A finite volume parallel adaptive mesh refinement method for solid-liquid phase change

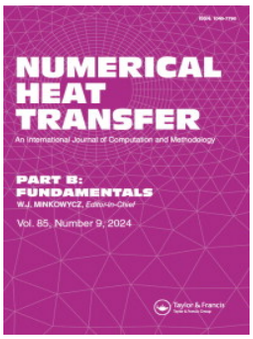
Kaaks, Bouke Johannes; Rohde, Martin; Kloosterman, Jan Leen; Lathouwers, Danny

DOI
10.1080/10407790.2024.2387841

Publication date
2024

Document Version
Final published version

Published in
Numerical Heat Transfer, Part B: Fundamentals



Taylor & Francis
Taylor & Francis Group

Numerical Heat Transfer, Part B: Fundamentals
An International Journal of Computation and Methodology

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/unhb20

A finite volume parallel adaptive mesh refinement method for solid-liquid phase change

Bouke Johannes Kaaks, Martin Rohde, Jan-Leen Kloosterman & Danny Lathouwers

To cite this article: Bouke Johannes Kaaks, Martin Rohde, Jan-Leen Kloosterman & Danny Lathouwers (08 Nov 2024): A finite volume parallel adaptive mesh refinement method for solid-liquid phase change, Numerical Heat Transfer, Part B: Fundamentals, DOI: 10.1080/10407790.2024.2387841

To link to this article: https://doi.org/10.1080/10407790.2024.2387841

Published online: 08 Nov 2024.

NUMERICAL HEAT TRANSFER, PART B: FUNDAMENTALS
https://doi.org/10.1080/10407790.2024.2387841

A finite volume parallel adaptive mesh refinement method for solid-liquid phase change

Bouke Johannes Kaaks, Martin Rohde, Jan-Leen Kloosterman, and Danny Lathouwers

Department of Radiation Science and Technology, Delft University of Technology, Delft, Netherlands

ABSTRACT

We present a finite volume adaptive mesh refinement method for solid-liquid phase change problems with convection. The refinement criterion consisted of three different error estimators for the solid-liquid interface, the flow field, and the temperature field respectively. For the solid-liquid interface, the cells undergoing phase change were refined based on the maximum difference in the liquid fraction over the cell faces. For the flow field and the temperature field, an error indicator was used based on the cell residual method. To maintain a high parallelization efficiency, a dynamic load balancing procedure was used. The adaptive mesh refinement strategy was verified through three different test cases, these are the gallium melting in both 2D and 3D cavities, and the molten salt reactor freeze valve. For all three cases, very good agreement was obtained between the adaptive mesh results and the reference solutions. In addition, more accurate results were obtained with the adaptive meshes compared to static meshes with a similar amount of mesh cells. This illustrated the potential of the current approach for developing computationally efficient numerical methods for solid-liquid phase change problems.

ARTICLE HISTORY
Received 4 December 2023
Revised 11 July 2024
Accepted 26 July 2024

KEYWORDS
Adaptive mesh refinement; residual error estimate; solid-liquid phase change

1. Introduction

Solid-liquid phase change problems are encountered in many practical applications, such as metallurgy [1], latent heat storage [2, 3], electronics cooling [4, 5], and the design of the Molten Salt Reactor (MSR) [6–11]. Over the last few decades, many numerical methods have been developed for solving solid-liquid phase change problems, which can be divided into two main families: multi-domain methods and single-domain methods [12, 13].

In multi-domain methods the liquid and solid phases are clearly separated and the interface movement is being governed by the Stefan condition. Multi-domain methods may be broadly separated into four categories: moving grid methods [14], where the mesh is deformed to follow the interface during its displacement, transformed grid methods [15, 16], where a mapping function is used to immobilize the moving interface on a uniform and fixed grid, front fixing methods [17], where an adaptable time step is used in order to have the interface always lie on a mesh point and front-tracking methods, where the interface position is explicitly tracked through a set of marker particles [18–20]. Multi-domain methods are also considered sharp interface methods, due to their ability to apply the solid and the liquid phase boundary conditions at the exact

CONTACT Danny Lathouwers d.lathouwers@tudelft.nl Department of Radiation Science and Technology, Delft University of Technology, Delft, Netherlands.

Nomenclature

Acronyms

AMR	adaptive mesh refinement
DLB	dynamic load balancing
MSR	Molten Salt Reactor
PDE	partial differential equation

Computational parameter

γ	generalized diffusion coefficient ($m^2\ s^{-1}$)
$\mathbb{L}$	characteristic length scale (m)
$\mathbb{T}$	characteristic time scale (s)
δ	distance from the cell center to the neighboring cell center (m)
d	vector pointing from the cell center to the neighboring cell center (m)
s	outward-pointing face area vector (m^2)
x	position vector (m)
$\mathcal{L}$	matrix coefficient corresponding to the linear component of the source term
ω	damping parameter
ϕ	generalized variable
ϕ_l	liquid fraction
b	small parameter to prevent division by zero
C	Darcy Constant ($kg\ m^{-3}\ s^{-1}$)
e_r	estimated numerical discretization error
F_{norm}	normalization factor
r_P	cell residual
res	residual
tol	tolerance
V_P	cell volume (m^3)

Other symbols

Δ	incremental difference
H, W	height, width (m)
x, y, z	cartesian coordinate system (m)

Physical quantity

g	gravitational acceleration ($m\ s^{-2}$)
H	volumetric enthalpy ($J\ m^{-3}$)
p	pressure (Pa)
t	time (s)
T	temperature (K)
u	velocity ($m\ s^{-1}$)
X_s^*	relative total solid fraction

Sub- and superscripts

$*$	normalized
^{i}	internal / at iteration 'i'
^{n}	at time step 'n'
$_\Omega$	over the computational domain
$_f$	at the cell face
$_l$	liquid
$_m$	at the melting point
$_P$	at the cell center
$_s$	solid
$_0$	initial
$_L$	left
$_{rel}$	relative
$_R$	right

Thermophysical property

β	thermal expansion coefficient (K^{-1})
λ	thermal conductivity ($W\ m^{-1}\ K$)
μ	dynamic viscosity (Pa s)
ρ	density ($kg\ m^{-3}$)
c_p	specific heat ($J\ kg^{-1}\ K^{-1}$)
L	latent heat of fusion ($kJ\ kg^{-1}$)

location of the interface. As such, they are generally more accurate than single-domain methods, where the interface is inevitably smeared out over a finite width (i.e. mushy zone, diffuse interface method). However, the drawback of these methods is that they are generally difficult to implement (especially as a general framework which is also suitable for solving complex multi-dimensional problems), due to the requirement of remeshing, coordinate transformation, or the explicit tracking of the interface.

In single-domain methods, the liquid and solid phases are described by a single set of partial differential equations (PDE), which are solved on the entire domain. Two different classes of single-domain methods can be distinguished: front-capturing methods and enthalpy methods. In front-capturing methods, the interface is implicitly followed by solving a separate PDE for the phase-field variable (i.e. phase-field methods [21, 22]) or the distance function (i.e. level set methods [23–25]). Front capturing methods allow a microscopic description of the interface, and are therefore typically applied for microscale phase change problems such as dendritic solidification or crystal growth. On the other hand, in enthalpy methods, no additional equation is solved for the solid-liquid interface. Instead, the solid-liquid phase change is accounted for by including the latent heat into the enthalpy-temperature coupling, resulting in a non-linear energy equation.

Special treatment is required for solving the non-linear energy equation, and currently three approaches are available. In the apparent heat capacity method [26–28], the heat capacity is modified around the melting point, in order to add the release of the latent heat to the energy balance. In the source-based enthalpy approach [29–31] the latent heat release is captured through a source term. Finally, in the 'linearized enthalpy approach' [32–35], the enthalpy is linearized around the latest temperature values, and the energy equation is iterated until convergence has been reached. Compared to the apparent heat capacity method and the source-based enthalpy approach, the 'linearized enthalpy approach has been shown to have superior performance in terms of a fast and accurate iterative procedure and inherent conservation of thermal energy [32–35]. Enthalpy methods are applied to macroscale phase change problems only, although recently a mesoscopic phase change model was proposed coupling a phase change model for calculating the solid microstructure with an enthalpy method for calculating the macroscale solid-liquid interface evolution [36].

In the present work, a single-domain enthalpy method was used based on the linearized enthalpy approach [35]. Compared to multi-domain approaches, single-domain approaches have the advantage of solving the same set of equations in the solid and the liquid phases, therefore not requiring coordinate transformations, grid generation, or the tracking of the interface. In addition, unlike the phase-field method or the level set method, these methods do not rely on complex thermodynamic derivations or the calculation of interface curvatures. Therefore, enthalpy methods are simpler to implement, require less human or computational efforts, and are applicable to a wide range of solid-liquid phase problems [12, 13, 15]. For this reason, enthalpy methods have been the most popular choice for modeling macroscale phase change phenomena in industrial applications. However, the drawback of these methods is their overall lower accuracy, especially compared to the multi-domain approaches. Therefore, a very fine mesh resolution may be required locally to properly resolve the recirculation zones in the fluid region (as a result of natural convection where the fluid flows up along a hot wall and down along the solid-liquid interface) and accurately capture the solid-liquid interface [37]. This will result in a considerable increase of computational time when using a uniform mesh, even for 2D cases.

To overcome these difficulties, recent investigations have explored the use of discontinuous Galerkin finite element methods (DG-FEM) [35, 38, 39], the use of an extended finite element approach [40–42] or the use of adaptive mesh refinement (AMR) [43, 44]. DG-FEM was explored due to the ability to easily implement high-order schemes that give high accuracy even on coarse meshes. However, it was found that even high-order schemes yielded a mesh convergence rate of O(h) due to their inability to resolve the discontinuity in the enthalpy field at the solid-liquid interface [35]. Whilst qualitatively similar results on a significantly coarser mesh were obtained for the gallium melting in a rectangular cavity problem using DG-FEM-based numerical methods as compared to the finite volume-based reference solution [35, 39], this was believed to be due to a more accurate resolution of the multicellular flow region (i.e. a natural convection driven flow containing multiple convection cells) rather than a more accurate capturing of the solid-liquid interface [35]. On the other hand, adaptive meshing methods can increase the overall accuracy of the results by locally and dynamically increasing the resolution in areas of interest, such as the solid-liquid interface and the recirculation zones in the fluid region.

Adaptive meshing strategies have successfully been developed for a variety of applications, such as large eddy simulations of turbulent flow [45, 46], vortex-dominated flows [47, 48], multiphase flow [49, 50], and ship hydrodynamics [51, 52]. However, for solid-liquid phase change problems with convection, few contributions are reported in the literature. *Palle et al* [53] presented a finite element method with adaptive mesh refinement for melting and solidification problems using the Zienkiewicz and Zhu error estimator [54]. *Lan et al* [55] developed an adaptive mesh refinement algorithm for solidification problems using the finite volume method, employing the normalized gradient or an estimate for the normalized truncation error as the

refinement criterion. However, it was difficult to establish a universal rule for a cost-effective refinement procedure using the aforementioned approach, since further refinement (applying a stricter tolerance for the chosen error estimator) did not always result in a further decrease of the error. Using a similar gradient-based error estimation approach, *Mencinger et al* [56] presented an r-adaptive finite volume method for the numerical simulation of melting in a 2D cavity. A simple moving finite element mesh technique was presented in *Tenchev et al* [57], which concentrates nodes in the vicinity of the phase change front. In addition, several finite element remeshing methods have been developed based on an equidistribution of the edge interpolation error, using an estimation of the Hessian [43, 58, 59] or a gradient recovery approach [44] as the basis for the error estimate and the construction of the metric tensor used to modify the mesh. Compared to adaptive mesh refinement, adaptive remeshing can lead to a more optimal distribution of the error (and therefore requiring less elements). However, the grid generation and the projection of variables unto the new mesh also present a more significant computational overhead compared to the adaptive mesh refinement (and coarsening) procedure.

In the present work, we present an adaptive finite volume mesh refinement algorithm for melting and solidification problems using OpenFOAM. The cell residual error estimate of *Jasak et al* [60, 61] is used for estimating the error in the solution for the flow field and the temperature field, which has proven to be a more accurate error estimate compared to the traditional truncation-based error estimates for the finite volume method. In addition, the elements in the vicinity of the solid-liquid interface are refined based on the maximum difference in the liquid fraction over the cell faces. With these refinement criteria, we expect to obtain a locally high resolution in both the solid-liquid interface and the critical flow and heat transfer regions, hereby achieving a high numerical accuracy at an affordable computational cost. To ensure a high parallelization efficiency of our code, changes to the mesh are followed by a dynamic load balancing step, using the dynamic mesh library of *Rettenmeier et al* [62]. The performance of the proposed adaptive mesh refinement algorithm was demonstrated for three test cases: Gallium melting in a 2D cavity, Gallium melting in a 3D cavity, and an MSR freezeplug design. This is the first time that the cell residual error estimate of *Jasak et al* has been used as a refinement criterion for solid-liquid phase change simulations, and that an adaptive mesh refinement strategy for solid-liquid phase change problems has been applied to a more complex problem involving both solid-liquid phase change and conjugate heat transfer. The remainder of this paper is organized as follows: in Section 2 we describe our mathematical model for modeling melting and solidification. In Section 3 we describe our numerical discretization and matrix solution procedure, the adaptive mesh refinement implementation in OpenFOAM, and our refinement criterion. In Section 4 we demonstrate the performance of our adaptive mesh refinement against 3 cases: gallium melting in a 2D cavity, gallium melting in a 3D cavity, and the MSR freezeplug. Finally, in Section 5 we present our conclusions.

2. Governing equations

The coupled energy and momentum equations are solved numerically using the finite volume method. Here, the energy transport equation is written as:

$$\frac{\partial H}{\partial t} + \nabla \cdot (\boldsymbol{u} H) = \nabla \cdot (\lambda \nabla T), \tag{1}$$

where H is the volumetric enthalpy, t is the time, $\boldsymbol{u}$ is the velocity, λ is the thermal conductivity, and T is the temperature. The energy transport equation contains two unknowns, the volumetric enthalpy in the accumulation and the convection terms and the temperature in the diffusion term, coupled through the enthalpy-temperature relationship. For most heat transfer problems, the enthalpy-temperature relationship is smooth and the enthalpy-temperature gradient is defined

throughout the domain, allowing one of the unknowns (enthalpy or temperature) to be eliminated from the equation. However, for solid-liquid phase change the enthalpy-temperature relationship is non-smooth as a result of the latent heat release at the melting point. Assuming isothermal phase change and constant thermophysical properties in each phase, the resulting non-linear enthalpy-temperature coupling is written as

$$T(H) = \begin{cases} \dfrac{H}{\rho_s c_{p,s}}, & H \leq \rho_s c_{p,s} T_m \\ T_m, & \rho_s c_{p,s} T_m < H < \rho_s c_{p,s} T_m + \rho_{l/s} L \\ T_m + \dfrac{H - (\rho_s c_{p,s} T_m + \rho_{l/s} L)}{\rho_l c_{p,l}}, & H \geq \rho_s c_{p,s} T_m + \rho_{l/s} L. \end{cases} \tag{2}$$

where ρ_s is the density of the solid and ρ_l of the liquid phase, $c_{p,s}$ is the heat capacity of the solid and $c_{p,l}$ of the liquid phase, T_m is the melting point, and L is the latent heat of the solid-liquid phase change. Here, following the approach of *Galione et al* [63], the latent heat release is scaled with the solid or liquid density depending on whether the material is melting or freezing.

Solving the non-linear energy equation is challenging because the enthalpy-temperature derivative is undefined at the melting point [see Eq. (2)]. Therefore, a special numerical solution procedure is required. In this work, the 'linearized enthalpy approach' was used [35]. Unlike the source-based enthalpy approach, where the total enthalpy is split into a latent and a sensible contribution of which the latter is included in the energy equation as a source term, the 'linearized enthalpy approach directly solves for the total enthalpy [31, 34, 35]. In the 'linearized enthalpy approach', the total volumetric enthalpy is linearized around the latest known temperature value:

$$H^{n+1,i+1} = H^{n+1,i} + \frac{dH}{dT}(T^{n+1,i+1/2} - T^{n+1,i}), \tag{3}$$

where $n+1$ refers to the latest time step, $i+1$ refers to the newest iteration, i indicates the previous iteration, and $i+1/2$ denotes the use of an intermediate value, calculated from the solution of the 'linearized enthalpy equation' [see Eq. (6)]. Here, the following form for the enthalpy-temperature derivative is used:

$$\frac{dH}{dT} \approx \begin{cases} (1-\phi_l)\rho_s c_{p,s} + \phi_l \rho_l c_{p,l}, & T \neq T_m \\ \dfrac{\rho_l L}{2\epsilon}, T = T_m \end{cases} \tag{4}$$

where ϵ is a small parameter and ϕ_l is the liquid fraction:

$$\phi_l = \begin{cases} 0, & T \leq T_m \\ 1, & T > T_m. \end{cases} \tag{5}$$

The enthalpy-temperature derivative is exact, except at the melting point where the dirac delta due to the release of latent heat is approximated by a large value. The use of an approximation is justified since it has been shown previously that this does not affect the accuracy of the iterative solution procedure [33].

Performing the second-order accurate BDF2 time integration and substituting the linearized enthalpy yields the 'linearized enthalpy equation'

$$\begin{aligned} \frac{dH}{dT}\left(\frac{3T^{n+1,i+1/2}}{2\Delta t} + \nabla \cdot (\boldsymbol{u} T^{n+1,i+1/2})\right) - \nabla \cdot (\lambda \nabla T^{n+1,i+1/2}) \\ = \frac{dH}{dT}\left(\frac{3T^{n+1,i}}{2\Delta t}\right) - \frac{3H^{n+1,i} - 4H^n + H^{n-1}}{2\Delta t}. \end{aligned} \tag{6}$$

Here, the 'sensible enthalpy only' formulation was used for the heat convection term ([64]).

The 'linearized enthalpy equation' is iterated until convergence. At each iteration, the volumetric enthalpy at the cell centers is updated and the temperature is recalculated according to the enthalpy-temperature relationship [see Eq. (2)]:

$$H^{n+1,i+1} = H^{n+1,i} + \omega \frac{dH}{dT}\left(T^{n+1,i+1/2} - T^{n+1,i}\right) \tag{7a}$$

$$T^{n+1,i+1} = T(H^{n+1,i+1}), \tag{7b}$$

where $\omega \leq 1$ is a damping parameter to ensure stability.

The non-linear enthalpy-temperature iterations are terminated when the following convergence criterion is reached:

$$\max\left[res, \text{maxval}\left(T^{n+1,i+1} - T^{n+1,i}\right)\right] < tol. \tag{8}$$

In this work, $\text{tol} = 10^{-6}$, which was found to be a good compromise between accuracy and speed of computation [35]. Here, *res* is the scaled residual of the original formulation of the transport equation [see Eq. (1)] which is calculated using the known solution values after each iteration:

$$res = \frac{\int_\Omega \left(\left(\frac{\partial H}{\partial t} + \nabla \cdot (\boldsymbol{u}H) - \nabla \cdot (\lambda \nabla T)\right)\right) dV}{\int_\Omega \frac{H}{\Delta t} dV}. \tag{9}$$

As such, $res \rightarrow 0$ is a measure of the thermal energy conservation. In addition, for $(T^{n+1,i+1} - T^{n+1,i}) \rightarrow 0$, the linearization terms cancel out and the 'linearized enthalpy' equation is equal to the energy equation in its conservative form [35]. Therefore, the solution to the 'linearized enthalpy equation' approaches the solution to the original enthalpy transport equation in conservative form upon convergence.

For the momentum equation, we consider incompressible flow and a Newtonian fluid with constant viscosity. The *Darcy* source term approach is used for modeling the velocity attenuation at the solid-liquid interface. This approach is most commonly used and has demonstrated better performance compared to other approaches such as the switch-off technique which is prone to numerical instabilities or the variable viscosity technique which lacks physical basis [13, 30, 31]. Natural convection is treated through the Boussinesq approximation. As such, the momentum equation reads

$$\frac{\partial(\rho_l \boldsymbol{u})}{\partial t} + \nabla \cdot (\boldsymbol{u} \otimes (\rho_l \boldsymbol{u})) = \nabla \cdot \left[\mu\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right)\right] - \nabla p \ + \rho_l \boldsymbol{g} \beta (T - T_m) - C \frac{(1-\phi_l)^2}{\phi_l^3 + b} \boldsymbol{u}. \tag{10}$$

Here, C is a large parameter (10^6 in this work) and b is a small parameter to avoid division by zero (10^{-3} in this work);[1] μ is the dynamic viscosity, p is the pressure, $\boldsymbol{g}$ is the gravitational acceleration, and β is the thermal expansion coefficient.

On a final note, we would like to mention that our modeling approach makes use of the following assumptions:

1. Incompressible flow with a Newtonian viscosity behavior.
2. Constant and isotropic thermophysical properties in each phase.
3. Volume change occurring during solid-liquid phase change is considered in the solution of the energy equation only, and not for the solution of the momentum equation (approach of *Galione et al*, [63]).
4. Validity of the *Boussinesq* approximation for modeling solid-liquid phase change.
5. No air bubbles or impurities were present.

3. Numerical methodology

3.1. Adaptive mesh refinement

An h-adaptive refinement approach was used to dynamically obtain high mesh resolution only in the regions where the estimated numerical error exceeds a certain threshold. To this end, a custom-built OpenFOAM solver for solid-liquid phase change simulations was coupled to the adaptive mesh refinement library developed by *Rettenmeier et al* [62], of which an OpenFOAM v9 port is available in the blastFOAM project [65]. This library presents a major enhancement of the standard AMR functionality present in OpenFOAM, introducing AMR for arbitrary polyhedral cell shapes and 2D and 2.5D (i.e. axisymmetric) cases, whereas the standard version of OpenFOAM only supports AMR for hexahedral cells in 3D. In addition, the AMR library of *Rettenmeier et al* features dynamic load balancing, maintaining a high parallelization efficiency during adaptive mesh refinement and therefore greatly enhancing the computational efficiency.

When solving a transient problem using parallel computing, AMR will change the load on the processors. The processor with the largest load becomes a bottleneck, which may significantly reduce the parallelization efficiency [62]. Using the dynamic load balancing (DLB) implementation of *Rettenmeier et al* [62], the cells were redistributed among the processors following a refinement step, based on a user-prescribed maximum imbalance. To ensure that refined sibling cells remain on the same sub-domain, the refinement history was imposed as constraint during both the initial decomposition and during the redistribution. *Rettenmeier et al* previously showed that simulation times were halved for 125 processors when using DLB compared to a non-balanced simulation for a dam-break benchmark case using AMR [62]. It can be argued that for solid-liquid phase change simulations, where the region of interest (i.e. the solid-liquid interface) migrates throughout the entire domain, the gain in parallelization efficiency obtained through DLB is even higher than for the dam-break case.

The AMR library in OpenFOAM consists of three classes, these are the refinement engine (which contains the permanent mesh data, the functions to create the connectivity structures of the mesh, and the functions to select the candidates for refinement or coarsening), the mesh cutter (which splits the parent cells into 8 or 4 child cells for 3D or 2D cases respectively), and the refinement tree containing the history of refinement (following an octree and quadtree approach for the 3D and 2D cases respectively) [46]. Here, cell level = 0 corresponds to the original cells, cell level = 1 corresponds to the first level of refinement etcetera. See Figure 1 for a schematic of the adaptive mesh refinement. After each refinement step, the solution needs to be mapped between the parent and the child cells. OpenFOAM adopts a pseudo-staggered approach where variables (e.g. velocity and pressure) are stored as volume fields and defined at the cell centers,

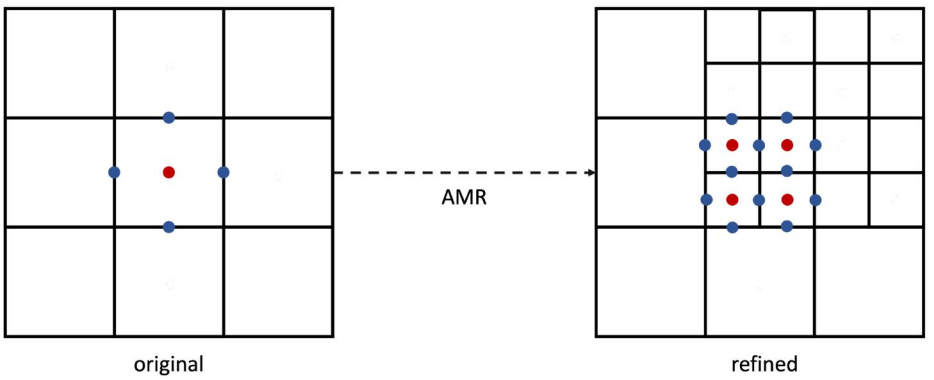


Figure 1. Schematic of adapive mesh refinement (in 2D, following the quadtree approach). The variables are stored as volume fields at the cell centers (red dots) and the fluxes are stored as surface fields at the face centers (blue dots).

whereas the convective flux is defined as a surface field and stored at the face centers [62]. The mapping of the cell-centered volume fields is straightforward: for each refinement step, the child-cells receive the cell-centered value of the parent cell whereas for a coarsening step, the volume average of the child cells' cell-centered values is set on the parent cell centers [62]. The mapping of the convective flux is more complicated however, since new faces internal to the parent cell are not related to any master face. A good approximation of the convective flux on those fields is obtained from an arithmetical averaging of the face-centred velocity field $\mathbf{u_f}$ of the four adjacent neighboring faces (which receive their values from the master face) [49, 62]. Due to the non-conservative mapping of the surface fields, a pressure correction step is introduced after each AMR step to ensure a divergence free mass flux field.

3.2. Refinement criterion

In this work, a refinement criterion consisting of a combination of three different sub-criteria is proposed. These are based on the variation in liquid fraction (to refine the solid-liquid interface), and the numerical discretization errors of the velocity and temperature fields (to accurately capture the recirculation regions and the temperature gradients near the solid wall in case of conjugate heat transfer).

To obtain a sharp solid-liquid interface, the cells undergoing a rapid variation of the liquid fraction are flagged for refinement. This is done based on the maximum difference of the liquid fraction over the cell faces. Subsequently, the cells are refined up to the maximum refinement level. The refinement criterion for the solid-liquid interface is thus expressed as:

$$\begin{cases} \text{if } \left(\max([[\phi_l]]_f) > 0.01 \quad \text{cell level} < \text{maximum cell level}\right) \text{ refine} \\ \text{if } \left(\max([[\phi_l]]_f) < 0.001 \quad \text{cell level} >= 1\right) \text{ coarsen} \end{cases} \tag{11}$$

Here, $[[\cdot]]_f$ is the jump operator, defined as the face value of the cell minus the face value of the adjacent neighboring cell.

To accurately resolve the flow field and the temperature field, the numerical discretization errors are estimated based on the cell residual error estimate developed by *Jasak et al* [60, 61, 66], which has proven to be a more accurate error estimator compared to the traditional truncation based error estimates for the finite volume method. The cell residual, similar to the residual error estimate for the finite element method, uses the inconsistency between the face interpolated values and the cell volume integration of the finite volume method as a measure of the numerical discretization error (see Figure 2 for a graphical representation of this inconsistency).

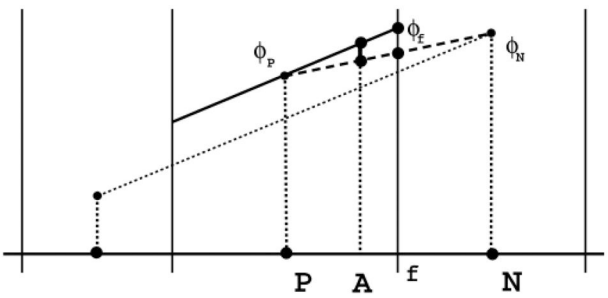


Figure 2. Inconsistency between the face interpolation and the cell volume integration, taken from [61]. here, P is the Center of the control volume, N is the center of the neighboring cell and f is the face between cell P and neighbor N. The face interpolation (the dashed line, assuming a linear variation between P and N) and the cell volume integration (solid line, assuming a linear variation over the control volume) predict a different value of the generic variable ϕ at an arbitrary point a inside the cell.

For the steady-state transport equation of a generic conserved quantity ϕ (scalar or vector), the cell residual is calculated as [60]:

$$\begin{aligned} r_{P,steady}(\phi) &= \int_{V_P}[\nabla\cdot(\mathbf{u}\phi) - \nabla(\gamma\nabla\phi) - S - \mathcal{L}\phi_P]dV \\ &= \sum_f \left(\mathbf{s}\left[\mathbf{u}_f\phi_f - \gamma_f(\nabla\phi)_f\right] - SV_P - \mathcal{L}\phi_P V_P\right) \end{aligned} \tag{12}$$

Here, γ is the diffusion coefficient and $S(\phi) = \mathcal{S} + \mathcal{L}\phi$ represents the linear decomposition of the source term. The subscript 'P' denotes the center of the control volume. To obtain the cell residual, the face values are calculated assuming a linear variation of ϕ over the control volume (instead of through a linear interpolation between the cell center values ϕ_P and ϕ_N as is the case in standard finite volume discretization, see Figure 2):

$$\phi_f = \phi_P + (\mathbf{x_f} - \mathbf{x_P})\cdot\nabla\phi_P, \quad (\nabla\phi)_f = \nabla\phi_P. \tag{13}$$

Here, $\mathbf{x}$ is the position vector.

In transient calculations, the numerical discretization accuracy is determined through both the temporal and spatial discretization. Decisions on the mesh refinement will be based on the estimate of the spatial error only, whilst the temporal error estimate can be used for the purpose of automatic time step control [66]. In this work, automatic time step control is based on the maximum value of the Courant number, to ensure stability of the simulation. No sub-cycling strategy is employed, therefore a single-time step is used across all the levels. The time step is therefore typically determined by the smallest cells. Taking the transient term into account, the cell residual for the transient transport equation is calculated as [66]:

$$r_{P,transient}(\phi) = \int_{V_P}\frac{\partial\phi}{\partial t}dV + r_{P,s}(\phi). \tag{14}$$

Since $r_{P,t}$ has the dimensions of $\frac{[\phi][\mathbb{L}]^3}{[\mathbb{T}]}$, with $\mathbb{L}$ the characteristic length scale and $\mathbb{T}$ the characteristic time scale, an appropriate normalization is needed to obtain the magnitude of the error [60]. The normalization is based on the characteristic convection and diffusion transport for the control volume. The normalization factor is calculated as $F_{norm} = F_{conv} + F_{diff} + \mathcal{L}$, where $F_{diff} = \frac{1}{V_P}\sum_f\left[|\mathbf{s}|\frac{\gamma f}{|\mathbf{d}|}\right]$ (i.e. approximation of the total diffusion transport coefficient) and $F_{conv} = \frac{1}{V_P}\sum_f max(\mathbf{u}_f\cdot\mathbf{s}, 0)$ (i.e. estimation of the convective transport coefficient). Here, $\mathbf{s}$ is the outward-pointing face area vector and $\mathbf{d}$ is the vector pointing from the cell center to the neighboring cell center. Since the error contribution from the linear part of the source term ($\mathcal{L}\phi_P$) scales with the error in ϕ_P, the matrix coefficient corresponding to the linear part of the source ($\mathcal{L}$) should be included into the normalization. The residual error estimate (REE) now becomes:

$$e_r(\phi) = \frac{r_{P,t}(\phi)}{V_P F_{norm}}. \tag{15}$$

The above form of the residual error estimate gives the absolute error estimate. However, for transient solid-liquid phase change simulations, the natural convection loop can strengthen (or weaken) over time and therefore it is difficult to specify an appropriate absolute error as a refinement criterion for the flow field. Instead, the absolute error in the flow field is normalized using the maximum absolute velocity value in the domain ($\max(|\mathbf{u}|)$). To avoid attributing too much importance to very low-velocity values in the early stages of the simulation, the normalization factor's lower bound is set at the absolute velocity value for which $Pe = 1$, i.e. $\frac{\alpha}{\mathbb{L}}$ with α the thermal diffusivity and $\mathbb{L}$ the characteristic length scale. Please note that the cell residual for the velocity is a vector, whereas OpenFOAM only supports scalar refinement criteria. As such, the Euclidean norm of the cell residual is used. The final form of the relative residual error estimate

for the flow field is:

$$e_{r,rel}(|\mathbf{u}|) = \frac{|r_{P,t}(\mathbf{u})|}{V_P \times F_{norm} \times \max(\max(|\mathbf{u}|), \alpha/\mathbb{L})}. \tag{16}$$

The estimated absolute numerical discretization error in the temperature field is normalized using the temperature difference over the domain:

$$e_{r,rel}(T) = \frac{|r_{P,t}(T)|}{V_P \times F_{norm} \times (\max(T)\text{-}\min(T))}. \tag{17}$$

If the estimated error exceeds the tolerance level, the cells are refined until the maximum refinement level is reached. Vice versa, the cell is coarsened once the estimated error is smaller than one fifth of the error tolerance:

$$\begin{cases} \text{if } (e_{r,rel}(|\mathbf{u}|, T) > tol \quad \text{cell level} < \text{maximum cell level}) \text{ refine} \\ \text{if } (e_{r,rel}(|\mathbf{u}|, T) < 1/5\ tol \quad \text{cell level} >= 1) \text{ coarsen} \end{cases} \tag{18}$$

The three adaptive mesh refinement criteria for the solid-liquid interface [see Eq. (11)], the velocity field and the temperature field [see Eq. (18)] are evaluated simultaneously. For a cell to be refined, either one of the refinement criteria needs to be true. For a cell to be coarsened, both the coarsening criteria need to be true. In general, since the velocity in the solid-liquid interface is equal to zero, the refinement criteria take effect in different parts of the domain.

3.3. Numerical solution procedure

The coupled energy and momentum equations were implemented in OpenFOAM v9 and solved using the finite volume method. The diffusive term was discretized using the second-order linear differencing scheme, and the convective terms were discretized using OpenFOAM's second-order *linearUpwind* scheme (which obtains second-order accuracy by applying an explicit gradient-based correction to the standard upwind scheme). The time derivatives were discretized using the second-order BDF2 formulation. To reduce the computational overhead from the adaptive mesh refinement, the mesh update was performed once per every five time steps. To ensure a smooth transition between the different refinement levels and reduce the numerical discretization error due to mesh skewness, four buffer layers were used between each refinement level based on the recommendation of *Rettenmeier et al*, who suggested the use of at least two buffer layers [62]. Please note that a minimum of one buffer layer is automatically enforced by the refinement engine which restricts the refinement level difference to neighboring cells to one, based on the one-irregularity constraint.

The resulting matrix equations are solved using GAMG (with Gauss-Seidel smoother) for the pressure equation and the stabilized bi-conjugate gradient matrix solver (PBiCGStab) (with DILU preconditioner) for the momentum and energy equations. The matrix tolerances are set to 10^{-8} for the pressure and pressure correction equations, (with an additional relative tolerance of 0.05 for solving the initial pressure correction steps), 10^{-12} for the momentum equation and 10^{-10} for the energy equation. To solve the velocity-pressure coupling, 5 PISO steps are used. Finally, three outer corrector steps are used to solve the coupling between the energy and the momentum equations.

The full solution algorithm can be summarized by the following steps (Figure 3) :

1. Evaluate the refinement criterion, update the mesh and perform the pressure correction to compensate for the non-conservative mapping of the surface fields, ensuring a divergence free mass flux field (once per every five time steps).
2. Solve the momentum equation to obtain the velocity predictor.

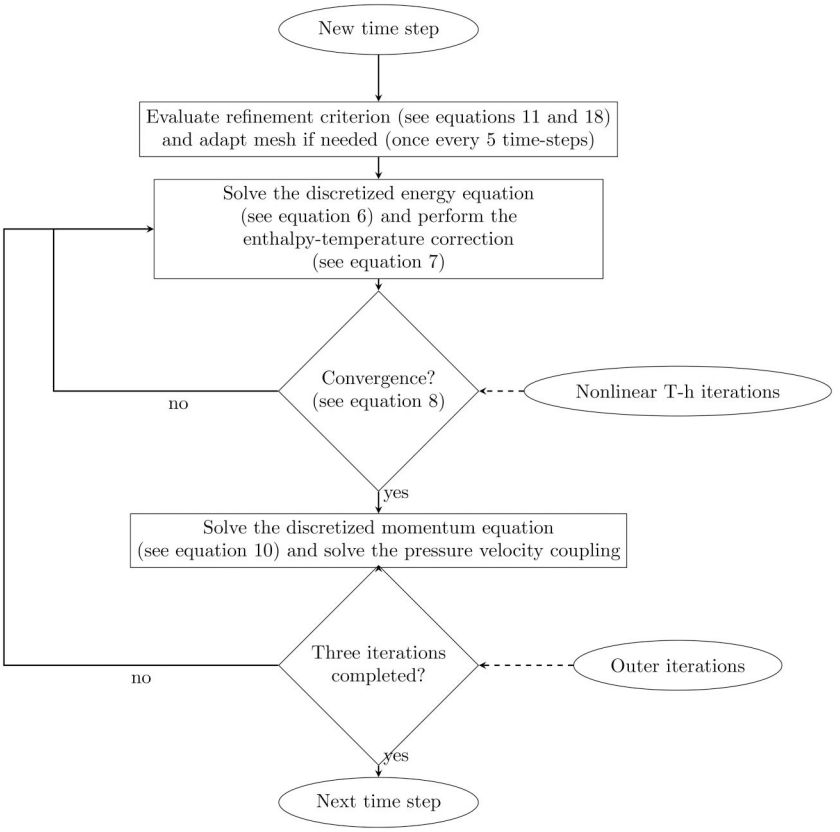


Figure 3. Flowchart of the solution algorithm, including the adaptive mesh refinement step, non-linear temperature-enthalpy iterations and the coupling of the energy and the momentum equations.

3. Solve the energy equation through a series of non-linear enthalpy-temperature iterations until convergence is reached.
4. Solve the pressure equation and perform the velocity correction steps (in this work, 5 velocity corrector steps are used).
5. Repeat steps 2-4 until the total number of outer iterations has been reached (in this work, 3 outer correctors are used).

4. Results and discussion

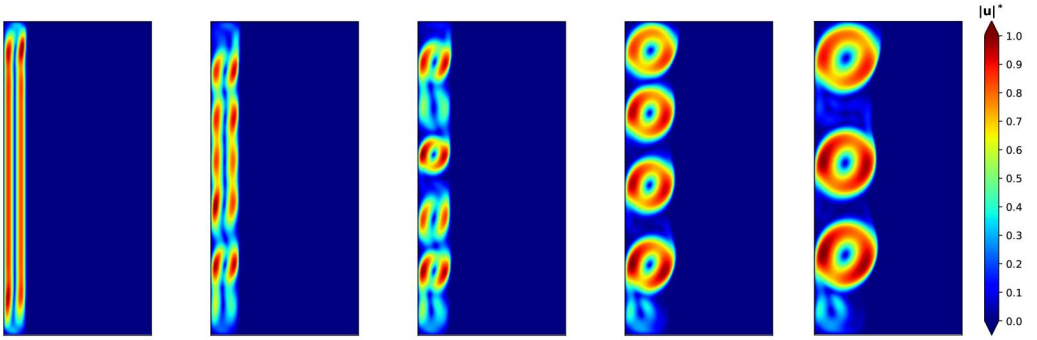
4.1. Melting of gallium in a 2D cavity

The first test case is the melting of gallium in a 2D rectangular enclosure of $L \times H = 88.9 \text{ mm} \times 63.5 \text{ mm}$. The initial temperature is $T_0 = 301.3$ K and at $t = 0$ s, the left wall is suddenly heated to $T_L = 311.0$ K with the right wall being kept at $T_R = 301.3$ K. The bottom and top walls are considered to be adiabatic. Due to the multicellular flow pattern, this problem is known to be sensitive to the mesh size, requiring a high spatial resolution to correctly predict the number of flow cells at each given time step [37]. For this reason, the melting of gallium in a 2D rectangular cavity is a challenging first test case for the adaptive mesh refinement technique presented in this paper.

Results are presented at five different times (i.e. 20 s, 32 s, 42 s, 85 s, 120 s), using both a uniform mesh of 1120×800 quadrilateral cells (i.e. the reference mesh) and an adaptive mesh. The

Table 1. Thermophysical properties of gallium [37].

Property	Solid	Liquid
Density [kg m^{-3}]	6093	6093
Specific heat capacity [J kg^{-1}]	381.5	381.5
Thermal conductivity [W m^{-1} K^{-1}]	32	32
Latent heat [J kg^{-1}]	80160	
Melting Temperature [K]	302.78	
Thermal expansion coefficient [1/K]	$1.2 \cdot 10^{-4}$	
Dynamic viscosity [Pas]	$1.81 \cdot 10^{-3}$	



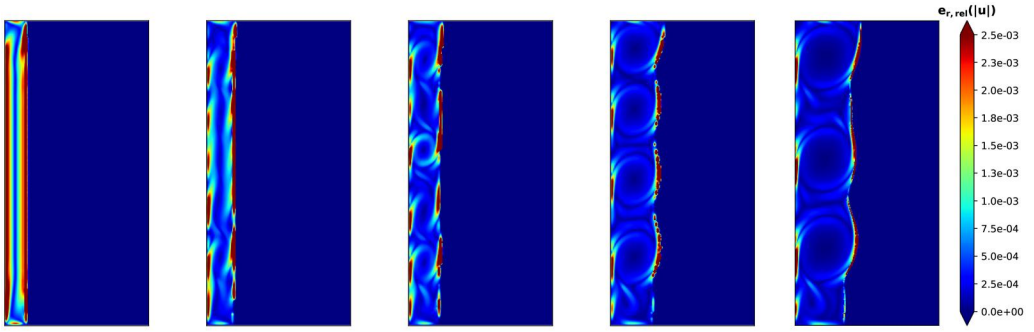
(*a*) 20 s (*b*) 32 s (*c*) 42 s (*d*) 85 s (*e*) 120 s

Figure 4. Normalized absolute velocity contour plots of gallium melting in a 2D cavity. Results were obtained using the uniform reference grid of 1120 x 800 elements.

adaptive mesh consists of a 70×50 coarse mesh at cell level 0 with a maximum local refinement level of 4. At the highest cell level, the resolution therefore corresponds to that of the uniform reference mesh. The tolerance level for the estimated numerical discretization error in the flow field was set to $e_{r,rel}(|u|) < 0.0025$. No refinement criterion was imposed for the numerical discretization error in the temperature field. Simulations were performed for four progressively coarser uniform meshes (with 560×400, 280×200, 140×100, and 70×50 elements respectively), in order to better quantify the performance of the adaptive mesh. For all cases, adaptive time stepping was used based on a maximum Courant number of $Co = 3$, with the maximum time step size set to $\Delta t = 0.025$ s. The thermophysical properties of gallium are given in Table 1.

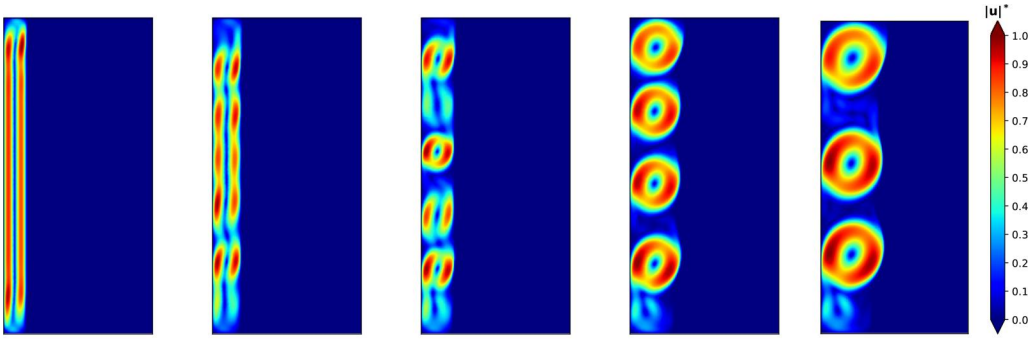
Figure 4 shows the solution for the normalized absolute velocity $|u|^* = \frac{|u|}{\max(|u|)}$, obtained with the uniform mesh of 1120×800 hexahedral elements. Here, $\max(|u|)$ corresponds to the maximum absolute velocity. At all times, both the number and the position of the convection cells correspond to those predicted by *Hannoun et al* [37] for the chosen simulation times (i.e. 20 s, 32 s, 42 s, 85 s). As such, the current numerical solution is in good agreement with the reference solution. Figure 5 shows the estimated relative numerical discretization error in the flow field, obtained using the element residual error estimate of *Jasak et al* [60]. Consistent with expectations, the largest errors were observed near the left boundary, the solid-liquid interface, and around the boundaries of the flow cells (where the highest vorticity is to be expected). Please note that these errors correspond to the estimated discretization error at each given time step, and not to the total accumulated numerical discretization error throughout the numerical simulation (to obtain the latter, one would have to solve the error transport equation).

Figure 6 shows the solution for the normalized absolute velocity $|u|^*$, obtained with the adaptive mesh. A very good agreement was observed between the results with the (significantly smaller) adaptive mesh and the results on the uniform mesh of 1120×800 elements, demonstrating the potential of the proposed adaptive mesh refinement approach. Figure 7 shows the cell level for the adaptive mesh. The highest degree of refinement was found near the left boundary,



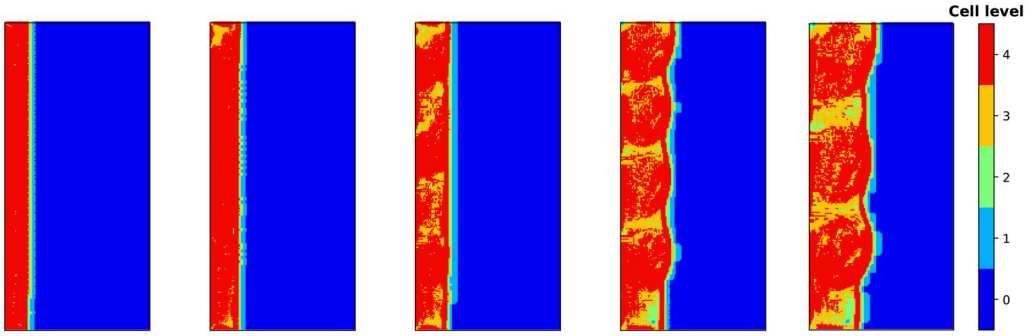
(*a*) 20 s (*b*) 32 s (*c*) 42 s (*d*) 85 s (*e*) 120 s

Figure 5. Estimated relative error in the absolute velocity for gallium melting in a 2D cavity. Results were obtained using the uniform reference grid of 1120 x 800 elements.



(*a*) 20 s (*b*) 32 s (*c*) 42 s (*d*) 85 s (*e*) 120 s

Figure 6. Normalized absolute velocity contour plots of gallium melting in a 2D cavity. Results were obtained using the adaptive mesh grid.



(*a*) 20 s (*b*) 32 s (*c*) 42 s (*d*) 85 s (*e*) 120 s

Figure 7. Cell levels of adaptive mesh grid for gallium melting in a 2D cavity.

the solid-liquid interface, and around the boundaries of the vortices. The pattern of refinement qualitatively matches the adapted meshes obtained by *Belhamadia et al* [44] who used a finite element higher-order reconstruction approach for a posteriori error estimation, indicating the element residual error estimate for the finite volume method predicts a similar distribution of the numerical error. Table 2 shows the total number of cells for each time, as well as the cell distribution (both as a fraction of the total number of cells and as a fraction of the total volume).

Table 2. Total number of cells and distribution of the different cell levels (% cells/% volume) for the 2D gallium melting case.

Time	Total number of cells	Level 0	Level 1	Level 2	Level 3	Level 4
20 s	50,681	6.4%/92.9 %	0.2%/0.8%	0.8%/0.7%	2.6%/0.6%	90.0%/5.1%
32 s	55,901	5.6%/91.4%	0.3%/1.1%	0.6%/0.6%	5.5%/1.4%	87.9%/5.5%
42 s	50,498	6.2%/90.1%	0.4%/1.4%	0.8%/0.7%	15.1%/3.4%	77.5%/4.4%
85 s	65,141	4.7%/87.1%	0.2%/1.0%	0.8%/0.8%	19.0%/5.5%	75.3%/5.5%
120 s	69,254	4.3%/84.4%	0.2%/1.2%	1.1%/1.4%	25.0%/7.7%	69.4%/5.4%

Table 3. L^2 Relative difference of the adaptive mesh and four progressively coarser uniform grids (with 560×280, 280×200, 140×100 and 70×50 elements respectively) against the reference solution (with a uniform grid of 1120×560).

Time	$L^2_{\lvert u\rvert}$ (AMR)	$L^2_{\lvert u\rvert}$ (560×400)	$L^2_{\lvert u\rvert}$ (280×200)	$L^2_{\lvert u\rvert}$ (140×100)	$L^2_{\lvert u\rvert}$ (70×50)
20 s	0.0100	0.0254	0.0827	0.2367	0.4529
32 s	0.0398	0.0343	0.1401	0.3631	0.4757
42 s	0.0357	0.0596	0.5769	0.6502	0.6976
85 s	0.0500	0.0743	0.2040	0.7140	0.7074
120 s	0.0469	0.1030	0.1101	0.3330	0.7777

These results show that whilst the mesh refinement occurs in only a small part of the domain (i.e. where the gallium is melting), the refined cells are dominated by the two finest levels (with level 1 and level 2 predominantly used as buffer cells between the parent cells and the two finest levels). This was attributed to the use of a relative coarse background mesh, whilst resolving the multicellular flow patterns requires a very fine mesh, as was shown by *Hannoun et al* [37].

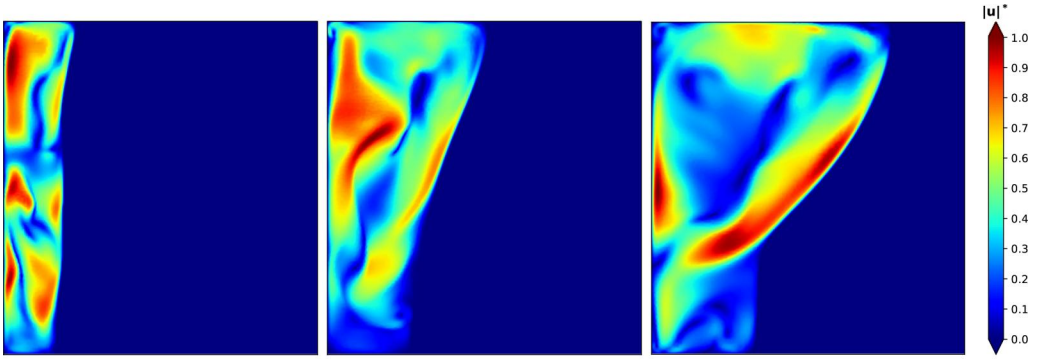
Table 3 shows the L^2 relative difference (against the reference solution) for the 5 different times, evaluated for the absolute velocity field and calculated for five different meshes, i.e. the adaptive mesh and four different uniform meshes of progressively decreasing resolution. In addition, the number of cells for the adaptive mesh at each evaluated time step is given. The finest uniform mesh of size 1120×800 elements is used as the reference solution. The L^2 relative difference is given by:

$$\left(\frac{\int_V \left[|u|^* - |u|^*_{ref} \right]^2 dV}{\int_V \left[|u|^*_{ref} \right]^2 dV} \right)^{1/2} \tag{19}$$

To calculate the L^2 relative difference, the solution data from the adaptive mesh or one of the uniform coarser meshes was interpolated onto the reference mesh using a cubic 2D interpolant, such that the locations of the data points matched. Subsequently, the integrals were numerically approximated using the trapezium method. The L^2 relative differences between the adaptive solution and the uniform reference solution are below 5% for all times. The number of cells for the adaptive mesh are between 50000 and 70000, which is a similar mesh size as the uniform 280×200 mesh. Compared to the 280×200 uniform mesh, the relative difference between the adaptive mesh and the reference solution is much smaller for all times. For the coarsest uniform meshes of 140×100 and 70×50 elements, very large differences were obtained, due to the inability of the coarsest meshes to properly resolve the multicellular flow patterns, leading to an incorrect prediction of the number of convective cells. As such, the analysis of the L^2 relative differences demonstrates that our adaptive mesh refinement approach can lead to similar numerical accuracy (as the uniform 1120×800 mesh) at significantly lower computational cost (4 h and 40 min to simulate 280s with AMR and DLB on 12 cores as opposed to 47h and 17.3 min for the uniformly refined mesh, therefore resulting in a speed-up of one order of magnitude).

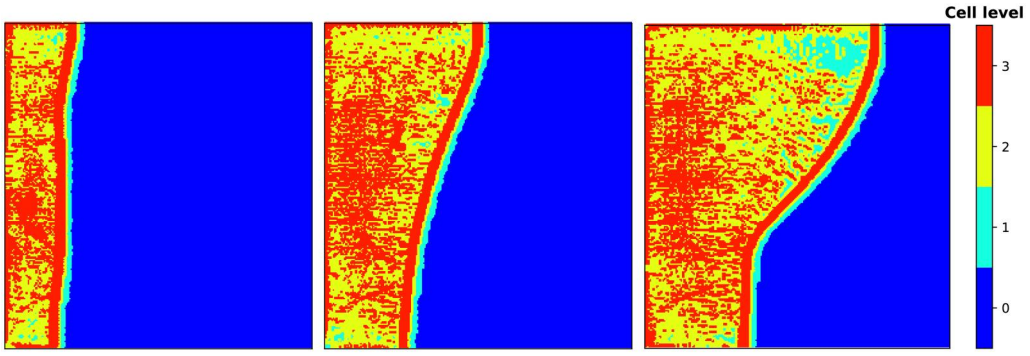
4.2. Melting of gallium in a 3D cavity

The melting of gallium in a 3D cuboid enclosure of $L \times H \times W = 88.9$ mm $\times$ 63.5 mm $\times$ 38.1 mm was selected for the second benchmark case. To reduce the computational cost, a



(*a*) 120 s (*b*) 360 s (*c*) 600 s

Figure 8. Normalized absolute velocity contour plots of gallium melting in a 3D cavity, at $z = 0$ (Central plane). Results were obtained using the adaptive mesh grid.



(*a*) 120 s (*b*) 360 s (*c*) 600 s

Figure 9. Cell level of the adaptive mesh for gallium melting in a 3D cavity, at $z = 0$ (Central plane).

half-sized computational domain of $L \times H \times W = 88.9$ mm $\times$ 63.5 mm $\times$ 19.05 mm was used, where a symmetry condition was applied at the center plane, at $z = 0$. The initial temperature is $T_0 = 301.3$ K and at $t = 0$ s, the left wall is suddenly heated to $T_L = 311.0$ K with the right wall being kept at $T_R = 301.3$ K. All other walls are considered to be adiabatic. The flow structure of the gallium melting in a 3D cavity case is known to differ significantly from that of the 2D case, due to the wall effect in the third dimension [14]. For this reason, the gallium melting in a 3D case is a suitable benchmark case for demonstrating the performance of our adaptive mesh method in three dimensions. The adaptive mesh consisted of a uniform background mesh of $70 \times 50 \times 10$ elements, with a maximum cell refinement level of 3 and an error threshold of $e_{r,rel}(|U|) < 0.01$.[2]. Compared to the 2D case, less strict refinement criteria were imposed due to the high computational cost associated with performing a solid-liquid phase change simulation in three dimensions. Again, adaptive time stepping was used based on a maximum Courant number of $Co = 3$, with the maximum time step size set to $\Delta t = 0.025$ s.

Figures 8 and 9 show the normalized absolute velocity and the corresponding cell levels after 120s, 360s and 600s. The flow features differed significantly between the 3D and the 2D case. Due to the suppression of the flow by the wall in the third dimension, the vortical structure in the melt is not as strong as in the 2D case. In addition, the multicellular flow patterns observed in the 2D simulations were absent in the 3D simulations, possibly due to the onset of a 3D instability destroying the formation of any 'long-lived' multi-vortex structures [67]. The flow in

Table 4. Total number of cells and distribution of the different cell levels (% cells/% volume) for the 3D gallium melting case.

Time	Total number of cells	Level 0	Level 1	Level 2	Level 3
120 s	1,574,517	1.9%/85.4%	0.2%/1.0%	8.1%/5.7%	89.9%/7.9%
180 s	1,904,714	1.5%/82.1%	0.2%/1.2%	8.4%/7.1%	89.9%/9.6%
360 s	2,586,220	1.0%/73.5%	0.2%/1.6%	10.5%/12.1%	88.3%/12.7%
480 s	2,989,777	0.8%/67.7%	0.2%/2.0%	11.8%/15.7%	87.2%/14.6%
600 s	3,225,292	0.7%/62.1%	0.3%/3.1%	13.5%/19.4%	85.6%/15.4%
750 s	3,554,208	0.5%/55.0%	0.3%/4.4%	15.1%/24.0%	84.0%/16.7%
900 s	3,831,464	0.4%/48.2%	0.5%/6.3%	16.3%/27.8%	82.8%/17.7%
1020s	4,086,880	0.4%/42.7%	0.5%/7.9%	16.8%/30.6%	82.3%/18.8%
1140s	4,187,673	0.3%/37.3%	0.7%/10.4%	17.8%/33.3%	81.2%/19.0%

the melt is refined quite uniformly, with the strongest refinement found near the solid-liquid interface and the left and top walls.

Table 4 shows the total number of cells at different times as well as the distribution of the different cell levels. Please note that for the gallium melting in a 3D cavity case, no uniform reference solution was generated due to the very high computational costs involved. Even with adaptive mesh refinement, the total number of elements used is still very high (though significantly less than when uniformly applying the maximum refinement level, which would result in a mesh size of $560 \times 400 \times 80 = 17,920,000$ elements). Whilst in the early stages most of the cells were refined up to the highest level (similar to the 2D Gallium case), after 480 s the most significant part of the domain was refined up to the second level. However, the third refinement level continued to dominate the overall cell count, at over 80%, due to the cells being 8 times as small compared to the second level. As mentioned before, isotropic mesh refinement was used in this work. Possibly, an anisotropic refinement strategy could yield similarly accurate results with even smaller mesh sizes, in particular for the 3D case, by refining cells only in the direction where needed. However, as of yet anisotropic mesh refinement has not been implemented in OpenFOAM.

Figure 10 shows the solid-liquid interface position at different times. The results from the present numerical campaign were compared with those of *Kumar et al* [14], and the experimental results of *Gau et al* [1]. Overall, a good agreement was observed between the results from the current numerical campaign and those of *Kumar et al*. Compared to the experimental measurements, both the current numerical campaign and the one of *Kumar et al* slightly over-predict the melting rate at the earlier times (until approximately $t = 600$ s). On the other hand, at later times the results from the present numerical campaign under-predict the melting rate. A possible cause for the discrepancies between the numerical results and the experimental measurements is the thermal inertia of the experiment during the startup phase, the use of the intrusive measurement techniques, and the overall lack of thermal insulation during the experiment [1]. However, the discrepancy could also be a consequence of modeling assumptions such as the use of the *Boussinesq* approximation or imposing adiabatic boundary conditions on the bottom, top, front, and back walls. Compared to the arbitrary Lagrangian-Eulerian formulation of *Kumar et al* [14], the results from the present numerical campaign predict an overall lower melting rate. This discrepancy may be caused by factors such as the mesh resolution, the time resolution, and a difference in solid-liquid phase change modeling approach. Please note that the present formulation guarantees the conservation of the thermal energy through the convergence criterium [see Eq. (8)].

4.3. The Molten Salt Reactor freeze plug

For the third and final test case, the Molten Salt Reactor (MSR) freeze valve was considered. The freeze valve is a key safety component of the MSR, which is designed to melt in case of an accident scenario.[3]. Recent work on the freeze valve includes a preliminary numerical investigation into both a single- and multi-plug configuration [6], a numerical investigation of a freeze valve design for a small modular molten salt reactor [10], a numerical benchmark study designed after

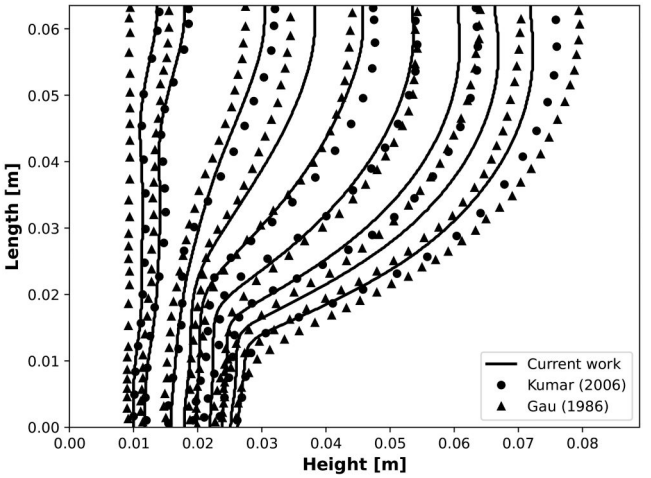
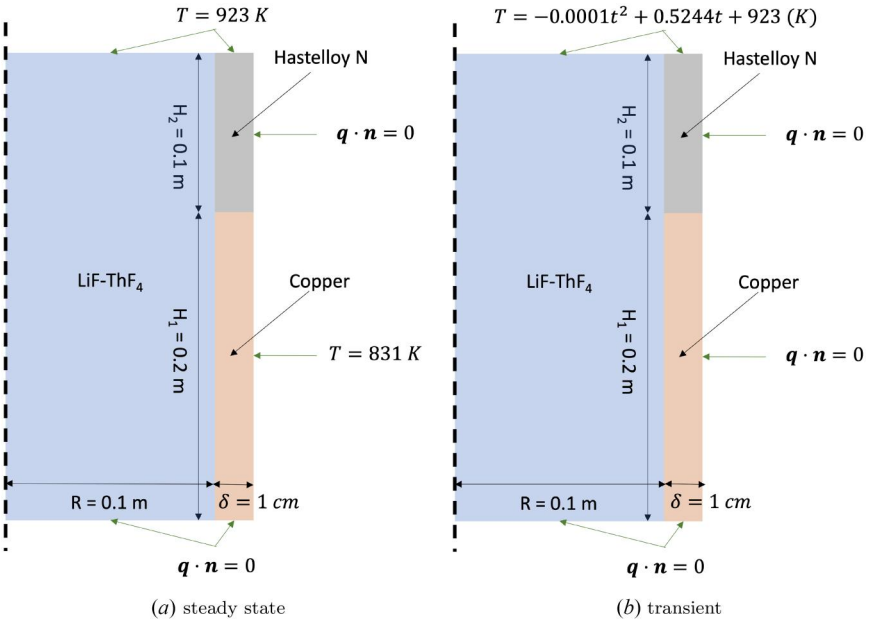


Figure 10. Solid-liquid interface position for the gallium melting in a 3D cavity, at $z = 0$ (Central plane). The results from the current numerical campaign are compared with the experimental results of *gau et al* [1], and the results from the numerical campaign of *kumar et al* [14].results are shown for 120 s, 180 s, 360 s, 480s, 600, 750, 900, 1020 and 1140 s.

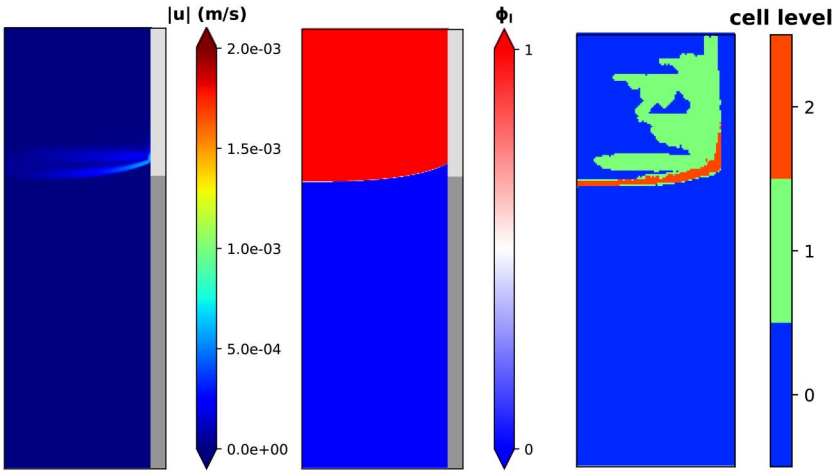


(*a*) steady state (*b*) transient

Figure 11. Geometry and boundary conditions of the MSR freeze-plug (single plug, decay heat design), for both the steady and the transient case.

the MSR freeze valve [11] and enhancing the heat transfer through the addition of fins [68] or by changing the inclination angle of the freeze valve [69]. For a recent and comprehensive overview of the different freeze valve designs and their development, please refer to *Chisholm et al* [9]. The MSR freeze valve was chosen as the final test case in order to demonstrate the applicability of our adaptive mesh refinement method to problems of industrial significance.

Figure 11 shows the geometry of the freeze-plug and its boundary conditions. For this test case, the single plug design was selected, with a recess depth of 0.1 m. This design was chosen due to its relative simplicity compared to other freeze-plug designs, making it a suitable demonstration case for the proposed adaptive mesh refinement strategy. The freeze valve consists of a



(a) $|u|$ (m/s) (b) ϕ_l (c) cell level

Figure 12. Numerical solution for the steady state freeze-plug case (which is the initial condition for the transient solution) for the absolute velocity (a), the liquid fraction (b) and the cell level (c), obtained with the adoptive mesh. The light and dark grey regions in the velocity and liquid fraction solutions refer to the hastelloy and the copper regions respectively, where these fields are not being calculated.

Table 5. Thermophysical properties of the salt, copper and hastelloy [6, 70, 71].

Property	$LiF\text{-}ThF_4$		Hastelloy-N	Copper
	Solid	Liquid		
Thermal conductivity [W m^{-1} K^{-1}]	1.5	1.5	23.6	401
Density [kg m^{-3}]	4502	4390	8860	8960
Specific heat capacity [J kg^{-1} K^{-1}]	815	1000	578	377
Latent heat [J kg^{-1}]	$1.59 \cdot 10^5$		–	–
Melting temperature [K]	841		–	–
Thermal expansion coefficient [K^{-1}]	$2.5 \cdot 10^{-4}$		–	–
Dynamic Viscosity [Pa s]	$7.5 \cdot 10^{-3}$		–	–

cylinder of salt with a radius of $R = 0.1$ m and a height of $H = 0.3$ m, surrounded by a metal wall with a thickness of $\delta = 1$ cm, of which the bottom 0.2 m consists of copper and the top 0.1 m consists of hastelloy. The thermophysical properties of the salt, the copper, and the hastelloy are given in Table 5. Due to the axisymmetrical nature of the problem, the 3D geometry is approximated by a 5° wedge with a thickness of one cell, and OpenFOAM's wedge boundary condition[4] is applied to the front and back face of the wedge. A segregated approach was adopted for the conjugate heat transfer modeling. Thus, separate matrix equations were solved for each region, and the temperature at the region interfaces was calculated through harmonic averaging:

$$T_{IF} = \frac{T_\alpha \times (\lambda_\alpha/\delta_\alpha) + T_\beta \times (\lambda_\beta/\delta_\beta)}{\lambda_\alpha/\delta_\alpha + \lambda_\beta/\delta_\beta}. \tag{20}$$

Here, α and β refer to the neighboring cells at each side of the interface, and δ refers to the distance from the cell-centroid to the interface.

The initial mesh consists of a uniform 110×300 quadrilateral mesh, and a maximum time step size of 0.25 s was used throughout the simulations, with a maximum Co number of $Co = 3$. For this case, conductive heat transfer through the metal walls played a significant role in determining the heat transfer rate. As such, a refinement criterion based on the solid-liquid interface and the numerical discretization error in the flow field alone was not deemed sufficient, as opposed to the previous two cases. Therefore, the estimated relative numerical discretization error

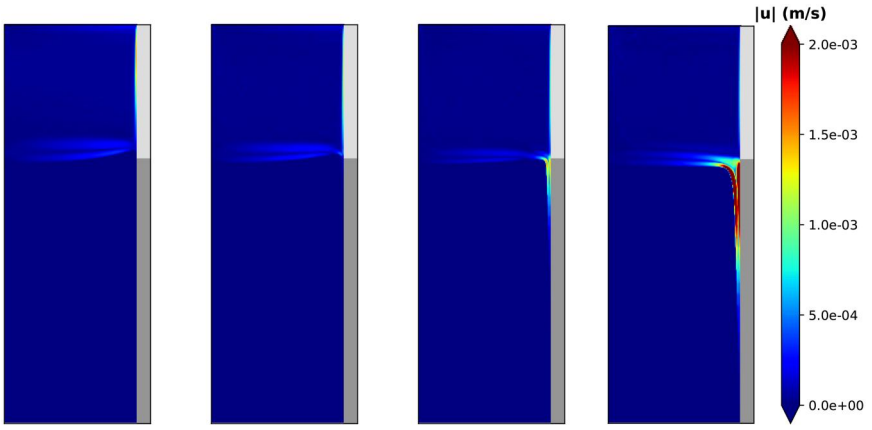
in the temperature field was included in the refinement criterion. The numerical solution was found to be more sensitive to small errors in the numerical discretization of the temperature field than of the flow field. A maximum refinement level was set of 2, an error threshold of $e_{r,rel}(|U|) < 0.01$ was set for the flow field and an error threshold of $e_{r,rel}(T) < 0.0001$ was set for the temperature field.

In order to simulate the melting of the freeze-plug, first a steady-state solution of the freeze-plug needs to be obtained. For the steady-state simulation, the top temperature was set to 923 K and the temperature of the right copper boundary was set to 831 K. Adiabatic boundary conditions were imposed at the bottom boundaries and at the right hastelloy wall. The simulation was run for 25000 s in order to reach the steady-state solution. Subsequently, the steady-state freeze-plug solution is used as the initial condition for the transient melting of the freeze-plug case.

Figure 12 shows the results for the steady-state freeze-plug, the absolute velocity, the liquid fraction, and the cell level. Compared to the freeze-plug shape obtained by *Tiberga et al* [6], the present simulations predict a smaller curvature of the solid-liquid interface. This is believed to be due to the effect of natural convection (which was not considered by *Tiberga et al*). Indeed, a recirculation zones was present near the solid-liquid interface. Near the cold copper wall, the salt flows down along the freeze front toward the center of the domain (bottom leg of the recirculation zone). Subsequently, the salt is recirculated back toward the hastelloy wall (top leg of the recirculation zone). Hereby, the mixing within the salt is enhanced, resulting in an overall flatter steady-state freeze-plug shape compared to a conduction-only simulation. The cells are most significantly refined close to the hastelloy wall and the solid-liquid interface, where the largest velocities are reached. The predicted numerical discretization error in the temperature field for the copper and hastelloy regions was very small, resulting in no refinement within these regions.

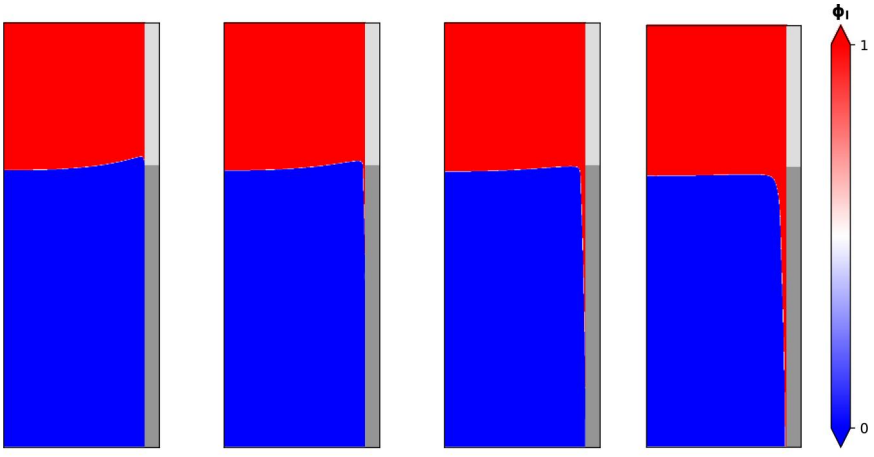
To start the transient simulation, the top boundary condition was changed from $T = 923$ K to $T(t) = -0.0001t^2 + 0.5244t + 923$ K, representing the average temperature increase as a result of the decay heat following a reactor shutdown [6]. In addition, a zero-gradient condition was imposed at the right copper boundary, since the cooling system has now been switched off. Figures 13, 14, and 15 show the results for the transient case for the absolute velocity, the liquid fraction, and the cell level, evaluated after 500 s, 1000 s, 1500 s, and 2000 s. Since the hastelloy has a higher thermal conductivity than the salt, the heat from the top penetrates further into the hastelloy wall compared to the salt, resulting in higher temperatures in the vicinity of the hastelloy wall compared to the center of the domain. As a result, the salt starts to flow up along the hastelloy wall, enhancing the mixing in the liquid salt layer above the freeze-plug. When the freeze-plug starts to melt, an additional recirculation zone appears between the solid-liquid interface and the copper wall, increasing the melting rate at the top of the freeze-plug (but decreasing the melting rate at the sides, thus prolonging the opening time as shown by *Kaaks, Pater et al* [11] and *Aji* [69]). The cells are most significantly refined close to the hastelloy wall, the solid-liquid interface and at the top of the domain, where most of the recirculation takes place. Few cells are refined in the copper and the hastelloy walls, indicating that the largest numerical discretization errors are present in the salt. As the salt melts, a gap starts to appear between the freeze-plug and the copper wall in which recirculation of the salt takes place, effectively shifting the heat transfer from the copper wall to the top of the freeze plug. Close to the copper wall, a thin layer of cells in the frozen layer of salt (located below the solid-liquid interface position) was also refined, attributed to the importance of the heat transfer through the copper wall during the melting of the freeze valve.

Figure 16 shows the relative total solid fraction of the freeze-plug at the metal wall for the adaptive mesh and three uniform meshes consisting of 440×1200, 220×600, and 110×300 elements. The total solid fraction of the freeze-plug at the metal wall (at $x = 0.1$ m) was calculated for each mesh at a time interval of 25 s by approximating the integral of $\phi_s|_{x=0.1\ \mathrm{m}} = 1 - \phi_l|_{x=0.1\ \mathrm{m}}$ over the height (y) using the trapezium rule. The relative total solid fraction was



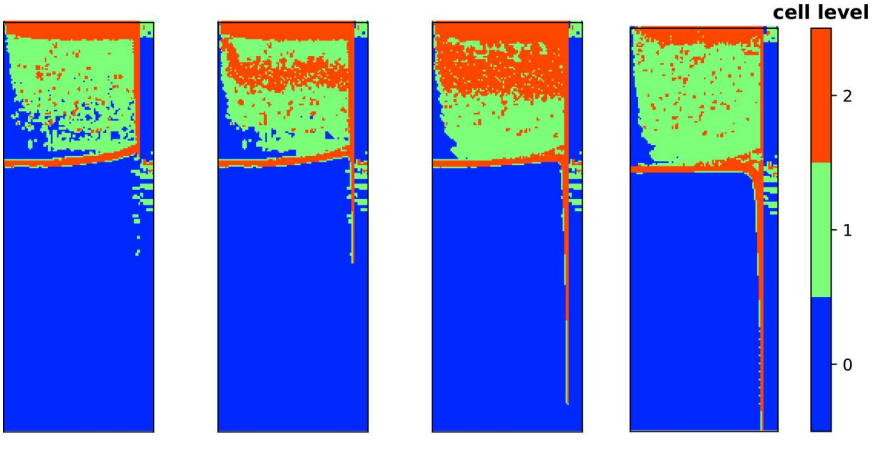
(*a*) 500 s (*b*) 1000 s (*c*) 1500 s (*d*) 2000 s

Figure 13. Numerical solution for the transient freeze-plug case for the absolute velocity, obtained with an adaptive mesh. The light and dark grey regions refer to the hastelloy and the copper respectively.



(*a*) 500 s (*b*) 1000 s (*c*) 1500 s (*d*) 2000 s

Figure 14. Numerical solution for the transient freeze-plug case for the liquid fraction, obtained with an adaptive mesh. The light and dark grey regions refer to the hastelloy and the copper respectively.



(*a*) 500 s (*b*) 1000 s (*c*) 1500 s (*d*) 2000 s

Figure 15. Cell level of the adaptive mesh for the MSR freeze-valve.

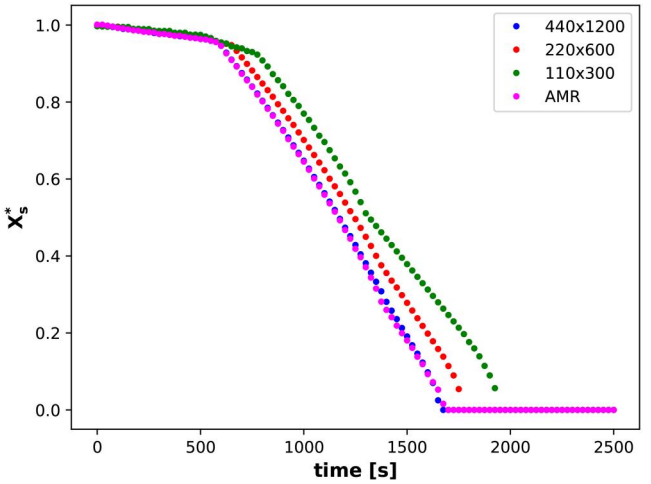


Figure 16. Relative solid fraction of the freeze-plug at the metal wall ($R = 0.1$ m), showing the melt trajectory of the plug. Results from the adaptive mesh were compared to the solutions of three progressively coarser uniform meshes.

Table 6. Total number of cells of the adaptive mesh at different simulation times for the MSR freeze valve melting case.

Time	Number of cells (AMR)
0 s	47,583
500 s	73,260
1000s	83,427
1500s	98,280
2000s	84,030

obtained by normalizing the total solid fractions with the reference solid fraction, evaluated for the 440×1200 mesh at $t = 0$:

$$X_s^* = \frac{\int_0^{y=0.3\ \mathrm{m}} \left(1 - \phi_l(t)|_{x=0.1m}\right) dy}{\int_0^{y=0.3\ \mathrm{m}} \left(1 - \phi_l^{ref}(t = 0\ \mathrm{s})|_{x=0.1\ \mathrm{m}}\right) dy} \tag{21}$$

The relative total solid fraction at the copper wall is the quantity of interest for the melting of the MSR freeze valve, since the freeze valve is expected to fall into the drainage tank below once it has become detached from the solid wall. As such, $X_s^* = 0$ corresponds to the opening time of the freeze valve. The opening time is much longer than the 715 s predicted by *Tiberga et al* [6]. Therefore, including the role of natural convection appears to result in a prolonged opening time of the freeze plug.

A very good agreement was obtained between the adaptive mesh and the reference mesh with 440×1200 elements. On the other hand, the coarser meshes over-predicted the opening time, with a difference of more than 200 s in the predicted opening time between the coarsest mesh of 110×300 elements and the finest mesh of 440×1200 elements. At all times, the number of elements of the adaptive mesh was smaller than the 220×600 uniform mesh (see Table 6). As such, it is evident that the adaptive mesh yields similarly accurate results compared to the uniform mesh with 440×1200 elements, at a much lower computational cost.

5. Conclusions

This work present a finite volume adaptive mesh refinement method for solid-liquid phase change problems. The refinement criteriion consisted of a combination of three different sub-criteria, based on the variation in liquid fraction (to refine the solid-liquid interface) and the numerical discretization errors of the velocity and temperature fields. For the solid-liquid interface, the cells undergoing the phase transition

were flagged based on the maximum face jump in the liquid fraction and refined up to a user-specified refinement level. For the flow field and the temperature field, the cell residual method of *Jasak et al* [60] was used to estimate the numerical discretization error, and the cells which exceeded the user-specified error tolerance were refined until the maximum refinement level was reached. To maintain a high parallelization efficiency during the adaptive mesh refinement, a dynamic load balancing procedure was used.

The adaptive mesh refinement strategy was verified for three different test cases, these are the gallium melting in a 2D rectangular cavity case, the gallium melting in a 3D cuboid cavity case and the MSR freeze-plug. For the gallium melting in a 2D cavity case, a very good agreement was obtained between the adaptive mesh and the uniform reference mesh, with L^2 relative differences of 5% or lower for all evaluated times. The uniform reference mesh was more than 10 times larger than the adaptive mesh, demonstrating the potential of the proposed adaptive mesh refinement approach to locally achieve a high resolution (and therefore an accurate solution) with limited computational cost. Indeed, a speed-up of an order of magnitude was achieved for the 2D gallium melting case when using the presented adaptive mesh refinement approach. For the gallium melting in a 3D case, we demonstrated very good agreement between the solution with the adaptive mesh and the numerical and experimental reference solutions. Whilst the results with the AMR mesh were shown to be accurate, the overall mesh size was still very large, reaching over 4,000,000 cells toward the end of the simulation. Possibly, similar accuracy could be achieved with even smaller meshes using an anisotropic mesh refinement approach, where cells are only defined in the direction where needed (instead of being refined isotropically using a quadtree or octtree hierarchical refinement tree approach). The results for the MSR freeze valve demonstrated the applicability of our adaptive mesh refinement strategy to more complex solid-liquid phase change problems which also involved conjugate heat transfer, with a very good agreement obtained between the predicted opening time with the uniform reference mesh and the more than five times smaller adaptive mesh.

A possible area of future research could be the use of an anisotropic adaptive mesh refinement strategy, which would be the next step in the development of computationally efficient finite volume methods for solving solid-liquid phase change problems. Whilst the refinement criterium presented in this work appears to be promising, it is not applicable to anisotropic AMR (since the refinement criterium is a scalar instead of a tensor) and some trial and error was still needed for selecting the optimal value for the tolerance, which proved to be problem dependent. As such, further research could be additionally dedicated toward the development of universal and robust error estimators for finite volume methods with (anisotropic) adaptive mesh refinement, which can be applied to solid-liquid phase change problems.

Notes

1. In our case, we are dealing with isothermal phase change and the value of the mushy zone parameters therefore have limited effect on the results as long as 'C' is large enough to ensure sufficient attenuation of the velocity at the solid-liquid interface [31]. However, a too large value of 'C' may result in numerical instabilities, similar to the velocity cutoff technique. In our experience, a value of $C = 10^6$ was the best compromise between achieving sufficient attenuation and avoiding spurious oscillations at the solid-liquid interface.
2. Again, no refinement criterion was imposed for the numerical discretization error in the temperature field.
3. To prevent structural damage to the reactor, the freeze-plug needs to melt within 1000s [6].
4. The wedge boundary condition features an axisymmetrical boundary condition using a rotational transformation to map between a cartesian and cylindrical/spherical coordinate system.

Disclosure statement

No potential conflict of interest was reported by the author(s)

Funding

This project has received funding from the Euratom research and training programme 2014-2018 under grant agreement no. 847527.

ORCID

Bouke Johannes Kaaks http://orcid.org/0000-0003-0036-0965

Data availability statement

The numerical data generated for the melting of gallium in the 2D and the 3D cavities and the melting of the freeze valve can be found under the associated link: https://doi.org/10.5281/zenodo.10058162. In addition, the developed OpenFOAM solvers for solid-liquid phase change with adaptive mesh refinement have been made publicly available: see: https://doi.org/10.5281/zenodo.10074609.

References

[1] C. Gau and R. Viskanta, "Melting and solidification of a pure metal on a vertical wall," *J. Heat Transf.*, vol. 108, no. 1, pp. 174–181, 1986. DOI: 10.1115/1.3246884.

[2] M. Faden, C. Linhardt, S. Höhlein, A. König-Haagen, and D. Brüüggemann, "Velocity field and phase boundary measurements during melting of n-octadecane in a cubical test cell," *Int. J. Heat Mass Transf.*, vol. 135, pp. 104–114, 2019. 10.1016/j.ijheatmasstransfer.2019.01.056. DOI: 10.1016/j.ijheatmasstransfer.2019.01.056.

[3] A. Benkaddour, M. Faraji, and H. Faraji, "Numerical study of the thermal energy storage behaviour of a novel composite PCM/Concrete wall integrated solar collector," *Mater. Today Proc.*, vol. 30, pp. 905–908, 2020. DOI: 10.1016/j.matpr.2020.04.348.

[4] H. Faraji, M. E. Alami, A. Arshad, and Y. Hariti, "Numerical survey on performance of hybrid NePCM for cooling of electronics: effect of heat source position and heat sink inclination," *J. Therm. Sci. Eng. Appl.*, vol. 13, no. 5, p. 051010, 2021. DOI: 10.1115/1.4049431.

[5] H. Faraji, C. , Yıldız, A. Arshad, M. Arıcı, K. Choukairy, and M. El Alami, "Passive thermal management strategy for cooling multiple portable electronic components: hybrid nanoparticles enhanced phase change materials as an innovative solution," *J. Energy Storage*, vol. 70, pp. 108087, Oct. 2023. https://linkinghub.elsevier.com/retrieve/pii/S2352152X23014846. DOI: 10.1016/j.est.2023.108087.

[6] M. Tiberga, D. Shafer, D. Lathouwers, M. Rohde, and J. L. Kloosterman, "Preliminary investigation on the melting behavior of a freeze-valve for the molten salt fast reactor," *Ann. Nucl. Energy*, vol. 132, pp. 544–554, 2019. DOI: 10.1016/j.anucene.2019.06.039.

[7] G. Cartland Glover, *et al.*, "On the numerical modelling of frozen walls in a molten salt fast reactor," *Nucl. Eng. Des.*, vol. 355, p. 110290, Mar. 2019. DOI: 10.1016/j.nucengdes.2019.110290.

[8] V. Voulgaropoulos, N. L. Brun, A. Charogiannis, and C. N. Markides, "Transient freezing of water between two parallel plates: a combined experimental and modelling study," *Int. J. Heat Mass Transf.*, vol. 153, p. 119596, 2020. DOI: 10.1016/j.ijheatmasstransfer.2020.119596.

[9] B. M. Chisholm, S. L. Krahn, and A. G. Sowder, "A unique molten salt reactor feature – The freeze valve system: design, operating experience, and reliability," *Nucl. Eng. Des.*, vol. 368, pp. 110803, Aug. 2020. 10.1016/j.nucengdes.2020.110803. DOI: 10.1016/j.nucengdes.2020.110803.

[10] M. C. Kahraman, and S. S, entürk Lüle, "Numerical investigation of behaviour of small modular molten salt reactor freeze plug under various reactor operating conditions," *Numer. Heat Transf. Part A Appl*, vol. 85, no. 11, pp. 1747–1766, 2023.10.1080/10407782.2023.2210258. DOI: 10.1080/10407782.2023.2210258.

[11] M. Pater, B. Kaaks, B. Lauritzen, and D. Lathouwers, "A numerical benchmark for modelling phase change in molten salt reactors," *Ann. Nuclear Energy*, vol. 194, p. 110093, Aug. 2023. DOI: 10.1016/j.anucene.2023.110093.

[12] K. R. Sultana, S. R. Dehghani, K. Pope, and Y. S. Muzychka, "Numerical techniques for solving solidification and melting phase change problems," *Numer. Heat Transf. Part B Fundam.*, vol. 73, no. 3, pp. 129–145, 2018. 10.1080/10407790.2017.1422629. DOI: 10.1080/10407790.2017.1422629.

[13] A. König-Haagen, E. Franquet, E. Pernot, and D. Brüggemann, "A comprehensive benchmark of fixed-grid methods for the modeling of melting," *Int. J. Thermal Sci.*, vol. 118, pp. 69–103, 2017. DOI: 10.1016/j.ijthermalsci.2017.04.008.

[14] V. Kumar, F. Durst, and S. Ray, "Modeling moving-boundary problems of solidification and melting adopting an arbitrary lagrangian-eulerian approach," *Numer. Heat Transf. Part B Fundam.*, vol. 49, no. 4, pp. 299–331, 2006. DOI: 10.1080/10407790500379981.

[15] M. Lacroix, and V. R. Voller, "Finite difference solutions of solidification phase change problems: transformed versus fixed grids," *Numer. Heat Transf. Part B Fundam.*, vol. 17, no. 1, pp. 25–41, 1990. DOI: 10.1080/10407799008961731.

[16] R. Viswanath, and Y. Jaluria, "A comparison of different solution methodologies for melting and solidification problems in enclosures," *Numer. Heat Transf. Part B Fundam.*, vol. 24, no. 1, pp. 77–105, 1993. DOI: 10.1080/10407799308955883.

[17] M. Błasik, and M. Klimek, "Numerical solution of the one phase 1D fractional Stefan problem using the front fixing method," *Math. Methods Appl. Sci.*, vol. 38, no. 15, pp. 3214–3228, 2015. DOI: 10.1002/mma.3292.

[18] H. Udaykumar, R. Mittal, and W. Shyy, "Elafint: a mixed Euler-Lagrangian method for fluid flows with complex and moving boundaries," *Int. J. Numer. Methods Fluids*, vol. 22, no. 8, pp. 691–712, Jun. 1996. DOI: 10.1002/(SICI)1097-0363(19960430)22:8<691::AID-FLD371>3.0.CO;2-U.

[19] H. S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna, "A sharp interface Cartesian grid method for simulating flows with complex moving boundaries," *J. Comput. Phys.*, vol. 174, no. 1, pp. 345–380, 2001. DOI: 10.1006/jcph.2001.6916.

[20] D. Juric, and G. Tryggvason, "A front-tracking method for dendritic solidification," *J. Comput. Phys.*, vol. 123, no. 1, pp. 127–148, 1996. DOI: 10.1006/jcph.1996.0011.

[21] W. J. Boettinger, J. A. Warren, C. Beckermann, and A. Karma, "Phase-field simulation of solidification," *Annu. Rev. Mater. Res.*, vol. 32, no. 1, pp. 163–194, 2002. DOI: 10.1146/annurev.matsci.32.101901.155803.

[22] M. Greenwood, *et al.*, "Quantitative 3D phase field modelling of solidification using next-generation adaptive mesh refinement," *Comput. Mater. Sci.*, vol. 142, pp. 153–171, 2018. DOI: 10.1016/j.commatsci.2017.09.029.

[23] S. Chen, B. Merriman, S. Osher, and P. Smereka, "A simple level set method for solving stefan problems," *J. Comput. Phys.*, vol. 135, no. 1, pp. 8–29, 1997. DOI: 10.1006/jcph.1997.5721.

[24] H. Zhang, L. L. Zheng, V. Prasad, and T. Y. Hou, "A curvilinear level set formulation for highly deformable free surface problems with application to solidification," *Numer. Heat Transf. Part B: Fundam.*, vol. 34, no. 1, pp. 1–30, 1998. DOI: 10.1080/10407799808915045.

[25] L. Tan and N. Zabaras, "A level set simulation of dendritic solidification with combined features of front-tracking and fixed-domain methods," *J. Comput. Phys.*, vol. 211, no. 1, pp. 36–63, 2006. DOI: 10.1016/j.jcp.2005.05.013.

[26] C. Bonacina, G. Comini, A. Fasano, and M. Primicerio, "Numerical solution of phase-change problems," *Int. J. Heat Mass Transf.*, vol. 16, no. 10, pp. 1825–1832, 1973. DOI: 10.1016/0017-9310(73)90202-0.

[27] K. Morgan, "A numerical analysis of freezing and melting with convection," *Comput. Methods Appl. Mech. Eng.*, vol. 28, no. 3, pp. 275–284, 1981. DOI: 10.1016/0045-7825(81)90002-5.

[28] Q. T. Pham, "A fast, unconditionally stable finite-difference scheme for heat conduction with phase change," *Int. J. Heat Mass Transf.*, vol. 28, no. 11, pp. 2079–2084, 1985. DOI: 10.1016/0017-9310(85)90101-2.

[29] V. Voller and M. Cross, "Accurate solutions of moving boundary problems using the enthalpy method," *Int. J. Heat Mass Transf.*, vol. 24, no. 3, pp. 545–556, 1981. DOI: 10.1016/0017-9310(81)90062-4.

[30] V. R. Voller and C. Prakash, "A fixed grid numerical modelling methodology for convection diffusion mushy region phase change problems," *Int. J. Heat Mass Transf.*, vol. 30, no. 8, pp. 1709–1719, 1987. DOI: 10.1016/0017-9310(87)90317-6.

[31] A. D. Brent, V. R. Voller, and K. J. Reid, "Enthalpy-porosity technique for modeling convection-diffusion phase change: application to the melting of a pure metal," *Numer. Heat Transf.*, vol. 13, no. 3, pp. 297–318, 1988. DOI: 10.1080/10407788808913615.

[32] C. R. Swaminathan and V. R. Voller, "On the enthalpy method," *Int. J. Numer. Methods Heat Fluid Flow*, vol. 3, no. 3, pp. 233–244, 1993. DOI: 10.1108/eb017528.

[33] B. Nedjar, "An enthalpy-based finite element method for nonlinear heat problems involving phase change," *Comput. Struct.*, vol. 80, no. 1, pp. 9–21, 2002. DOI: 10.1016/S0045-7949(01)00165-1.

[34] M. Faden, A. König-Haagen, and D. Brüggemann, "An optimum enthalpy approach for melting and solidification with volume change," *Energies*, vol. 12, no. 5, pp. 868, 2019. DOI: 10.3390/en12050868.

[35] B. J. Kaaks, M. Rohde, J. L. Kloosterman, and D. Lathouwers, "An energyconservative DG-FEM approach for solid–liquid phase change," *Numer. Heat Transf. Part B Fundam.*, vol. 84, no. 5, pp. 487–513, 2023. DOI: 10.1080/10407790.2023.2211231.10.1080/10407790.2023.2211231.

[36] M. Tano, P. Rubiolo, and O. Doche, "Progress in modeling solidification in molten salt coolants," *Model. Simul. Mater. Sci. Eng.*, vol. 25, no. 7, p. 074001, 2017. DOI: 10.1088/1361-651X/aa8345.

[37] N. Hannoun, V. Alexiades, and T. Z. Mai, "Resolving the controversy over tin and gallium melting in a rectangular cavity heated from the side," *Numer. Heat Transf. Part B Fundam.*, vol. 44, no. 3, pp. 253–276, 2003. DOI: 10.1080/713836378.

[38] J. S. Cagnone, K. Hillewaert, and N. Poletz, "A discontinuous Galerkin method for multiphysics welding simulations," *KEM*, vol. 611–612, pp. 1319–1326, 2014. DOI: 10.4028/www.scientific.net/KEM.611-612.1319.
[39] P. W. Schroeder and G. Lube, "Stabilised dG-FEM for incompressible natural convection flows with boundary and moving interior layers on non-adapted meshes," *J. Comput. Phys.*, vol. 335, pp. 760–779, 2017. http://dx.doi.org/10.1016/j.jcp.2017.01.055. DOI: 10.1016/j.jcp.2017.01.055.
[40] J. Chessa, P. Smolinski, and T. Belytschko, "The extended finite element method (XFEM) for solidification problems," *Int. J. Numer. Methods Eng.*, vol. 53, no. 8, pp. 1959–1977, 2002. DOI: 10.1002/nme.386.
[41] S. Soghrati, A. M. Aragon, C. A. Duarte, and P. H. Geubelle, "An interfaceenriched generalized FEM for problems with discontinuous gradient fields," *Numer. Methods Eng.*, vol. 89, no. 8, pp. 991–1008, 2012. DOI: 10.1002/nme.3273.
[42] J. Zhang, S. J. van den Boom, F. van Keulen, and A. M. Aragón, "A stable discontinuity-enriched finite element method for 3-D problems containing weak and strong discontinuities," *Comput. Methods Appl. Mech. Eng.*, vol. 355, pp. 1097–1123, 2019. 10.1016/j.cma.2019.05.018. DOI: 10.1016/j.cma.2019.05.018.
[43] I. Danaila, R. Moglan, F. Hecht, and S. Le Masson, "A Newton method with adaptive finite elements for solving phase-change problems with natural convection," *J. Comput. Phys.*, vol. 274, pp. 826–840, 2014. http://dx.doi.org/10.1016/j.jcp.2014.06.036. DOI: 10.1016/j.jcp.2014.06.036.
[44] Y. Belhamadia, A. Fortin, and T. Briffard, "A two-dimensional adaptive remeshing method for solving melting and solidification problems with convection," *Numer. Heat Transf. Part A Appl.*, vol. 76, no. 4, pp. 179–197, 2019. 10.1080/10407782.2019.1627837. DOI: 10.1080/10407782.2019.1627837.
[45] O. Antepara, O. Lehmkuhl, R. Borrell, J. Chiva, and A. Oliva, "Parallel adaptive mesh refinement for large-eddy simulations of turbulent flows," *Comput. Fluids*, vol. 110, pp. 48–61, 2015. http://dx.doi.org/10.1016/j.compfluid.2014.09.050. DOI: 10.1016/j.compfluid.2014.09.050.
[46] L. Li, D. Hu, Y. Liu, B. Wang, C. Shi, J. Shi, and C. Xu, "Large eddy simulation of cavitating flows with dynamic adaptive mesh refinement using OpenFOAM," *J. Hydrodyn.*, vol. 32, no. 2, pp. 398–409, 2020. DOI: 10.1007/s42241-019-0041-1.
[47] S. J. Kamkar, A. M. Wissink, V. Sankaran, and A. Jameson, "Feature-driven Cartesian adaptive mesh refinement for vortex-dominated flows," *J. Comput. Phys.*, vol. 230, no. 16, pp. 6271–6298, 2011. http://dx.doi.org/10.1016/j.jcp.2011.04.024. DOI: 10.1016/j.jcp.2011.04.024.
[48] J. P. P. Magalhães, D. M. S. Albuquerque, J. M. C. Pereira, and J. C. F. Pereira, "Adaptive mesh finite-volume calculation of 2D lid-cavity corner vortices," *J. Comput. Phys.*, vol. 243, pp. 365–381, 2013. DOI: 10.1016/j.jcp.2013.02.042.
[49] D. Deising, D. Bothe, and H. Marschall, "Direct numerical simulation of mass transfer in bubbly flows," *Comput. Fluids*, vol. 172, pp. 524–537, 2018. DOI: 10.1016/j.compfluid.2018.03.041.
[50] O. Antepara, N. Balc'azar, J. Rigola, and A. Oliva, "Numerical study of rising bubbles with path instability using conservative level-set and adaptive mesh refinement," *Comput. Fluids*, vol. 187, pp. 83–97, 2019. DOI: 10.1016/j.compfluid.2019.04.013.
[51] J. Wackers, G. Deng, A. Leroyer, P. Queutey, and M. Visonneau, "Adaptive grid refinement for hydrodynamic flows," *Comput. Fluids*, vol. 55, pp. 85–100, 2012. http://dx.doi.org/10.1016/j.compfluid.2011.11.004. DOI: 10.1016/j.compfluid.2011.11.004.
[52] J. Wackers, *et al.*, "Can adaptive grid refinement produce gridindependent solutions for incompressible flows?," *J. Comput. Phys.*, vol. 344, pp. 364–380, 2017. http://dx.doi.org/10.1016/j.jcp.2017.04.077. DOI: 10.1016/j.jcp.2017.04.077.
[53] N. Palle, and J. A. Dantzig, "An adaptive mesh refinement scheme for solidification problems," *Metall. Mater. Trans. A.*, vol. 27, no. 3, pp. 707–717, 1996. DOI: 10.1007/BF02648957.
[54] C. Zienkiewicz, and J. Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis," *Int. J. Numer. Methods Eng.*, vol. 24, no. 2, pp. 337–357, 1987. DOI: 10.1002/nme.1620240206.
[55] C. W. Lan, C. C. Liu, and C. M. Hsu, "An adaptive finite volume method for incompressible heat flow problems in solidification," *J. Comput. Phys.*, vol. 178, no. 2, pp. 464–497, 2002. DOI: 10.1006/jcph.2002.7037.
[56] J. Mencinger, "Numerical simulation of melting in two-dimensional cavity using adaptive grid," *J. Comput. Phys.*, vol. 198, no. 1, pp. 243–264, Jul. 2004. DOI: 10.1016/j.jcp.2004.01.006.
[57] R. T. Tenchev, J. A. Mackenzie, T. J. Scanlon, and M. T. Stickland, "Finite element moving mesh analysis of phase change problems with natural convection," *Int. J. Heat Fluid Flow*, vol. 26, no. 4, pp. 597–612, 2005. DOI: 10.1016/j.ijheatfluidflow.2005.03.003.
[58] P. Nithiarasu, "Adaptive finite element procedure for solidification problems," *Warme- Und Stoffubertragung Zeitschrift*, vol. 36, no. 3, pp. 223–229, 2000. DOI: 10.1007/s002310050389.
[59] Y. Belhamadia, A. Fortin, and E. Chamberland, "Three-dimensional anisotropic mesh adaptation for phase change problems," *J. Comput. Phys.*, vol. 201, no. 2, pp. 753–770, 2004. DOI: 10.1016/j.jcp.2004.06.022.
[60] H. Jasak and A. D. Gosman, "Residual error estimate for the finite volume method," *Numer. Heat Transf. Part B: Fundam*, vol. 39, no. 0, pp. 1–19, 2001.

[61] H. Jasak and A. D. Gosman, "Element residual error estimate for the finite volume method," *Comput. Fluids*, vol. 32, no. 2, pp. 223–248, 2003. DOI: 10.1016/S0045-7930(02)00004-X.

[62] D. Rettenmaier, *et al.*, "Load balanced 2D and 3D adaptive mesh refinement in OpenFOAM," *SoftwareX*, vol. 10, pp. 100317, 2019. 10.1016/j.softx.2019.100317. DOI: 10.1016/j.softx.2019.100317.

[63] P. A. Galione, O. Lehmkuhl, J. Rigola, and A. Oliva, "Fixed-grid numerical modeling of melting and solidification using variable thermo-physical properties - Application to the melting of n-Octadecane inside a spherical capsule," *Int. J. Heat Mass. Transf.*, vol. 86, pp. 721–743, 2015. http://dx.doi.org/10.1016/j.ijheatmasstransfer.2015.03.033. DOI: 10.1016/j.ijheatmasstransfer.2015.03.033.

[64] A. König-Haagen, E. Franquet, M. Faden, and D. Brüggemann, "Influence of the convective energy formulation for melting problems with enthalpy methods," *Int. J. Thermal Sci.*, vol. 158, pp. 106477, Jul. 2020. DOI: 10.1016/j.ijthermalsci.2020.106477.

[65] "blastFoam: a solver for compressible multi-fluid flow with application to high-explosive detonation." Available: https://github.com/synthetiktechnologies/blastfoam. Accessed Apr. 13, 2020.

[66] H. Jasak, "Error analysis and estimation for the finite volume method with applications to fluid flows," PhD thesis, Imperial College, 1996.

[67] K. Wittig and P. A. Nikrityuk, "Three-dimensionality of fluid flow in the benchmark experiment for a pure metal melting on a vertical wall," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 27, no. 1, p. 012054, 2011. DOI: 10.1088/1757-899X/27/1/012054.

[68] X. Y. Jiang, H. J. Lu, Y. S. Chen, Y. Fu, and N. X. Wang, "Numerical and experimental investigation of a new conceptual fluoride salt freeze valve for thoriumbased molten salt reactor," *Nucl. Sci. Tech.*, vol. 31, no. 2, p. 16, Feb. 2020. DOI: 10.1007/s41365-020-0729-5.

[69] I. K. Aji, "Investigation of basic parameters in developing highperformance freeze valvefor molten salt reactor," PhD thesis, University of Electro-Communications, 2020.

[70] O. Beneš and R. J. Konings, "Thermodynamic properties and phase diagrams of fluoride salts for nuclear applications," *J. Fluor. Chem.*, vol. 130, no. 1, pp. 22–29, 2009. DOI: 10.1016/j.jfluchem.2008.07.014.

[71] E. Capelli, O. Beneš, M. Beilmann, and R. J. Konings, "Thermodynamic investigation of the LiF-ThF4 system," *J. Chem. Thermodyn.*, vol. 58, pp. 110–116, 2013. DOI: 10.1016/j.jct.2012.10.013.