)$$

$$[K'] = [K] + [P] \quad (18b)$$

$$[P] = \sum_e \int_{\Gamma^e} ([N_v]^T)^{n+1} \beta [N_v]^{n+1} d\Gamma \quad (18c)$$

$$\{\mathbf{f}_p^{n+1}\} = \sum_e \int_{\Gamma^e} ([N_v]^T)^{n+1} \beta \left(\frac{\rho_l - \rho_s}{\rho_l} v_{\Gamma} \mathbf{n}_{\Gamma} \right) d\Gamma \quad (18d)$$

To solve the system of equations (17)-(18) the interpolation functions for the velocity and pressure fields must satisfy the LBB condition Babuska (1969); Brezzi (1974). In this work, a pair of stable Q₂-Q₁ quadrilateral elements was used for the velocity and pressure fields, respectively.

The validation problems presented in the results section are compared with the solution obtained through the commercial finite element code Comsol where the phase change problem was solved using a moving mesh algorithm (ALE) to capture the interface movement.

The interpolation scheme (9) is known to cause problems when the physical domain (liquid phase) covers a very small area of the node's support Lang *et al.* (2014). The small contribution of the concerned degree of freedom causes a significant increase in the condition number of the global system Lang *et al.* (2014), leading to divergent solutions. An efficient solution was developed in Lang *et al.* (2014). When a degree of freedom's contribution to the system is too small, it is removed from the system. The criteria for removing a degree of freedom is Lang *et al.* (2014)

$$\left(\frac{\max_{e \in E_i} \int_{\Omega^e} N_i(\mathbf{x}) d\Omega}{\int_{\Omega^e} N_i(\mathbf{x}) d\Omega} \right)^{-\frac{1}{2}} > T_{tol} \quad (19)$$

where E_i is the set of elements connected to node i , Ω_l^e the liquid domain area in the element, Ω^e the element area, $N_i(\mathbf{x})$ the interpolation function and T_{tol} a user defined tolerance value. The greater the value for T_{tol} , the smaller the contribution of the degree of freedom can be before it is removed.

The stopping criteria (19) is used on a stabilized Q₁-Q₁ in Lang *et al.* (2014), meaning that the velocity and pressure interpolation functions are identical, linear and positive-semidefinite. The quadratic interpolation used for velocity in this work however, is not positive-semidefinite. This means that certain interface positions would lead to near zero integrals in (19) even when the liquid area is large, because the negative-valued areas of the interpolation would cancel out the positive-valued areas. To maintain the original objective of evaluating the relative contribution of the degrees of freedom to the complete element, a modified criteria was used, given by equation

$$\left(\frac{\max_{e \in E_i} \int_{\Omega^e} |N_i(\mathbf{x})| d\Omega}{\int_{\Omega^e} |N_i(\mathbf{x})| d\Omega} \right)^{-\frac{1}{2}} > T_{tol} \quad (20)$$

where the absolute value of the interpolation function is used.

In Lang *et al.* (2014) a preconditioner is applied to the global system before solving, allowing the use of a higher value of T_{tol} while maintaining an optimal condition number and accurate solution. Considering the relatively heuristic modifications made to the removal of degrees of freedom caused by the use of a Q₂-Q₁ formulation and to simplify the implementation of our model, the preconditioner was not applied in this work.

The systems of equations (13) and (17) are coupled through the convection and mass flux boundary terms, respectively. To obtain a converged solution for both systems, a fixed point iteration scheme is used.

3.3. Level Set Formulation

Once an initial value $\phi(\mathbf{x}, t_0)$ is defined, the interface movement is governed by its transport equation

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = \frac{\partial \phi}{\partial t} + F \|\nabla \phi\| = 0 \quad (21a)$$

$$F = \frac{1}{\|\nabla \phi\|} \nabla \phi \cdot \mathbf{v} \quad (21b)$$

where \mathbf{v} is the convection velocity and F is the interface speed in the normal direction. The calculation of F is explained below.

Equation (21) is solved explicitly (forward Euler scheme) with the finite element method using a linear interpolation. The weak formulation and time discretization of (21) is given as follows:

$$\int_{\Omega} \delta \phi \frac{\phi^{n+1} - \phi^n}{\Delta t} d\Omega + \int_{\Omega} \delta \phi F^n \|\nabla \phi^n\| d\Omega = 0 \quad (22)$$

In most applications, the normal component F is only known on Γ . In order to solve (22) on Ω , a valid value for F must first be constructed on the entire domain using the following problem Chessa *et al.* (2002):

$$\text{sign}(\phi) \nabla F \cdot \nabla \phi = 0 \quad \mathbf{x} \in \Omega \quad (23a)$$

$$F(\mathbf{x}, t) = \frac{\nabla \phi}{\|\nabla \phi\|} \cdot \mathbf{v}_{\Gamma} \quad \mathbf{x} \in \Gamma \quad (23b)$$

This approach guarantees that the ϕ field velocity is everywhere normal to the interface and is coherent with the interface's physically determined velocity. For more details concerning the construction of F see Osher and Fedkiw (2003); Chessa *et al.* (2002). In this paper, the interface velocity is based on the jump in heat flux across the phase change boundary and is described in the following section.

Equation (22) is first order hyperbolic and must be stabilized to minimize the presence of oscillations in the solution Chessa *et al.* (2002); Bernauer and Herzog (2011). The GLS method is used here Hughes *et al.* (1989). The level set method offers several advantages. It is easily extensible to three dimensions and stores the interface location as a scalar variable. Furthermore, the level set field can be defined in a small region surrounding the interface and the level set formulation solved locally, reducing the impact on the total simulation computation time. It is also robust enough to handle interface merging and breaking naturally Osher and Fedkiw (2001).

The main disadvantage of the the level set method is its tendency to deviate from a signed distance function over time Osher and Fedkiw (2001). This error accumulates with additional time steps and degrades the quality of the solution, particularly the level set gradient near the interface. This distortion can be a source of error in the numerical solution of the level set formulation and the physical problem on which it is based. Therefore, it is necessary to reinitialize $\phi(\mathbf{x}, t)$ regularly to maintain an acceptable solution ($\|\nabla \phi\| \approx 1$). Another limitation to the algorithm presented here is the use of an explicit time scheme for the level set formulation, which limits the size of the time step. The explicit time step is required in order to determine the nodes to enrich. In other words, the interface position must be determined before systems (13) and (17) are solved.

3.4. Interface velocity calculation

The proper evaluation of the interface velocity is crucial in obtaining a precise and robust model Martin *et al.* (2016). For this particular problem, the interface velocity is determined by the jump in heat flux at the interface as described in equation (2). The use of a Lagrange multiplier to impose the melting temperature allows the evaluation of the jump in heat flux directly from the Lagrange field λ , given by

$$v_{\Gamma} = \frac{[\lambda] \cdot \mathbf{n}_s}{\rho_s L} = \frac{(\lambda_l - \lambda_s) \cdot \mathbf{n}_s}{\rho_s L} \quad (24)$$

where λ_s and λ_l are the heat flux at the interface approaching from the solid and liquid phases, respectively.

The final algorithm can be described as follows. Assuming that a given time t^n , solution $(T^n, \mathbf{v}^n, p^n, \phi^n)$ are known, the strategy to solve for $(T^{n+1}, \mathbf{v}^{n+1}, p^{n+1}, \phi^{n+1})$ consists in the following steps:

1. Compute the interface velocity $\mathbf{v}_{\Gamma}^{n+1}$ using (24)
2. Construct F on the level set domain by solving (23)
3. Solve for ϕ^{n+1} using (22)

4. Solve the coupled Stefan-Stokes problem:
 - 4.1. Solve for T_{i+1}^{n+1} using (13) and \mathbf{v}_i^{n+1}
 - 4.2. Solve for \mathbf{v}_{i+1}^{n+1} and p_{i+1}^{n+1} using (17) and T_{i+1}^{n+1}
5. Evaluate (13) and (17). If both residuals are below the tolerance criteria, go to step 6. If not, $i = i + 1$ and go to step 4
6. Set $t^{n+1} = t^n$ and go to step 1.

3.5. Numerical Integration

The introduction of discontinuous functions inside elements greatly reduces the precision of standard Gaussian quadrature and may lead to rank deficient matrices Chessa *et al.* (2002). An accurate but geometrically complex solution is to subdivide elements involving discontinuities into continuous subelements Moes *et al.* (1999); Chessa *et al.* (2002); Gerstenberger and Wall (2010). Each element is subdivided into a number of subelements (lines, triangles or tetrahedrons), as shown in figure 1, to properly fit the contour of the interface (point, line or surface) and element boundaries. The integral over the entire element I_e is then the sum of the integration of each subelement I_s using standard Hammer quadrature. It is important to note that subelements carry no degrees of freedom or interpolation functions. They are only required as a geometrical tool to construct the element integrals.

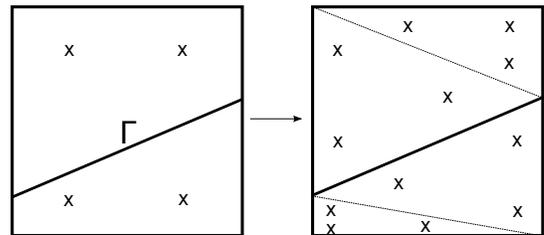


Fig. 1 Geometry subdivision for cut element integration

In transient problems the location of the quadrature points must change as the interface moves in time, requiring that every cut element be subdivided at each time step. However, the subdivision is applied only to a small number of elements, reducing the overall increase in computational effort required.

In transient problems, the interpolation functions at time steps n and $n + 1$ are based on different positions of the interface and are discontinuous at different places in the element. The integration scheme for the mass matrix (equations (13c) and (17d)) must take both intersections into account when generating the integration subelements to obtain optimal convergence Fries and Zilian (2009). This can be difficult and can significantly increases the number of subelements required to fit the geometry. However, previous authors have successfully used integration schemes considering the current interface position only Chessa *et al.* (2002); Chessa and Belytschko (2003) and this strategy is used in this work. As suggested in Fries and Zilian (2009), the test functions are evaluated using the current time step's level set values.

4. RESULTS

The Lagrange multiplier formulation used in this work to solve the Stefan problem (1) has been previously validated. For details on the specific simulations used and its performance compared to a finite difference approximation of equation (2), the interested reader is referred to Martin *et al.* (2016).

To validate the coupled model three benchmark problems were simulated. The first and second are based on the one- and two-dimensional analytical phase change problems Merle and Dolbow (2002); Rathjen and Jiji (1971). The third benchmark problem involves more realistic boundary conditions, using the material properties of cryolite, known for its important change in density ($\approx 25\%$).

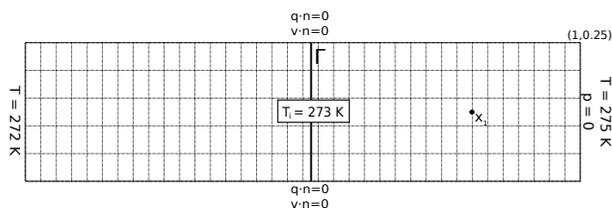


Fig. 2 1D problem definition

Properties	Solid	Liquid	Interface
ρ [kg/m ³]	1.0	0.75	-
c_p [J/kg K]	50	50	-
k [W/m K]	0.1	0.1	-
$\rho_s L$ [J/m ³]	-	-	2.5
T_m [K]	-	-	273.0
μ [kg/s·m]	-	10	-

Table 1 Material properties for 1D and 2D problems

In all cases, the simulations were also run in Comsol, using a moving mesh algorithm (ALE) to account for the displacement of the interface. The Stokes formulation uses a $P_2 - P_1$ formulation, the temperature field is linear and the Lagrange multiplier is constant per element [Martin et al. \(2016\)](#). In Comsol, the mesh geometry is quadratic. The results were then compared to the solution obtained using the purely XFEM approach. The Comsol simulations did not include a remeshing step during the simulation. An appropriate element size was used to maintain a low enough Peclet number to avoid oscillations in the Stefan problem.

These problems were selected for their relatively simple interface geometry and no reinitialization procedure was applied to the level set field during the simulation. For smooth interface shapes and relatively uniform displacements, the absence of a reinitialization step had little impact on the model's accuracy [Martin \(2016\)](#). More complex shapes and interface movements would require a reinitialization step as well as a remeshing step in the Comsol algorithm.

4.1. One Dimensional Phase Change Problem

The first benchmark problem is inspired by the one dimensional two phase analytical solution of the Stefan problem in a semi-infinite domain ($x > 0$), taken from [Merle and Dolbow \(2002\)](#). The thermal properties are constant except for the density and are given in table 1. The initial interface is at $x = 0.515$ m with the liquid phase on the right and solid phase on the left, as shown in figure 2. The initial temperature is T_m (see table 1). The top and bottom edges are insulated. At $t = 0$, the temperature on the left edge is lowered to 272 K and the right edge is increased to 275 K. For the Navier-Stokes equations, the right boundary is open (no stress) while for the top and bottom edges the following boundary condition is applied: $\mathbf{v} \cdot \mathbf{n} = 0$. The time step is 0.05 sec, $\beta = 1 \times 10^8$ and $T_{tol} = 1 \times 10^8$. The mesh contains 180 quadrilateral elements in XFEM and 196 in Comsol.

The interface position as a function of time for both Comsol and XFEM algorithms is shown in figure 3. The temperature at point x_1 over time is given in figure 4 for Comsol and XFEM algorithms. The convection velocity (constant in the liquid domain) is shown in figure 5 for both algorithms. These results show that the XFEM method reproduces the solution obtained through the standard finite element method (Comsol) using the moving mesh algorithm.

4.2. Two Dimensional Phase Change Problem

The second benchmark problem is two dimensional and based on the analytical solution of melting (or freezing) in a corner first solved in [Rathjen](#)

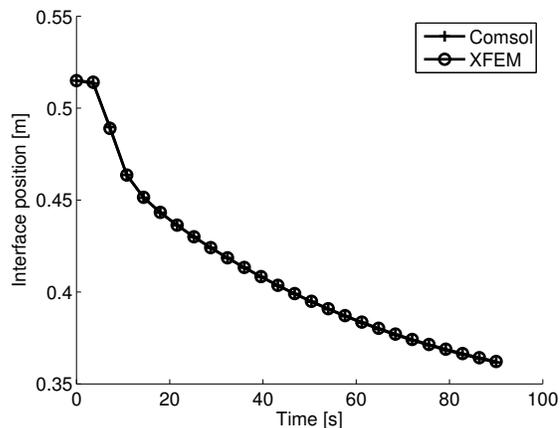


Fig. 3 Interface position vs time, 1D problem

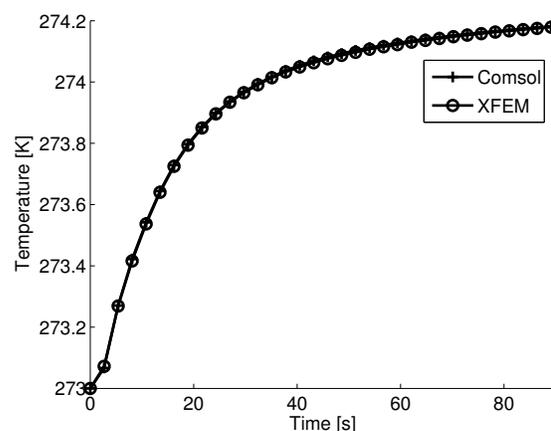


Fig. 4 Temperature at point x_1 , 1D problem (see figure 2)

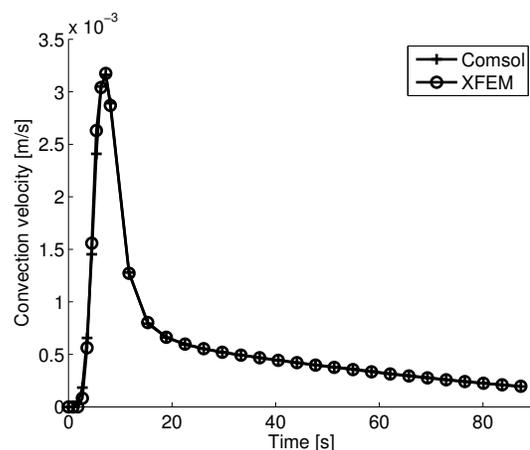


Fig. 5 Convection velocity in liquid phase, 1D problem

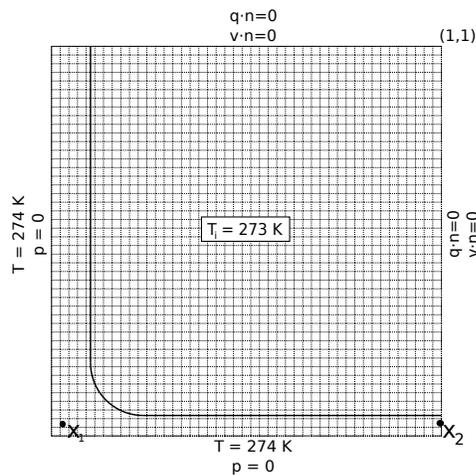


Fig. 6 2D problem definition.

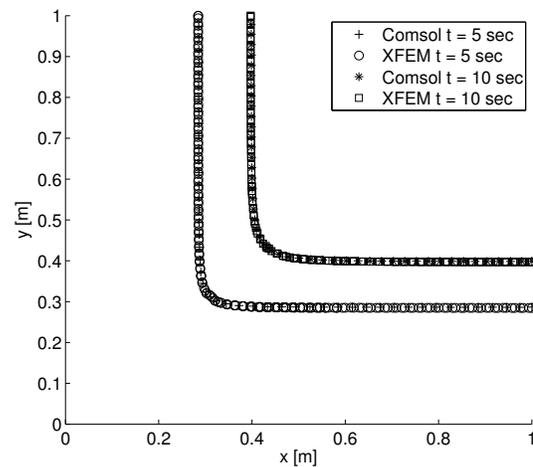


Fig. 7 Interface Cmsol and XFEM, 2D problem

and Jiji (1971). The thermal properties are constant except for the density and are identical to the first example, given in table 1. The initial interface is at $x = 0.1$ m from the left and bottom boundaries, with the liquid phase on the lower left and solid phase on the top right, as shown in figure 6. The initial temperature is T_m (table 1). The top and right edges are thermally insulated. At $t = 0$, the temperature on the left and bottom boundaries is increased to 274 K. For the Navier-Stokes equations, the left and bottom boundaries are open (no stress). For the top and right boundaries the boundary condition $\mathbf{v} \cdot \mathbf{n} = 0$ is applied. The time step is 0.05 sec, $\beta = 1 \times 10^4$ and $T_{tol} = 1 \times 10^2$. The mesh contains 3025 quadrilateral elements in XFEM and 6590 triangle elements in Cmsol.

The interface position for two different time steps for both Cmsol and XFEM algorithms is shown in figure 7. The figure shows that the Cmsol and XFEM algorithms give identical interface positions.

The temperature profile at the end of the simulation is shown in figure 8. Figure 9 shows the temperature at two points \mathbf{x}_1 and \mathbf{x}_2 over time (see figure 6). In both cases, the Cmsol and XFEM algorithms are in excellent agreement.

The convection velocity at the final time step for both algorithms is shown in figure 10. The velocity is in good agreement in both cases. Figure 11 shows the fluid velocity at points \mathbf{x}_1 and \mathbf{x}_2 over time, showing good agreement between the two algorithms, although fluctuations are present in the XFEM solution.

Two distinct causes contribute to these fluctuations. First, the interface geometry in XFEM (the level set field) is stored using a linear interpolation. Consequently, the curved interface is approximated by line segments which reduces the accuracy of the solution. Cmsol uses a quadratic interpolation for its moving mesh solution, allowing it to reproduce the interface curvature precisely. To validate this hypothesis, a solution was obtained using Cmsol and a linear geometry. Using a mesh size similar to figure 6, Cmsol is unable to produce a converged solution. To obtain a converged solution, over 24 000 triangle elements had to be used and the velocity solution showed small errors, similar to the XFEM solution in figure 11. Furthermore, refining the XFEM mesh, thus reducing the error caused by the linear geometry, reduces the error in the solution as can be seen in figure 12, where the error norm is defined as $\|v_C - v_X\|_2$, v_C is the Cmsol velocity and v_X the XFEM velocity over time at point \mathbf{x}_2 .

To eliminate this error, a quadratic interface geometry (level set solution) can be used for significantly curved interfaces. Note that this does not require the use of a quadratic interpolation of the mesh, (as in Cmsol) but the level set field only, limiting the number of additional degrees of freedom required. However, the geometric calculations done using the level set solution (element intersections, normals) becomes more com-

plex to implement and the algorithms supposing a linear interpolation must be rewritten. Note that the use of a linear interface may reduce precision but still leads to converged solutions, whereas Cmsol struggles to produce converged solutions using a linear mesh interpolation. These findings suggest that the use of the extended finite element method combined with the level set method allows for a significant reduction in the number of degrees of freedom required to reach a converged solution, as opposed to Cmsol's moving mesh algorithm.

The second factor is the absence of a preconditioning matrix Lang et al. (2014) for the Stokes problem, which meant that lower T_{tol} values had to be used to obtain a converged solution. This low T_{tol} value causes important local errors in the velocity near the interface for problematic time steps; the more significant jumps in 11 are caused by this. To evaluate the sensitivity of the solution with respect to T_{tol} , the problem was solved using three different values for T_{tol} : 1×10^0 , 1×10^2 and 1×10^4 . The velocity norm at point \mathbf{x}_1 for all three T_{tol} values compared to the Cmsol solution is given in figure 13. As can be seen in the figure, at $T_{tol} = 1 \times 10^4$, certain critical degrees of freedom are not removed and the error becomes quite significant at 3.5 seconds. This error distorts the level set solution and leads to a divergent solution a few time steps later. At $T_{tol} = 1 \times 10^0$ too many degrees of freedom are removed and the system is unable to reproduce the solution at any time step. Between these two values, at $T_{tol} = 1 \times 10^2$, the solution shows much smaller errors at critical time steps and reproduces the correct solution. These results indicate that without a preconditioner, the solution can be quite sensitive to a change in T_{tol} .

This source of error can be resolved by adding the preconditioning matrix defined in Lang et al. (2014) to the Stokes problem. Although this error impacted only certain time steps, even at the quite low T_{tol} , more general applications involving other sources of fluid flow may not be so stable Martin (2016).

4.3. Melting of Cryolite Problem

The last benchmark problem is the melting of cryolite inside a rectangular cavity. The material properties are taken from the FactSage software Bale et al. (2009) and assumed constant except for the density (see table 2). The initial interface is at $x = 0.05$ m with the liquid phase on the left and solid phase on the right, as shown in figure 14. The initial temperature is T_m (table 2). The top and bottom edges are thermally insulated. The temperature on the left and right boundaries are functions of y and given in figure 14. For the Navier-Stokes equations, all boundaries are no-slip walls ($\mathbf{v} = 0$) except for an open boundary ($p = 0$) on the top left side of width 0.01 m (see figure 14). The time step used is 2×10^3 sec,

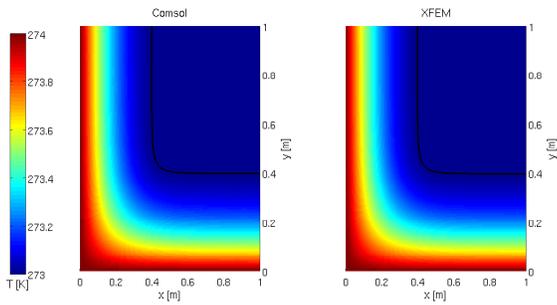


Fig. 8 Temperature at final time step, 2D problem

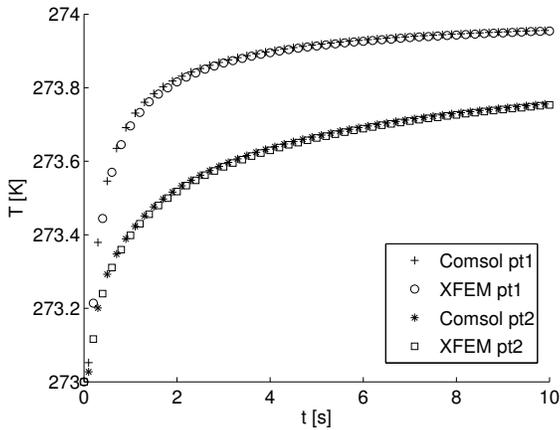


Fig. 9 Temperature as function of time at x_1 and x_2 , 2D problem (see figure 6)

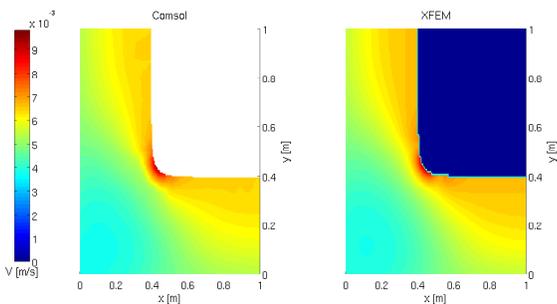


Fig. 10 Velocity at final time step, 2D problem

$\beta = 1 \times 10^4$ and $T_{tol} = 1 \times 10^1$. The mesh contains 1350 quadrilateral elements in XFEM and 2799 triangle elements in Comsol.

The varying temperature profile along the left and right boundaries will create a variation in heat flux along the interface, causing it to curve. As the solid phase melts, excess mass is released in the liquid phase and leaves the domain through the open boundary in order to fulfill the mass conservation principal.

The interface position for three different time steps for both Comsol and XFEM algorithms are shown in figure 15 and are in excellent agreement. The temperature profile at the end of the simulation is shown in figure 16 whereas figure 17 shows the temperature over time at two points x_1 and x_2 (see figure 14). In both cases, the Comsol and XFEM algorithms are in excellent agreement.

The velocity profile at the end of the simulation is shown in figure 18. Figure 19 shows the fluid velocity at point x_1 over time. Finally, figure 20 shows the mass flux across the open boundary over time. The graphs clearly indicate that the XFEM algorithm correctly solves the

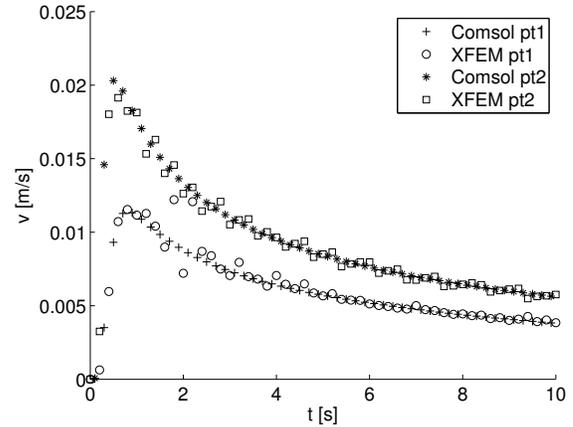


Fig. 11 Velocity as function of time at x_1 and x_2 , 2D problem (see figure 6)

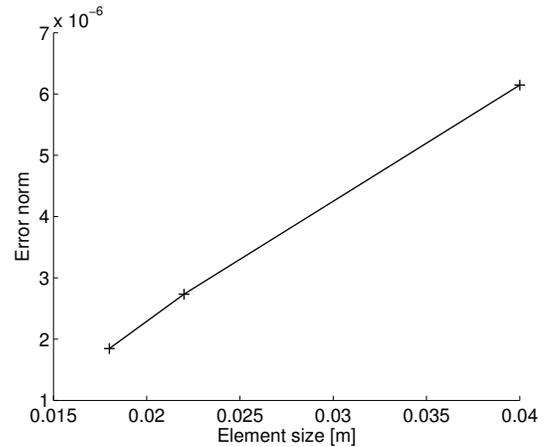


Fig. 12 Error norm of velocity over time of XFEM compared to Comsol for different mesh sizes

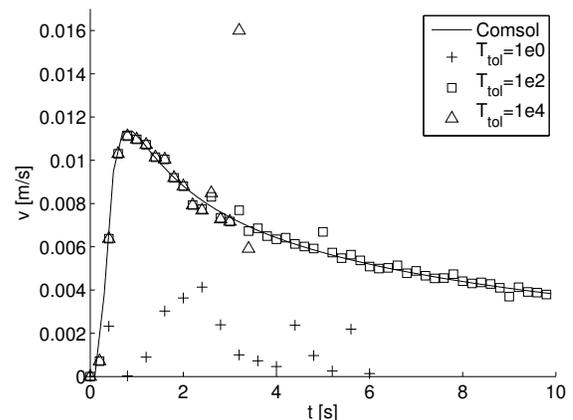


Fig. 13 Velocity at x_1 for different T_{tol} values

Properties	Solid	Liquid	Interface
ρ [kg/m ³]	2900	2050	-
c_p [J/kg K]	1650	1650	-
k [W/m K]	0.4	0.4	-
$\rho_s L$ [J/m ³]	-	-	2.81×10^9
T_m [K]	-	-	1000
μ [kg/s·m]	-	2.4×10^{-3}	-

Table 2 Material properties of cryolite, taken from FactSage [Bale et al. \(2009\)](#)

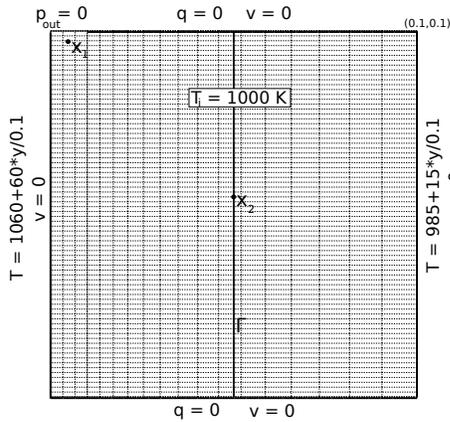


Fig. 14 Cryolite problem definition

interface, temperature and velocity variables. A mismatch between the Comsol and XFEM algorithms can be seen at earlier time steps, when the interface velocity varies rapidly. This is caused by the use of an explicit time stepping scheme for the level set field, requiring the use of the previous time step's temperature values to calculate the interface velocity (equation 2).

5. CONCLUSION

Coupled Stefan and Stokes formulations using the extended finite element method were developed for the resolution of phase change problems involving variable densities. The density jump at the interface was used to apply a velocity boundary condition and conserve the global mass of the system, using the penalty method. The temperature and velocity fields obtained using XFEM were compared to the moving mesh algorithm in

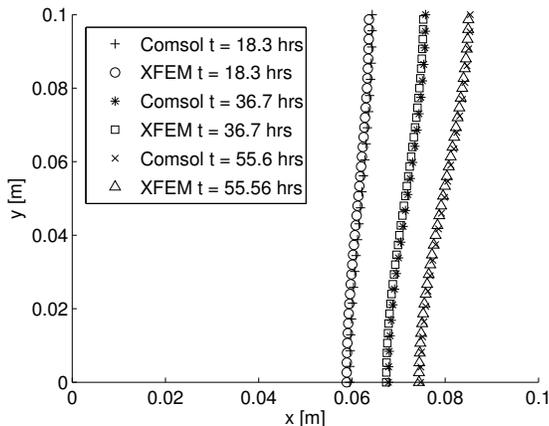


Fig. 15 Interface position for cryolite problem

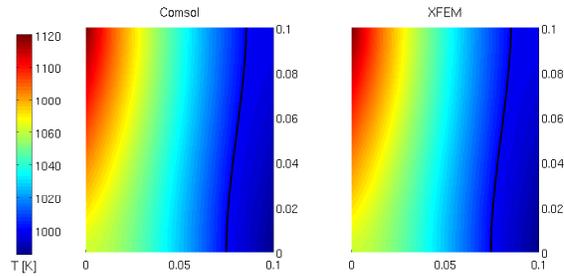


Fig. 16 Temperature profile at the final time step for cryolite problem

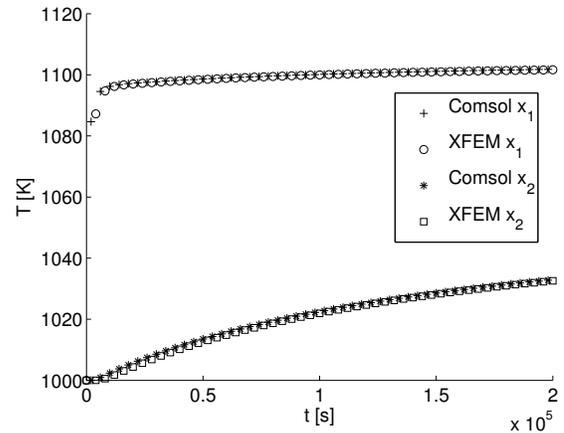


Fig. 17 Temperature as function of time at x_1 and x_2 for cryolite problem (see figure 14)

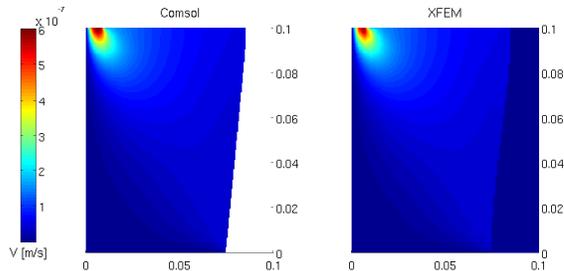


Fig. 18 Velocity profile at the final time step for cryolite problem

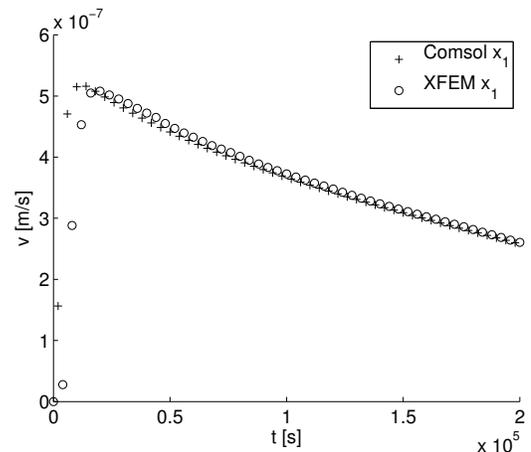


Fig. 19 Velocity as function of time at x_1 and x_2 for cryolite problem (see figure 14)

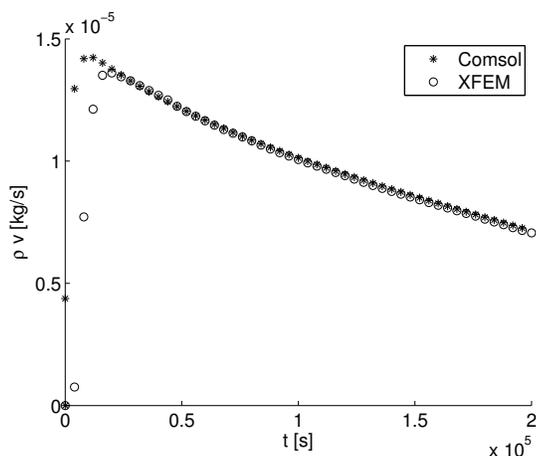


Fig. 20 Mass flux at outflow boundary, cryolite problem

Cmsol and are in good agreement. The use of a linear interpolation for the level set solution lead to errors in the mass flux velocity at the interface compared to the quadratic interpolation used by Cmsol but required fewer degrees of freedom. When a linear interpolation was used in Cmsol, the number of degrees of freedom required to obtain a converged solution was much greater than in XFEM. The simple removal of degrees of freedom with a small contribution to the system for the Q_2 - Q_1 Stokes formulation was shown to produce errors in the velocity field for problematic interface configurations. The same observation for a Q_1 - Q_1 formulation was made in Lang et al. (2014). The resolution of a more physically realistic benchmark problem using cryolite showed the XFEM algorithm to be quite effective at evaluating the mass flux caused by the density change. Future work will be done to include the complete Navier-Stokes equations and a stabilized Q_1 - Q_1 formulation to implement the preconditionner scheme.

Acknowledgement

The authors are grateful for the research support of the Natural Sciences and Engineering Research Council of Canada (File IRSCSA 394855 - 07) and Alcoa. A part of the research presented in this paper was financed by the Fonds de recherche du Québec - Nature et Technologies by the intermediary of the Aluminium Research Centre - REGAL. Special thanks to Patrice Goulet for his invaluable assistance with the development of the level set algorithm, and advice on the art of code writing.

REFERENCES

Babuska, I., and Melenk, J., 1997, "The Partition of Unity Method," *International Journal for Numerical Methods in Engineering*, **40**(4), 727–758.
[http://dx.doi.org/10.1002/\(SICI\)1097-0207\(19970228\)40:4<727::AID-NME86>3.3.CO;2-E](http://dx.doi.org/10.1002/(SICI)1097-0207(19970228)40:4<727::AID-NME86>3.3.CO;2-E).

Babuska, I., 1969, "Error-Bounds for Finite Element Method," *Numerische Mathematik*, **16**, 322–333.

Baiges, J., Codina, R., Henke, F., Shahmiri, S., and Wall, W.A., 2012, "A Symmetric Method for Weakly Imposing Dirichlet Boundary Conditions in Embedded Finite Element Meshes," *International Journal for Numerical Methods in Engineering*, **90**, 636–658.

Bale, C.W., Bélisle, E., Chartrand, P., Deckerov, S.A., G. Eriksson, K.H., Jung, I.H., Kang, Y.B., J. Melançon, A.D.P., Robelin, C., and Petersen, S., 2009, "FactSage Thermochemical Software and Databases - Recent Developments," *Calphad*, **33**, 295–331.

Beckermann, C., Diepers, H., Steinbach, I., Karma, A., and Tong, X., 1999, "Modeling Melt Convection in Phase-field Simulations of Solidification," *Journal of Computational Physics*, **154**(2), 468–496.
<http://dx.doi.org/10.1006/jcph.1999.6323>.

Belytschko, T., Moes, N., Usui, S., and Parimi, C., 2001, "Arbitrary Discontinuities in Finite Elements," *International Journal for Numerical Methods in Engineering*, **50**, 993–1013.
[http://dx.doi.org/10.1002/1097-0207\(20010210\)50:4<993::AID-NME164>3.0.CO;2-M](http://dx.doi.org/10.1002/1097-0207(20010210)50:4<993::AID-NME164>3.0.CO;2-M).

Belytschko, T., Gracie, R., and Ventura, G., 2009, "A Review of Extended/generalized Finite Element Methods for Material Modeling," *Modelling and Simulation in Materials Science and Engineering*, **17**.

Bernauer, M., and Herzog, R., 2011, "Implementation of an XFEM Solver for the Classical Two-Phase Stefan Problem," *Journal of Scientific Computing*, **52**, 271–293.

Brent, A., Voller, V., and Reid, K., 1988, "Enthalpy-Porosity Technique for Modeling Convection-Diffusion Phase Change: Application to the Melting of a Pure Metal," *Numerical Heat Transfer: An International Journal of Computation and Methodology*, **13**:3, 297–318.

Brezzi, F., 1974, "On the Existence, Uniqueness and Approximation of Saddle-Point Problems arising from Lagrangian Multipliers," *Revue française d'automatique, informatique, recherche opérationnelle Analyse numérique*, **8**, 129–151.

Chessa, J., and Belytschko, T., 2003, "An Extended Finite Element Method for Two-Phase Fluids," *Journal of Applied Mechanics*, **70**, 10.
<http://dx.doi.org/10.1115/1.1526599>.

Chessa, J., Smolinski, P., and Belytschko, T., 2002, "The Extended Finite Element Method (XFEM) for Solidification Problems," *International Journal for Numerical Methods in Engineering*, **53**(8), 1959–1977.
<http://dx.doi.org/10.1002/nme.386>.

Dolbow, J., Moës, N., and Belytschko, T., 2000, "Discontinuous Enrichment in Finite Elements with a Partition of Unity Method," *Finite Elements in Analysis and Design*, **36**, 235–260.
[http://dx.doi.org/10.1016/S0168-874X\(00\)00035-4](http://dx.doi.org/10.1016/S0168-874X(00)00035-4).

Fries, T.P., 2008, "A Corrected XFEM Approximation Without Problems in Blending Elements," *International Journal for Numerical Methods in Engineering*, **75**(5), 503–532.
<http://dx.doi.org/10.1002/nme.2259>.

Fries, T.P., and Zilian, A., 2009, "On Time Integration in the XFEM," *International Journal for Numerical Methods in Engineering*, **79**(1), 69–93.
<http://dx.doi.org/10.1002/nme.2558>.

Gerstenberger, A., and Wall, W.A., 2010, "An Embedded Dirichlet Formulation for 3D Continua," *International Journal for Numerical Methods in Engineering*, **82**(5), 537–563.
<http://dx.doi.org/10.1002/nme.2755>.

Gerstenberger, A., 2010, *An XFEM Based Fixed-grid Approach to Fluid-Structure Interaction*, Ph.D. thesis, Technical University of Munich.

Gerstenberger, A., and Wall, W.A., 2008, "An eXtended Finite Element Method/Lagrange Multiplier Based Approach for Fluid-Structure Interaction," *Computer Methods in Applied Mechanics and Engineering*, **197**(19-20), 1699–1714.
<http://dx.doi.org/10.1016/j.cma.2007.07.002>.

Helenbrook, B.T., 2013, "High-Order Adaptive Arbitrary-Lagrangian-Eulerian (ALE) Calculations of Solidification," *Proc. of the ASME FEDSM*, vol. 1C.

- Hughes, T.J., Franca, L.P., and Hulbert, G.M., 1989, "A New Finite Element Formulation for Computational Fluid Dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations," *Computer Methods in Applied Mechanics and Engineering*, **73**(2), 173 – 189. [http://dx.doi.org/10.1016/0045-7825\(89\)90111-4](http://dx.doi.org/10.1016/0045-7825(89)90111-4).
- Ji, H., Chopp, D., and Dolbow, J., 2002, "A Hybrid Extended Finite Element/Level Set Method for Modeling Phase Transformations," *International Journal for Numerical Methods in Engineering*, **54**(8), 1209–1233. <http://dx.doi.org/10.1002/nme.468>.
- Lang, C., Makhija, D., Doostan, A., and Maute, K., 2014, "A Simple and Efficient Preconditioning Scheme for Heaviside Enriched XFEM," *Computational Mechanics*, **54**, 1357–1374. <http://dx.doi.org/10.1007/s00466-014-1063-8>, 1312 . 6092.
- Martin, D., 2016, *Multiphase Modelling of Melting/Solidification with High Density Variations using XFEM*, Ph.D. thesis, Laval University.
- Martin, D., Chaouki, H., Robert, J.L., Ziegler, D., and Fafard, M., 2016, "A XEFM Lagrange Multiplier Technique for Stefan Problems," *Frontiers in Heat and Mass Transfer*, **7**. <http://dx.doi.org/10.5098/hmt.7.31>.
- Melenk, J., and Babuska, I., 1996, "The Partition of Unity Finite Element Method: Basic Theory and Applications," *Computer Methods in Applied Mechanics and Engineering*, **139**, 289–314.
- Merle, R., and Dolbow, J., 2002, "Solving thermal and Phase Change Problems with the Extended Finite Element Method," *Computational Mechanics*, **28**, 339–350. <http://dx.doi.org/10.1007/s00466-002-0298-y>.
- Moes, N., Dolbow, J., and Belytschko, T., 1999, "A Finite Element Method For Crack Growth Without Remeshing," *International Journal for Numerical Methods in Engineering*, **46**, 131–150.
- Morgan, K., 1981, "A Numerical Analysis of Freezing and Melting with Convection," *Computer M*, **28**, 275–284.
- Nedjar, B., 2002, "An Enthalpy-based Finite Element Method for Non-linear Heat Problems Involving Phase Change," *Computers & Structures*, **80**(1), 9 – 21. [http://dx.doi.org/10.1016/S0045-7949\(01\)00165-1](http://dx.doi.org/10.1016/S0045-7949(01)00165-1).
- Osher, S., and Fedkiw, R., 2001, "Level Set Methods: An Overview and Some Recent Results," *Journal of Computational Physics*, **169**(2), 463–502. <http://dx.doi.org/10.1006/jcph.2000.6636>.
- Osher, S., and Fedkiw, R., 2003, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag.
- Osher, S., and Sethian, J.A., 1988, "Fronts Propagating with Curvature-dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *Journal of Computational Physics*, **79**(1), 12 – 49. [http://dx.doi.org/10.1016/0021-9991\(88\)90002-2](http://dx.doi.org/10.1016/0021-9991(88)90002-2).
- Özişik, M.N., 1993, *Heat Conduction*, John Wiley & Sons, Inc.
- Postek, E.W., Lewis, R.W., and Gethin, D.T., 2008, "Finite Element Modelling of the Squeeze Casting Process," *International Journal of Numerical Methods for Heat and Fluid Flow*, **18**(3-4), 325 – 355.
- Rathjen, K., and Jiji, L., 1971, "Heat Conduction with Melting or Freezing in a Corner," *Journal of Heat Transfer, ASME*, **93**, 101–109.
- Reddy, J., 2006, *An Introduction to the Finite Element Method*, 3rd ed., McGraw-Hill.
- Shibanuma, K., and Utsunomiya, T., 2009, "Reformulation of XFEM Based on PUFEM for Solving Problem Caused by Blending Elements," *Finite Elements in Analysis and Design*, **45**(11), 806–816. <http://dx.doi.org/10.1016/j.finel.2009.06.007>.
- Vynnycky, M., and Kimura, S., 2007, "An Analytical and Numerical Study of Coupled Transient Natural Convection and Solidification in a Rectangular Enclosure," *International Journal of Heat and Mass Transfer*, **50**, 5204–5214.
- Yoo, H., and Tack, R.S., 1991, "Melting process with solid-liquid density change and natural convection in a rectangular cavity," *International Journal of Heat and Fluid Flow*, **12**(4), 365 – 374. [http://dx.doi.org/10.1016/0142-727X\(91\)90026-R](http://dx.doi.org/10.1016/0142-727X(91)90026-R).
- Zabaras, N., Ganapathysubramanian, B., and Tan, L., 2006, "Modelling Dendritic Solidification with Melt Convection using the Extended Finite Element Method," *Journal of Computational Physics*, **218**(1), 200–227. <http://dx.doi.org/10.1016/j.jcp.2006.02.002>.