DELFT UNIVERSITY OF TECHNOLOGY

REPORT 05-07

A level set method for particle dissolution in a binary alloy

E. Javierre[1], C. Vuik, F.J. Vermolen, A. Segal

ISSN 1389-6520

Reports of the Delft Institute of Applied Mathematics

Delft 2005

[1]Supported by the Dutch Technology Foundation (STW).

A level set method for particle dissolution in a binary alloy

E. Javierre[†], C. Vuik, F.J. Vermolen and A. Segal

Delft Institute of Applied Mathematics, *Scientific Computing*,
Mekelweg 4, 2628 CD Delft, the Netherlands

Abstract

A mathematical model is proposed for the dissolution of stoichiometric particles in binary alloys occurring during the heat treatments of as-cast aluminium alloys prior to hot extrusion. The Level Set method is used to capture the interface location implicitly. The front velocity, only defined on the moving interface, should be advected to the whole computational domain. This advection is done in Cartesian or in normal directions. The numerical solution is carried out by a combination of finite difference and finite element methods. The cut-cell method is employed to adapt the finite element mesh to the interface position. Two- and three-dimensional results are presented. The numerical solution is compared with analytical solutions. Conservation of mass is also evaluated.

AMS classification: 35R35; 65M06; 80A22

Keywords: Stefan problem; Phase transformations; Similarity solutions; Level set method; Cut-cell

1 Introduction

Heat treatment of metals is often used to optimize mechanical properties. During heat treatment, the metallurgical state of the alloy changes. This change can involve the phase present at a given location or the morphology of the various phases. Whereas equilibrium phases can be predicted quite accurately from thermodynamic models, there are no general models for microstructural changes nor for the kinetics of these changes. In the latter cases, both the initial morphology and the transformation mechanisms have to be prescribed explicitly. One of these processes, which is both of large industrial and scientific interest and amenable to modeling, is the dissolution of second-phase particles in a matrix with a uniform initial composition.

To describe this particle dissolution in solid media, several physical models have been developed. Long-distance diffusion was incorporated in [1, 2, 3], in which the concentration of the solute at the interface between the adjacent phases is the solid solubility. The long-distance diffusion models imply that the processes at the interface proceed infinitely fast. Nonequilibrium conditions at the interface were incorporated in [4, 5, 6, 7].

[†]corresponding author: e.javierre@ewi.tudelft.nl

Numerical solution methods for moving boundary problems can be classified in three categories: front-tracking methods [8, 9, 10, 11], front-capturing methods [12]-[18] and hybrid methods [19]. Front-tracking methods use an explicit representation of the interface, given by a set of points lying on the interface location, which must be updated at each time step. Front-capturing methods choose an implicit representation of the interface, adding an artificial unknown to the problem. Among other front-capturing methods, the enthalpy method [12, 13] and the phase field method [14, 15] have been extensively used in phase transformation problems, whereas the level set method [16] has been exploited for wide range of applications, as is described in [17, 18]. In the enthalpy method, the enthalpy function is used to measure the total heat of the system. It has a jump across the interface due to the absorption/release of heat required for phase change. In the phase field method, the domain is parameterized by the phase field function. This function equals a fixed constant in each phase and varies rapidly, but smoothly, between these values in the interface region. Therefore an adaptive mesh procedure, with a high resolution inside the interfacial region, is required to solve the governing equations accurately. Finally, in the level set method the interface is captured as the zero level set of a continuous function. The evolution of the interface is reformulated as an advection equation for the level set function. Hybrid methods consider a combination of both front-tracking and front-capturing methods. Crank [20] provides a good introduction to free boundary problems and presents an elaborate collection of numerical methods.

The present paper is focused on the numerical simulation of particle dissolution in binary alloys. The level set method, first introduced by Osher and Sethian [16], has been chosen to determine the interface location. Consequently the front velocity, only defined in the interface position, is extended into the whole computational domain. Chen *et al.* [21] use a set of advection equations to define a continuous extension of the front velocity. Gibou *et al.* [22] and Kim *et al.* [23] use a constant extrapolation in the normal direction for the same purpose. On the other hand, Adalsteinsson *et al.* [24] use the fast marching method to find an extension of the front velocity. In all these references only finite difference schemes have been used. Chessa *et al.* [25] use an enriched finite element method to solve a phase transformation problem. This method is able to deal with discontinuities in the gradient of the concentration, and requires the update of the enriched nodes as the interface evolves. Furthermore, stabilization terms should be used in order to prevent oscillations in the solution of hyperbolic equations. We simplify the ideas of Chen [21] and Gibou [22] to our problem in order to extend the interface velocity. The hyperbolic equations arising from the level set formulation, are solved with standard finite difference schemes. However, the diffusion equation is solved with the finite element method. The triangulation is locally adapted to the interface position with a cut-cell method, that will be presented. Interpolation between the finite difference and the finite element meshes is not necessary in our algorithm. A similar combination has been recently presented by Tan and Zabaras in [26] for the simulation of dendritic growth.

The outline of the paper is as follows. The governing equations of long-distance diffusion in binary alloys are given in Section 2. The level set method is introduced in Section 3. The numerical solution method is described in detail in Section 4, where several issues related to the level set method will be discussed: an extension of the front velocity in the Cartesian directions is compared with an extension in the normal direction in Section 4.1, the reinitialization of the level set function is presented in Section 4.3 and the cut-cell method is introduced in Section 4.4.1. A number of numerical results is given in Section 5, and conclusions are presented in Section 6.

2 The mathematical model

The as-cast microstructure is simplified to a representative cell Ω containing a diffusive phase Ω_{dp} and a particle Ω_{part} of constant composition c^{part}. The particle dissolves due to Fickian diffusion in the diffusive phase. The concentration at the interface Γ, separating Ω_{part} and Ω_{dp}, is assumed to be given by the constant value c^{sol}. This value is as predicted by thermodynamics. The concentration gradient on the side of Ω_{dp} at Γ causes its displacement. The governing equations and boundary conditions of this problem are:

$$\frac{\partial c}{\partial t}(\mathbf{x},t) = D\Delta c(\mathbf{x},t), \qquad \mathbf{x} \in \Omega_{dp}(t),\ t > 0, \tag{1}$$

$$c(\mathbf{x},t) = c^{part}, \qquad \mathbf{x} \in \Omega_{part}(t),\ t \geq 0, \tag{2}$$

$$c(\mathbf{x},t) = c^{sol}, \qquad \mathbf{x} \in \Gamma(t),\ t \geq 0, \tag{3}$$

$$(c^{part} - c^{sol})v_n(\mathbf{x},t) = D\frac{\partial c}{\partial \mathbf{n}}(\mathbf{x},t), \qquad \mathbf{x} \in \Gamma(t),\ t > 0, \tag{4}$$

where $\mathbf{x}$ denotes a point in the computational domain Ω, D denotes the diffusivity constant, $\mathbf{n}$ is the unit normal vector on the interface pointing outward with respect $\Omega_{part}(t)$ and v_n is the normal component of the velocity of the interface. The initial concentration $c(\mathbf{x},0)$ inside the diffusive phase is given. We assume no flux of concentration through the boundary not being a moving interface, i.e.

$$\frac{\partial c}{\partial \mathbf{n}}(\mathbf{x},t) = 0, \qquad \mathbf{x} \in \partial\Omega_{dp}(t) \setminus \Gamma(t) \quad t > 0, \tag{5}$$

hence mass is conserved. The above equations constitute a so-called Stefan problem.

3 The level set method

The level set method is used to capture the moving interface as the zero level set of a continuous function ϕ, called the level set function. This method was introduced by Osher and Sethian [16] and successfully applied to Stefan problems by Chen *et al.* [21]. The motion of the interface is related to the level set function by

$$\frac{\partial \phi}{\partial t} + v_n||\nabla\phi|| = 0,$$

which is only valid at the interface location. If the front velocity is continuously extended over Ω leading to the vector field $\mathbf{v}$, then the above equation can be generalized to an hyperbolic equation for the level set function

$$\frac{\partial \phi}{\partial t}(\mathbf{x},t) + \mathbf{v}(\mathbf{x},t)\cdot\nabla\phi(\mathbf{x},t) = 0, \quad \mathbf{x} \in \Omega, \quad t > 0. \tag{6}$$

In this framework, the normal vector to the interface can be obtained from the level set function

$$\mathbf{n} = \frac{\nabla\phi}{||\nabla\phi||},$$

which points into the region of $\phi > 0$. The level set function is initialized as a signed distance function:

$$\phi(\mathbf{x},0) = \begin{cases} +\operatorname{dist}(\mathbf{x},\Gamma(0)) & \text{if } \mathbf{x} \in \Omega_{dp}(0), \\ 0 & \text{if } \mathbf{x} \in \Gamma(0), \\ -\operatorname{dist}(\mathbf{x},\Gamma(0)) & \text{if } \mathbf{x} \in \Omega_{part}(0). \end{cases}$$

After advecting the interface using Eq. (6), the level set function is in general not a distance function at the new time step. It might lead to flat/steep gradients of ϕ, leading to inaccurate approximations of the interface curvature (given by $\kappa = \nabla \cdot \mathbf{n}$) and the normal velocity of the interface v_n. Therefore, the level set function is reinitialized by solving in pseudo-time τ the hyperbolic equation

$$\begin{aligned} \frac{\partial \varphi}{\partial \tau} &= \mathrm{S}(\phi)\big(1 - ||\nabla \varphi||\big), \\ \varphi(\mathbf{x},0) &= \phi(\mathbf{x},t), \end{aligned} \tag{7}$$

where S denotes the sign function and φ the reinitialized level set function. Note that the zero level set of $\varphi(\mathbf{x},\tau)$ is the same as that of $\phi(\mathbf{x},t)$, and as $\tau \to \infty$, $||\nabla\varphi|| \to 1$, which characterizes a distance function.

4 The numerical solution method

In our model the motion of the interface is determined by the gradient of the concentration, which can be computed from the solution of the diffusion equation. Local grid refinement and general interface geometry are conveniently handled by the finite element method. Therefore we use a finite element method for the Stefan problem (1)-(5). But for the hyperbolic level set equation (6) finite element methods generate much numerical diffusion (although this might be alleviated by recent progress, see [27, 28]). Therefore we use a finite difference method for (6) and (7). A similar idea has been recently applied by Tan *et al.* [26] to dendritic growth simulations.

For the numerical experiments presented in this paper we will use a finite difference mesh with equal and uniform grid spacing in each coordinate direction, although the algorithm is applicable to general Cartesian meshes. The finite element mesh consists of triangles and tetrahedra in two- and three-dimensional problems, respectively. We will restrict ourselves to finite difference and finite element meshes based on the same mesh points, so that interpolation between meshes is unnecessary. However, the solution method is valid for more general meshes.

Let the level set function ϕ^n and the concentration c^n be known at time t^n. In our numerical implementation, the level set function and concentration at the next discrete time will be obtained by the following algorithm:

1. A continuous extension $\mathbf{v}^n$ of the front velocity is obtained from ϕ^n and c^n.
2. The level set function ϕ^{n+1} at the new time t^{n+1} is obtained by solving Eq. (6).
3. The level set function is reinitialized (*if necessary*) by means of Eq. (7).

4. The concentration c^{n+1} at the new time t^{n+1} is obtained by solving the diffusion equation (1) in the diffusive phase domain determined by $\phi^{n+1} > 0$.

5. Set $n = n + 1$ and go back to 1.

In the next subsections we give details. For simplicity we restrict ourselves to two dimensions. Generalization to three dimensions is straightforward.

4.1 Extension of the front velocity

The front velocity v_n is only defined at the interface location, and an artificial continuous extension of it is required. We present two different extensions. Both require the solution of hyperbolic equations, which define a velocity field in the whole domain Ω. In the first extension procedure, the Cartesian components of the front velocity are decoupled during the extension, as done by Chen *et al.* [21]. In the second extension procedure, the normal velocity is extended in the normal direction, see Gibou *et al.* [22]. These extensions are presented below in more detail.

4.1.1 Extension in the Cartesian directions

Considering the Stefan condition Eq. (4), and the fact that the concentration equals a constant at the interface Eq. (3), the normal velocity vector $\mathbf{v} = v_n\mathbf{n}$ of the interface can be rewritten as

$$\mathbf{v}(\mathbf{x}, t) = \lambda \nabla c(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), t > 0,$$

with $\lambda = \frac{D}{c^{part} - c^{sol}}$. Let $\mathbf{v} = (v_1, v_2)^t$. An extension of the front velocity might be obtained by solving separately

$$\begin{cases} \frac{\partial v_1}{\partial \tau} + \mathrm{S}(\phi\phi_x)\frac{\partial v_1}{\partial x} = 0, \\ \\ v_1(\mathbf{x}, 0) = \lambda \frac{\partial c}{\partial x}(\mathbf{x}, t), \ \mathbf{x} \in \Gamma(t), \end{cases} \qquad \begin{cases} \frac{\partial v_2}{\partial \tau} + \mathrm{S}(\phi\phi_y)\frac{\partial v_2}{\partial y} = 0, \\ \\ v_2(\mathbf{x}, 0) = \lambda \frac{\partial c}{\partial y}(\mathbf{x}, t), \ \mathbf{x} \in \Gamma(t), \end{cases} \tag{8}$$

where $\tau > 0$ denotes a pseudo-time used during the extension of the front velocity, and t is the physical time in the phase transformation. The characteristic lines of (8) point away from the interface location, which implies that a Dirichlet condition is needed on $\Gamma(t)$ only. In three-dimensional problems the extension of the z-component of the front velocity is done similarly.

Let $\mathbf{x}_{ij}$ be a node in the diffusive phase which has one of its left/right neighbors inside the particle. Then $v_{1,ij}$ is computed by the following discretization

$$v_{1,ij} = \lambda \, \mathrm{S}\left(\frac{c^0 - c^{sol}}{c^{part} - c^{sol}}\right) \frac{|c_{ij} - c^{sol}|}{d_{ij}},$$

where

$$d_{ij} = \begin{cases} \phi_{ij} \frac{x_i - x_{i-1}}{\phi_{ij} - \phi_{i-1j}}, & \text{if } \phi_{i-1j} < 0, \\ -\phi_{ij} \frac{x_{i+1} - x_i}{\phi_{i+1j} - \phi_{ij}}. & \text{if } \phi_{i+1j} < 0, \end{cases}$$

is an approximation of the distance between $\mathbf{x}_{ij}$ and the interface located between $\mathbf{x}_{ij}$ and $\mathbf{x}_{i-1j}$ in the first case, and $\mathbf{x}_{ij}$ and $\mathbf{x}_{i+1j}$ in the second case, which has been obtained by linear interpolation of the level set function. Subsequently, this velocity is also designated to the left/right neighbor that lies inside the particle. The velocity v_2 in the adjacent nodes to the interface, i.e. $\phi_{ij-1} < 0 < \phi_{ij}$ or $\phi_{ij} > 0 > \phi_{ij+1}$ in this case, is obtained analogously.

To solve the equations for the extended velocity numerically, the forward Euler method and first order upwind discretizations are used. The CFL condition for stability yields $\Delta\tau < \Delta x$ (resp. $\Delta\tau < \Delta y$ for the extension v_2). Since it suffices to extend the velocity only in a band around the interface to get an accurate update of the moving interface $\Gamma(t)$, we only compute a small number of pseudo-time steps.

4.1.2 Extension in the normal direction

Extension of the front velocity in the normal direction is an alternative. If ϕ is a distance function, the normal velocity $v_n = \lambda \frac{\partial c}{\partial n}$ can be calculated with

$$v_{n,ij} = \lambda \, \mathrm{S}\Big(\frac{c^0 - c^{sol}}{c^{part} - c^{sol}}\Big) \frac{|c_{ij} - c^{sol}|}{\phi_{ij}},$$

in the nodes $\mathbf{x}_{ij}$ within the diffusive phase. To extend this velocity into the particle domain the following equation is solved

$$v_{n,\tau} + \min(\mathrm{S}(\phi), 0) \; \mathbf{n} \cdot \nabla v_n = 0, \quad \mathbf{x} \in \Omega, \; \tau > 0, \tag{9}$$

where τ represents a pseudo-time defined only to carry out the extension. The forward Euler and first order upwind methods are used here. The normal vector is computed using central difference approximations of the gradient of ϕ. The CFL condition is now given by $\Delta\tau \max\big(\frac{|n_1|}{\Delta x} + \frac{|n_2|}{\Delta y}\big) < 1$, where the normal vector is denoted by $\mathbf{n} = (n_1, n_2)^t$. Only a small number of pseudo-time steps is used to obtain the extension of the front velocity in a sufficiently wide band around the interface.

4.2 Advection of the level set function

The level set equation (6) is solved by forward Euler discretization in time and first order upwind discretization in space. This scheme leads to a stability condition on the time steps

$$\Delta t \; \max\left(\frac{|v_1|}{\Delta x} + \frac{|v_2|}{\Delta y}\right) < 1, \tag{10}$$

where $\mathbf{v} = (v_1, v_2)^t$ denotes the continuous extension of the front velocity.

4.3 Reinitialization of the level set function

After moving the interface, Eq. (7) needs to be solved in order to reinitialize the level set function as a signed distance function. Peng *et al.* [29] proposed the modified reinitialization problem:

$$\frac{\partial \varphi}{\partial \tau} = \mathrm{S}(\varphi)\big(1 - ||\nabla\varphi||\big), \quad \text{in } \Omega, \tag{11a}$$

$$\varphi(\mathbf{x},0) = \phi(\mathbf{x},t), \quad \text{in } \Omega, \tag{11b}$$

with pseudo-time $\tau > 0$ and

$$\mathrm{S}(\varphi) = \frac{\varphi}{\sqrt{\varphi^2 + ||\nabla\varphi||^2 \Delta x^2}}, \tag{12}$$

in order to maintain the sign of ϕ during the reinitialization procedure. The monotone Godunov upwind scheme Eq. (13) is used in the numerical solution of (11). The term $1 - ||\nabla\phi||$ of Eq. (11a) (see [29]) is approximated by

$$\mathrm{G}(\varphi_{ij}) = \begin{cases} 1 - \sqrt{\max(a_+^2, b_-^2) + \min(c_+^2, d_-^2)}, & \text{if } \varphi_{ij} > 0, \\ 1 - \sqrt{\max(a_-^2, b_+^2) + \min(c_-^2, d_+^2)}, & \text{if } \varphi_{ij} < 0, \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

where $a = D_x^- \varphi_{ij}$, $b = D_x^+ \varphi_{ij}$, $c = D_y^- \varphi_{ij}$ and $d = D_y^+ \varphi_{ij}$ are used to denote the left/right sided derivatives of φ with respect to x and y respectively, $h_- = \min(h, 0)$ and $h_+ = \max(h, 0)$. The time integration is done by a third order accurate TVD Runge-Kutta scheme [29], and the space derivatives are approximated by a WENO scheme [30].

The reinitialization procedure is terminated when $\big|\mathrm{S}(\varphi_{ij})G(\varphi_{ij})\big|$ is smaller than a prescribed tolerance in a band around the interface location. This criterion is also used to decide whether reinitialization should be applied or not. The number of iterations required to reinitialize the level set function varies during the dissolution process. When the movement of the interface is fast or leads to the breaking of the interface, more iterations are required to achieve the termination criterion. However, when the movement of the interface is smooth and slow, the level set function is only reinitialized after several time steps of the main dissolution process.

4.4 Diffusion

The diffusion equation is solved by the finite element method. The underlying triangulation is adapted according to the interface location with the cut-cell method. The backward Euler method is used for the time discretization. Moreover, the time-stepping has to be adapted in dissolution problems when the interface crosses a node. To illustrate this point, let $\mathbf{x}_{ij}$ be a node which was inside the particle at time t^n and lies in the diffusive phase at time t^{n+1}, hence $\phi_{ij}^n < 0 < \phi_{ij}^{n+1}$. Notice that there is a jump of the concentration from c^{part} to c^{sol} in such a node. The time t_{ij}^* at which the interface was exactly at the node $\mathbf{x}_{ij}$ can be approximated by linear interpolation of the level set function:

$$t_{ij}^* = t^n - \phi_{ij}^n \frac{t^{n+1} - t^n}{\phi_{ij}^{n+1} - \phi_{ij}^n}.$$

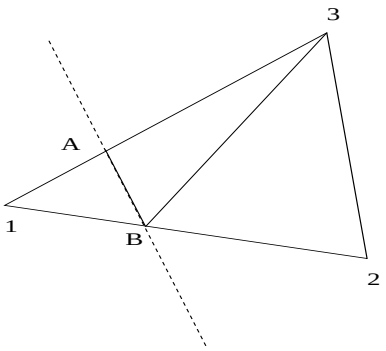
Subsequently, the diffusion equation is discretized for the node $\mathbf{x}_{ij}$ as follows:

$$\frac{c_{ij}^{n+1} - c^{sol}}{t^{n+1} - t_{ij}^*} = D\Delta_h c_{ij}^{n+1},$$

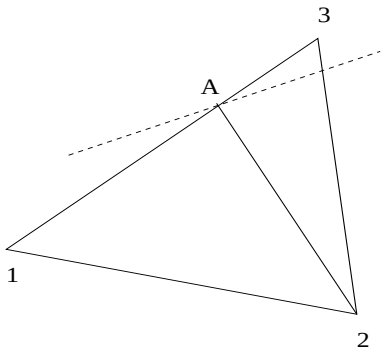
where Δ_h denotes the discretized Laplacian operator.

4.4.1 The cut-cell method

We are dealing with linear elements only (i.e. triangles in 2D and tetrahedrons in 3D). In each vertex a value of the level set function is given. If in an element the level set function changes sign, we know that the element is intersected by the interface. The intersection of the interface with the edges of the element is computed by linear interpolation, so these intersections are uniquely defined. In a triangle we may have between zero and two intersections, in a tetrahedron between zero and four. We assume that within an element the interface is linear. We subdivide each element in subelements, based on the number of intersection points of the edges. For example, if a triangle has one intersection point, it is subdivided into two triangles and in case of two intersection points into three triangles. One of the edges of the new triangles coincides with the interface (see Figure 1(a)).



(a) Subdivision into three elements



(b) Subdivision into two elements

Figure 1: Subdivision of an element by the cut-cell method

In order to avoid ill-shaped elements, only intersections not too close to the vertices are taken into account. If the intersection is near a vertex, we move the intersection point to that vertex, as in Figure 1(b), where the intersection point of edge 2,3 is supposed to be in point 3. Hence, the extra error we make in this way is order $\mathcal{O}(h)$, where h denotes the diameter of the triangulation.

The advantage of the subdivision is that we reconstruct the interface in a relatively simple way, without destroying the original basic mesh. Prescribing boundary conditions on this approximated interface is a straightforward task. After one time step the intersection points are removed and we are back to the basic mesh.

5 Numerical results

This section contains some numerical experiments. The accuracy of the time integration is Δt. The time step is determined by $\Delta t = \min(\Delta t_{CFL}, \Delta t_{min})$, where Δt_{CFL} is given by the CFL condition Eq. (10) and Δt_{min} is used to prevent excessively large time steps. We choose Δt_{min} to be proportional to the mesh width, therefore the complete algorithm is first order accurate in time and space.

The governing equations are such that the mass of the system

$$m(t) = \int_{\Omega} c(\mathbf{x}, t) d\mathbf{x} = c^{part} \int_{\Omega_{part}(t)} d\mathbf{x} + \int_{\Omega_{dp}(t)} c(\mathbf{x}, t) d\mathbf{x}. \tag{14}$$

should be conserved during the complete dissolution process. The numerical computation of Eq. (14) demands the computation of the percentage of a computational cell occupied by $\phi > 0$ (resp. $\phi < 0$), which is done as in [31] by means of the VOF (volume-of-fluid) function. This scheme, equivalent to the trapezoidal rule for continuous functions, provides a second order accurate approximation of $m(t)$.

First we will compare the numerical solution with a similarity solution. Next we will study the convergence to the steady state solution, which is determined by the initial concentration distribution. Mass conservation will be also analyzed. Finally a three-dimensional dissolution problem will be considered, in which the particle breaks up into several subparticles. The performance of the level set method will also be compared in the 2D test problems with the moving grid method [8].

The computational domain Ω will be either the square $[0,1]^2$ for the two-dimensional problems or the cube $[0,1]^3$ for the three-dimensional problems. A uniform finite difference grid will be used, with grid spacing $\Delta x = \Delta y = \Delta z = \frac{1}{N}$. The number of iterations used in the processes involving pseudo-times is as follows. For the extension of the front velocity Eqs. (8) and (9) we use ten pseudo-time steps. The reinitialization iteration Eq. (11) is terminated when $|S(\varphi_{ij})G(\varphi_{ij})| < 0.05$ in a band of width $5\Delta x$ around the interface.

An estimate of the amount of work and memory usage

The amount of work and memory required for the level set method can be summarized by analysis of the following subproblems.

- Solve a diffusion equation. This demands the storage of a symmetric matrix, which only needs to be adapted at the elements adjacent to the moving interface, and the solution of a linear system of equations.
- Find the front velocity. For the Cartesian extension this requires the storage of two vectors (for 2D problems) or three vectors (for 3D problems) with the velocities of the nodes adjacent to the moving interface. For the normal extension this requires the storage of the normal velocity v_n for the nodes inside the diffusive phase.
- Extension of the front velocity. This requires the storage of a tridiagonal matrix (for the

Cartesian extension) or a band matrix (for the normal extension) and a fixed number of matrix-vector multiplications.

- Update of the level set. This requires the storage of a band matrix a matrix-vector multiplication.
- Reinitialization of the level set. Each iteration requires the storage of an extra vector, the storage of a band matrix and a matrix-vector multiplication.

Similarly, we may summarize the amount of work and memory required for the moving grid method as follows:

- Solve a convection-diffusion equation. This requires the storage of a non-symmetric matrix, which should be adapted in the whole domain, and the solution of a linear system of equations.
- Compute the velocity of the interface. This requires the storage of a vector with the normal velocity of the nodes on the moving interface. In addition, the ALE method also needs a globally defined mesh velocity.
- Eventually remeshing is necessary. This requires the interpolation of the concentration between the old and the new mesh.

We compared both methods for a set of two-dimensional problems where the movement of the interface is smooth. Our experience is that the moving grid method is cheaper. If the solution of the diffusion problem is not considered, the level set method demands more memory and computational cost. In particular, the reinitialization can be rather time consuming due to the used numerical scheme (third order Runge-Kutta in time and WENO in space). On the other hand, the solution of the diffusion equation is cheaper with the level set method. Furthermore, after some time steps remeshing is necessary in the moving grid method, which is a time consuming process, especially for three-dimensional problems, and it is difficult to conserve mass (see [7]). In our opinion it is not straightforward to generalize the moving grid method to three-dimensional problems, especially if topological changes occur. This motivates us to use the level set method for three-dimensional problems.

5.1 Planar interface: comparison with a similarity solution

Let us consider an initial rectangular particle $\Omega_{part}(0) = [0, s_0) \times [0, 1]$, where $s_0 = 0.615$, with concentration $c^{part} = 0.45$. The concentration on the moving interface is $c^{sol} = 0.35$, the initial concentration on the diffusive phase $\Omega_{dp}(0) = (s_0, 1] \times [0, 1]$ is $c^0 = 0.3$, and the diffusivity constant $D = 1$. The similarity solution [32] is used to smooth out the discontinuous initial concentration near the interface. Therefore, a time $t_0 = 0.01$ is used to compute the interface position and concentration. The final position of the interface, calculated by a balance of mass, is given by

$$s_\infty = \frac{(c^{part} - c^0)s_0 + c^0 - c^{sol}}{c^{part} - c^{sol}}.$$

The interface remains planar during its evolution. Its position as a function of time is presented in Figure 2 with the similarity solution defined in the infinite domain. Agreement

between the similarity solution and the numerical results is observed at the beginning of the phase transformation, and this correspondence is improved under grid refinement. The boundedness of the computational domain causes divergence of the numerical results from the similarity solution as time evolves. Linear convergence to the steady solution, i.e. to s_∞, is observed as time evolves.

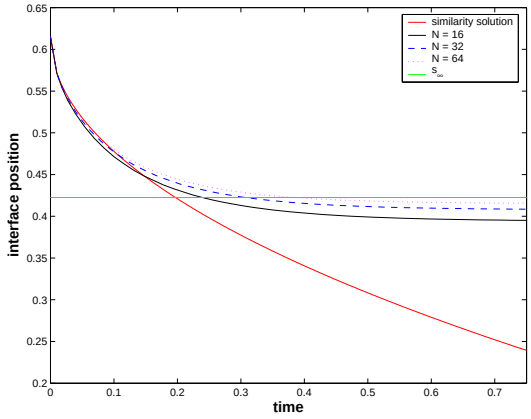


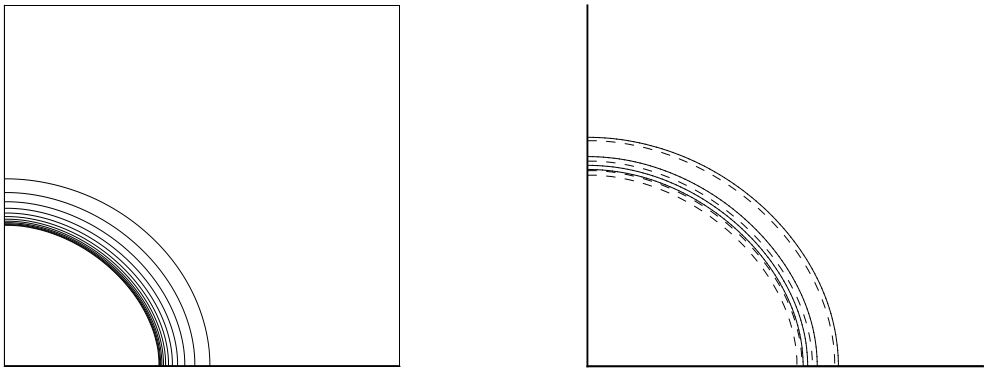
Figure 2: Movement of the interface as a function of time for the planar interface problem.

In Table 1 the relative error in the mass of the system (given by $|m(t) - m(0)|/m(0)$) and the interface position (given by $|s(t) - s_\infty|/|s_\infty|$) at time $t_{end} = 0.75$ is presented. The results obtained with the moving grid method are also given in Table 1. For both the level set and the moving grid methods linear convergence is observed. The meshes for both method have the same number of nodal points. The mesh in the moving grid method is only defined in the diffusive phase $\Omega_{dp}(t)$, and the mesh size varies from 38.5% (initially) to approximately 61% (at final time) of the mesh size in the level set method. This explains the differences in accuracy in Table 1.

	Level Set		Moving Grid	
N	Interface	Mass	Interface	Mass
16	6.483×10^{-2}	7.248×10^{-3}	6.376×10^{-3}	8.239×10^{-4}
32	3.318×10^{-2}	3.749×10^{-3}	3.278×10^{-3}	4.867×10^{-4}
64	1.632×10^{-2}	1.895×10^{-3}	1.867×10^{-3}	3.226×10^{-4}

Table 1: Relative errors in the interface position and mass for the planar interface problem at time $t_{end} = 0.75$. Initial mass of the system $m(0) = 0.39225$ and final position of the interface $s_\infty = 0.4225$.

For this easy test problem, the level set and moving grid methods are equally easy to apply and their accuracy is comparable.



(a) Time evolution of the interface (Cartesian extension).

(b) Snapshots of the interface with the LSM (dashed lines) and the MGM (solid lines).

Figure 3: Circular interface. Left: snapshots at intervals of 0.1s, using the Cartesian extension of the front velocity. Right: interface position at 0.25, 0.5, 0.75 and 1 (from right to left) for the MGM (solid lines) and the LSM (dashed lines).

5.2 The influence of the outer boundaries

Consider the domain $\Omega = [0,1]^2$ where the particle is initially defined by $\Omega_{part}(0) = \{ (x,y) \in \Omega \mid \sqrt{x^2+y^2} < s_0 \}$ where $s_0 = 0.615$. The concentrations are $c^{part} = 0.45$, $c^{sol} = 0.33$ and $c^0 = 0.3$. The diffusivity within the diffusive phase is $D = 1$. If the particle remains circular during the dissolution process, the radius of the particle at the steady solution can be computed by

$$r_\infty = \sqrt{\frac{(c^{part} - c^0)s_0^2 + (c^0 - c^{sol})\frac{4}{\pi}}{c^{part} - c^{sol}}}.$$

However, due to the influence of the outer boundaries this symmetry has not been obtained in the numerical results. This can be observed in Figure 3(a), where the evolution of the interface is presented for intervals of 0.1 seconds until the final time $t_{end} = 1.25$. Note that a larger displacement of the interface is observed in the diagonal direction. The same effect is observed with the moving grid method. The interface position computed with the level set method (LSM) and the moving grid method (MGM) are plotted in Figure 3(b). The picture has been enlarged in order to see the differences between both solutions. The normal extension of the front velocity is used in this case. Figure 3(b) shows that the movement of the interface is faster with the level set method. The front velocity is overestimated due to larger discretization errors. The discretizations obtained with the moving grid method are more accurate since the mesh is only defined in the diffusive phase $\Omega_{dp}(t)$ but has the same number of mesh points, which leads to smaller mesh sizes.

The outer boundary $\{ (x,y) \in \partial\Omega \mid x = 1 \text{ or } y = 1 \}$ causes the non-uniform movement of the interface. This is more clearly observed in Table 2, where the relative errors in the

positions of the interface $r^{(y=0)}(t_{end})$ and $r^{(y=x)}(t_{end})$ with respect to the radius of the steady solution r_∞ are presented. The values $r^{(y=0)}(t)$ and $r^{(y=x)}(t)$ are defined as follows:

$$r^{(y=0)}(t) = \Gamma(t) \cap \{\ (x,y) \in \Omega \mid y = 0\ \}, \tag{15}$$

$$r^{(y=x)}(t) = \Gamma(t) \cap \{\ (x,y) \in \Omega \mid y = x\ \}. \tag{16}$$

The error in the diagonal $r^{(y=x)}$ is always larger than the error in the edge $r^{(y=0)}$, independently of the method used to extend the front velocity. Furthermore, the errors in the edge $r^{(y=0)}$ and in the diagonal $r^{(y=x)}$ are larger when the Cartesian extension of the front velocity is used. The only exception occurs in the error in the edge $r^{(y=0)}$ with $N = 128$. The relative error in mass is also presented in Table 2, which shows that the normal extension of the front velocity is more conservative than the Cartesian extension. Furthermore, the errors in $r^{(y=x)}$ and mass show a linear convergence rate.

	Cartesian Extension			Normal Extension		
N	$r^{(y=0)}$	$r^{(y=x)}$	Mass	$r^{(y=0)}$	$r^{(y=x)}$	Mass
32	3.485×10^{-2}	5.655×10^{-2}	4.798×10^{-3}	2.461×10^{-2}	4.503×10^{-2}	3.846×10^{-3}
64	7.746×10^{-3}	3.092×10^{-2}	2.664×10^{-3}	1.814×10^{-3}	2.511×10^{-2}	2.160×10^{-3}
128	6.381×10^{-3}	1.699×10^{-2}	1.526×10^{-3}	1.016×10^{-2}	1.368×10^{-2}	1.208×10^{-3}

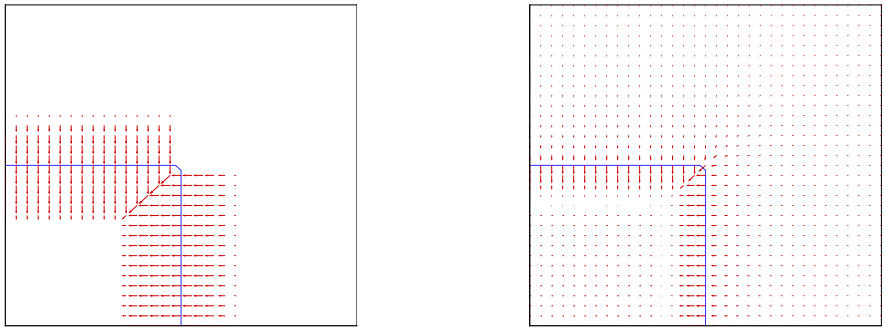
Table 2: Relative errors in the interface position and mass for the circular interface problem at time $t_{end} = 1.25$. The equilibrium radius $r_\infty = 0.3930$, and the initial mass of the system $m(0) = 0.3446$.

5.3 A comparison between Cartesian and normal extensions of the front velocity

Consider the domain $\Omega = [0,1]^2$ where the particle is initially defined by $\Omega_{part} = \{\ (x,y) \in \Omega \mid |x| < b \text{ and } |y| < b\ \}$ where $b = 0.55$. The concentrations are $c^{part} = 0.45$, $c^{sol} = 0.33$ and $c^0 = 0.3$, and the diffusivity inside the diffusive phase $D = 1$. The Cartesian and normal extensions of the front velocity are plotted in Figure 4 together with the interface position at the initial time.

The evolution of the interface with these extensions of the front velocity is presented in Figure 5(a). Minor differences are observed, although the normal velocity produces a slightly slower displacement of the interface around the corner of the initial interface, since the distance of the grid nodes next to the corner in the normal direction is larger than or equal to the distance in the Cartesian directions, leading to smaller estimate of the front velocity. The interface evolution with the moving grid method is presented in Figure 5(b). Contrary to the moving grid method, the level set method handles singularities on the interface in a natural manner.

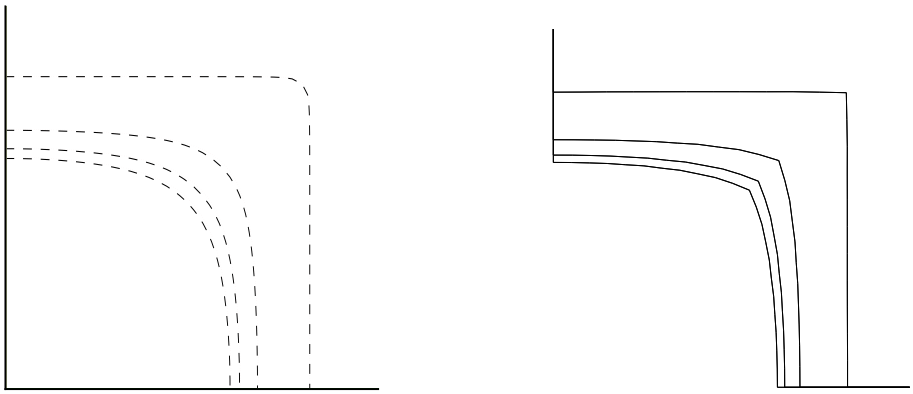
The relative error in the mass at time $t_{end} = 2.5$ is presented in Table 3. The normal extension of the front velocity produces slightly better results. The number of iterations



(a) Cartesian extension of the front velocity at t=0.

(b) Normal extension of the front velocity at t=0.

Figure 4: The initial position of the interface with the Cartesian extension (left) and the normal extension (right) of the front velocity.



(a) Evolution of the interface with the Level Set Method using the Cartesian (dashed curves) and the normal (dotted curves) extensions of the front velocity

(b) Evolution of the interface with the Moving Grid Method

Figure 5: Interface position at times 0.005, 0.205, 0.405 and 0.605 using the Level Set Method (left) and the Moving Grid Method (right).

required to reinitialize the level set function to a signed distance function is slightly lower for the normal extension than for the Cartesian extension of the front velocity.

N	Cartesian Extension	Normal Extension
16	7.438×10^{-3}	5.562×10^{-3}
32	6.082×10^{-3}	4.741×10^{-3}
64	2.538×10^{-3}	1.620×10^{-3}
128	1.515×10^{-3}	9.490×10^{-4}

Table 3: Relative errors in the mass for the squared interface problem. Initial mass: $m(0) = 0.3458$.

5.4 A 3D test problem

Consider a cylindrical particle in the cube $\Omega = [0,1]^3$. Let us assume that the surface of this cylindrical particle has been perturbed. The concentrations are $c^{part} = 0.45$, $c^{sol} = 0.33$ and $c^0 = 0.3$, and the diffusivity constant $D = 1$. The dissolution of the particle, using the Cartesian extension of the front velocity, is presented in Figure 6. The particle breaks into four almost spherical particles, of which the two particles in the extremes dissolve faster, since the flux of concentration is larger there than in the central part. The change of topology is handled in a natural manner with the level set method, due to its implicit representation of the interface. However, it is our experience with the moving grid method that merging/breaking of interfaces requires an arduous implementation task (although remarkable improvements have been obtained in this respect, see [11]).

6 Conclusions

We considered the dissolution process in solid state phase transformations. The level set method was used to model the resulting Stefan problem. A combination of finite difference and finite element methods has been shown to work. Two alternative extensions of the front velocity have been used. Both are easy to implement, and their extension to three-dimensional problems is straightforward. It has been shown that the normal extension of the front velocity gives slightly more accurate results and requires slightly fewer pseudo time-steps in the reinitialization than the Cartesian extension. The numerical scheme has been shown to be first order accurate, since the front velocity and the update of the level set are carried out by first order accurate schemes. Furthermore, singularities and topological changes are handled in a natural way, due to the implicit representation of the moving interface, in contrast with the moving grid method.

Acknowledgments

This research is financially supported by the Dutch Technology Foundation (STW). The authors thank prof. dr. ir. S. van der Zwaag for suggesting an interesting 3D test problem,

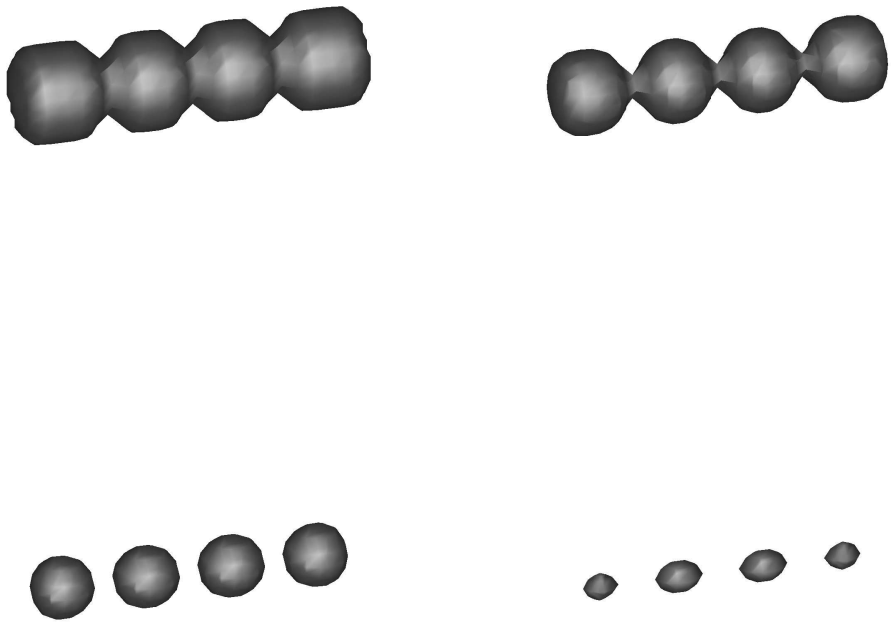


Figure 6: Dissolution of an initially perturbed cylindrical particle. Time evolution follows from left to right and from up to down.

which is relevant for materials science. The authors also want to thank prof. dr. ir. P. Wesseling for carefully reading this manuscript and giving valuable suggestions.

References

[1] M.J. Whelan, On the kinetics of particle dissolution, *Metal Sci. J.* **3** (1969) 95-97

[2] U.L. Baty, R.A. Tanzilli, R.W. Heckel, Dissolution kinetics of CuAl2 in Al-4Cu alloy, *Metall. Trans.* **1** (1970) 1651-1656

[3] U.H. Tundal, N. Ryum, Dissolution of particles in binary alloys: Part I, Computer simulations, *Metall. Trans.* **23A** (1992) 433-449

[4] H.B. Aaron, G.R. Kotler, Second phase dissolution, *Metall. Trans.* **2** (1972) 393-407

[5] F.V. Nolfi Jr., P.G. Shewmon, J.S. Foster, The dissolution and growth kinetics of spherical precipitates, *Trans. Metall. Soc. AIME* **245** (1969) 1427-1433

[6] F.J. Vermolen, S. van der Zwaag, A numerical model for the dissolution of spherical particles in binary alloys under mixed mode control, *Mater. Sci. Eng. A* **220** (1996) 140-146

[7] C. Vuik, A. Segal and F.J. Vermolen, A conserving discretisation for a Stefan problem with an interface reaction at the free boundary, *Comput. Visual. Sci.* **3** (2000) 109-114

[8] G. Segal, K. Vuik, F. Vermolen, A conserving discretization for the free boundary in a two-dimensional Stefan problem, *J. Comp. Phys.* **141** (1998) 1-21

[9] W.D. Murray, F. Landis, Numerical and machine solutions of transient heat-conduction problems involving melting or freezing, *Trans. ASME (C) J. Heat Trans.* **81** (1959) 106-112

[10] D. Juric, G. Tryggvason, A front-tracking method for dendritic solidification, *J. Comp. Phys.* **123** (1996) 127-148

[11] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas and Y.-J. Jan, A front-tracking method for the computation of Multiphase flow, *J. Comp. Phys.* **169** (2001) 708-759

[12] V.R. Voller, An implicit enthalpy solution for phase change problems: with application to a binary alloy solidification, *Appl. Math. Modelling* **11** (1987) 110-116

[13] Y.C. Lam, J.C. Chai, P. Rath. H. Zheng, M.V. Murukeshan, A fixed-grid method for chemical etching, *Ont. Comm. Heat MAss Transfer* **31** (2004) 1123-1131

[14] M. Fabbri, V.R. Voller, The phase-field method in the sharp-interface limit: comparison between model potentials, *J. Comp. Phys.* **130** (1997) 256-265

[15] J.A. Mackenzie, M.L. Robertson, A moving mesh method for the solution of the one-dimensional phase field equations, *J. Comp. Phys.* **181** (2002) 526-544

[16] S. Osher and J.A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *J. Comput. Phys.* **79** (1988) 12-49

[17] J.A. Sethian, *Level set methods and fast marching methods*, Cambridge University Press, New York, 1999

[18] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag, New York, 2003

[19] H.D. Ceniceros, A.M. Roma, A multi-phase flow method with a fast, geometry-based fluid indicator, *J. Comp. Phys.* **205** (2005) 391-400

[20] J. Crank, *Free and moving boundary problems*, Clanderon Press, Oxford, 1984

[21] S. Chen, B. Merriman, S. Osher and P. Smereka, A Simple Level Set Method for Solving Stefan Problems, *J. Comp. Phys.* **135** (1997) 8-29

[22] F. Gibou, R. Fedkiw, R. Caflisch and S. Osher, A Level Set Approach for the Numerical Simulation of Dendritic Growth, *J. Sci. Comp.* **19** Vol. 1-3 (2003) 183-199

[23] Y.-T. Kim, N. Goldenfeld and J. Dantzig, Computation of dendritic microstructures using a level set method, *Phys. Rew. E* **62** Nr. 2 (2000) 2471-2474

[24] D. Adalsteinsson, J.A. Sethian, The fast construction of extension velocities in level set methods, *J. Comp. Phys.* **148** (1999) 2-22

[25] J. Chessa, P. Smolinski and T. Belytschko, The extended finite element method (XFEM) for solidification problems, *Int. J. Numer. Meth. Engng.* **53** (2002) 1959-1977

[26] L. Tan and N. Zabaras, A level set simulation of dendritic solidification with combined features of front-tracking and fixed-domain methods, *J. Comp. Phys.* **211** 36-63 (2006)

[27] D. Kuzmin, S. Turek, High-resolution FEM-TVD schemes based on a fully multidimensional flux limiter, *J. Comp. Phys.* **198** (2004) 131-158

[28] D. Kuzmin, M. Möller, S.Turek, High-resolution FEM-FCT schemes for multidimensional conservations laws, *Computer Meth. Appl. Mech. Engrg.* **193** (2004) 4915-4946

[29] D. Peng, B. Merriman, S. Osher, H. Zhao and M. Kang, A PDE-Based Fast Local Level Set Method, *J. Comput. Phys.* **155** (1999) 410-438

[30] G.S. Jiang and D. Peng, Weighted ENO schemes for Hamilton-Jacobi equations, *SIAM J. Sci. Comput.* **21** (2000) 2126-2143

[31] S.P. van der Pijl, A. Segal, C. Vuik and P. Wesseling, A mass-conserving Level-Set method for modelling of multi-phase flows, *Inter. J. Numer. Meth. Fluids* **47** (2005) 339-361

[32] E. Javierre, C. Vuik, F. Vermolen and S. van der Zwaag, A comparison of numerical models for one-dimensional Stefan problems, *To appear in J. Comp. Appl. Math*