Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 224 (2007) 222–240
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A level set method for three dimensional vector Stefan problems: Dissolution of stoichiometric particles in multi-component alloys

E. Javierre *, C. Vuik, F.J. Vermolen, A. Segal

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

Received 15 September 2006; received in revised form 23 January 2007; accepted 30 January 2007
Available online 17 February 2007

Abstract

A sharp interface method is proposed for the dissolution of stoichiometric particles in multi-component alloys occurring during the heat treatments of as-cast aluminium alloys prior to hot extrusion. In the mathematical model, a number of non-linearly coupled diffusion equations are given to determine the position of the particle interface and the interfacial concentrations. A level set method is used to determine the interface position at each time step. Once the front position is known, a fixed-point iteration is used to find the interfacial concentrations. The model is applicable to both complete and incomplete dissolution in two and three spatial dimensions, and handles topological changes in a natural fashion. The numerical solution is compared with steady-state and self-similar exact solutions available for simple particle geometries. Subsequently, the model is applied to an AlMgSi-alloy to investigate the influence of the particle morphology in the dissolution kinetics.

Keywords: Particle dissolution; Multi-component alloy; Vector Stefan problem; Level set method; Cut-cell method; Velocity extension

1. Introduction

In the thermal processing of both ferrous and non-ferrous alloys, homogenization of the existing microstructure by annealing at such a high temperature that unwanted precipitates are fully dissolved, is required to obtain a microstructure suited to undergo heavy plastic deformation as well as providing an optimal starting condition for a subsequent precipitation hardening treatment. Such a homogenization treatment is for example applied prior to hot-rolling of Al killed construction steels, HSLA steels, all engineering steels, as well as in processing aluminum extrusion alloys. Although precipitate dissolution is not the only metallurgical process taking place during homogenization, it is often the most critical of the processes occurring. The minimum

* Corresponding author. Tel.: +31 (0) 15 2783732; fax: +31 (0) 15 2787209.
E-mail address: e.javierre@ewi.tudelft.nl (E. Javierre).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.01.038

temperature at which the annealing should take place can be determined from thermodynamic analysis of the phases present. However, the minimum annealing time at this temperature is not a constant but depends on particle size, particle geometry, particle concentration, overall composition, etc. To make the homogenization treatment more efficient, it is highly desirable to have robust physical models for the kinetics of particle dissolution as a function of thermodynamics and thermokinetics data as well as particle morphology and microstructural dimensions. Using such models, the minimum annealing times and optimum heating strategies can be calculated a priori, rather than be determined experimentally at great cost. Furthermore quantitative insight into the relation between the microstructure of the alloy and its mechanical properties is necessary. Dumont et al. [9,10] carry out an experimental study to link the volume fraction and size distribution of the particles with the hardness in the aluminium alloys.

Due to the scientific and industrial relevance of being able to predict the kinetics of particle dissolution, many models of various complexities have been presented and experimentally validated. The early models of particle dissolution (and/or growth) based on long-distance diffusion [42,3,32] consisted of analytic solutions in unbounded mediums under the assumption of local equilibrium at the interface. These models assume that the processes at the interface between the adjacent phases take place instantaneously, and therefore provide an upper bound for the dissolution kinetics. Non-equilibrium models incorporating the interface reactions are due to Nolfi et al. [23] and Aaron and Kotler [1], among others. More recently, modelling of particle dissolution has been extended to the introduction of multi-component particles [27,31,38]. All the analyses indicate that the addition of secondary alloying elements can influence the dissolution kinetics strongly.

In the abovementioned papers, particle dissolution was viewed as a Stefan problem with a sharp interface separating the constitutive phases. An alternative to this modellization is the phase-field approach, which is derived from a minimization of a free energy functional employing a diffuse interface between the phases in contact. Furthermore, the interface conditions are embedded in the formulation of the phase transformations on the diffuse interface, and the sharp interface problem is recovered in the limit of vanishing interface thickness [6]. This approach has been used by Wheeler et al. [41] to simulate phase transformations in binary alloys. More recently, Grafe et al. [13] and Kovačević and Šarler [18] have applied the phase-field method to solid-state phase transformations in multi-component alloys. Their results for one-dimensional problems show a perfect agreement with the sharp interface models. However, phase-field models have some drawbacks. Physically justifiable parameter values in the energy functional are generality derived from fitting simulations to experiments. Derivation of these parameters from the information available in thermodynamic databases is generally not yet possible. Furthermore, a fine resolution along the diffuse interface is necessary to accurately resolve the problem. Hence, adaptive mesh techniques [20,5] need to be implemented. Unfortunately, these strategies impose severe bounds on the time-stepping and consequently simulations require large computational times.

The aim of this study is to develop a numerical method to simulate particle dissolution in multi-component alloys in three spatial dimensions. As far as we know, the methods currently capable of dealing with higher-dimensional problems have been developed for binary alloys [28,35] only. The methods for dissolution in multi-component alloys either assume geometrical reductions in order to solve a one-dimensional problem [33,18] or only consider rather simple geometries [13] in two-dimensions. Furthermore, we are interested in studying the evolution of the interface concentrations under different thermal conditions and for different particle geometries. This motivates us to search for a suitable numerical method based on a sharp interface model.

Several numerical methods exist for solving Stefan problems with a sharp interface related to particle dissolution and phase transformations. A survey on classical methods for Stefan problems is given by Crank [8]. More advanced are the so-called front-tracking and front-capturing methods. The first use an explicit representation of the interface given by a set of points laying on the front which should be updated each time step and which may or may not be connected to the background mesh. Murray and Landis [21] introduced a moving grid method for one-dimensional solidification problems. In their approach, the computational mesh is connected to the interface and hence updated each time step. This approach was extended by Segal et al. [28] to solve particle dissolution in two dimensions. Juric et al. [16] combine a fixed grid method with a moving mesh on the interface to solve solidification problems. In their method, information is transferred from the discrete interface to the fixed mesh with the immersed boundary method.

On the other hand, in front-capturing methods the interface is identified with a specific property of a mark function. A great advantage of such a representation is that topological changes are easily handled. Within this class of methods, the most widely used are the enthalpy method and the level set method. Enthalpy methods have been successfully applied to solidification problems by Voller et al. [39,40] and Nedjar [22] and recently to solid state phase transformations by Lam et al. [19]. The heat equation is replaced by a balance of energy which relates the enthalpy with the heat of the system. The resulting discretized system is highly non-linear and advanced techniques are required for its solution [11]. The interface location is identified in a post-processing step as a corresponding isotherm of the heat distribution. As a consequence, 'stair-casing' effects seem to be inherent to enthalpy methods, which may result into inaccurate approximations to the interface, especially near steady-state regimes.

In this work, we use the level set method, firstly introduced by Osher and Sethian [25], to capture the interface separating the adjacent phases as the zero level set of a continuous function. Extensive descriptions of this method and its applications have been presented in [29,24]. Sussman et al. [30] use the fluid velocity to advect the interface in order to simulate two-phase flows. In other applications, however, the front velocity has to be extended onto the computational domain in order to update the front position. Different numerical procedures have been developed in the later years. Among others, Chen et al. [7] advect the Cartesian components of the front velocity in the proper upwind direction, Kim et al. [17] use a constant extrapolation in the normal direction and Adalsteinsson and Sethian [2] employ a fast marching method for the same purpose. Sussman et al. [30] introduce an additional hyperbolic equation to re-initialize the function capturing the interface as a distance function. Peng et al. [26] propose an improvement to Sussman's equation which ensures that mesh points do not change phase due to re-initialization. Other possibilities are the fast marching [29] and the fast sweeping [43] methods.

This paper is organized as follows. First, we introduce the three-dimensional model for particle dissolution in multi-component alloys. Subsequently, we present our numerical method, based on the level set method, in detail. Special attention will be given to the solution of the non-linearly coupled diffusion problems and to the extension of the front velocity. Finally, the performance of the method will be evaluated with a number of test problems, and the conclusions will be addressed.

2. The physical problem

After manufacturing, an alloy is cast into a mould. After subsequent solidification, the state of the alloy is referred to as the as-cast state. The as-cast microstructure is simplified into a representative cell Ω containing a diffusive phase Ω_{dp} and a particle Ω_{part}. The phases Ω_{dp} and Ω_{part} are disjoint, and the interface separating them is denoted by Γ. The multi-component alloy consists of aluminum and p secondary chemical species. Hence, the diffusive phase is also commonly referred to as the aluminum-rich phase since aluminum is the solvent in which the particle dissolves.

The dissolution is governed by the diffusional transport of atoms from the particle into the diffusive phase, given by Fick's Second Law

$$\frac{\partial c_i}{\partial t}(\mathbf{x},t) = \sum_{j=1}^{p} \nabla \cdot (D_{ij} \nabla c_j(\mathbf{x},t)), \quad \mathbf{x} \in \Omega_{\text{dp}}(t),\ t > 0,\ i \in \{1,\dots,p\}, \tag{1}$$

where c_i denotes the concentration of the ith alloying element, $i \in \{1,\dots,p\}$. The coefficients D_{ii} are the diffusion coefficients, whereas D_{ij} for $i \neq j$ denote the cross-diffusion coefficients, which are a measure of the interaction of diffusion within consecutive species. Thus, if $D_{ij} < 0$ then the jth alloying element deteriorates (i.e. delays) the diffusion of the ith species, whereas $D_{ij} > 0$ implies that the jth alloying elements facilitates (i.e. accelerates) the diffusion of the ith species. In the present work, the cross-diffusion coefficients are neglected. Hence, $D_{ij} = 0$ for $i \neq j$, and then the classical diffusion equations are recovered for each chemical species. Without loss of generality, we denote $D_i = D_{ii}$ for $i = 1,\dots,p$, and we postulate that the initial concentration $c_i(\mathbf{x},0)$ in the diffusive phase is given for all the species.

The particle is assumed to remain stoichiometric during the process. Hence, the particle concentrations stay constant

$$c_i(\mathbf{x},t) = c_i^{\text{part}}, \quad \mathbf{x} \in \Omega_{\text{part}}(t),\ t \geqslant 0,\ i \in \{1,\dots,p\}. \tag{2}$$

Furthermore, under the assumption of local thermodynamic equilibrium, the concentrations c_i^{sol} on the moving interface Γ satisfy the hyperbolic relation [37]

$$\prod_{i=1}^{p} (c_i^{\text{sol}}(\mathbf{x},t))^{\tilde{n}_i} = \mathscr{K}(T), \quad \mathbf{x} \in \Gamma(t),\ t > 0, \tag{3}$$

which is derived from the Gibbs free energy of the stoichiometric particle, where $\tilde{n}_i$ denotes the stoichiometric number of the ith chemical species and $\mathscr{K}(T)$ denotes the solubility product which depends on the temperature T through an Arrhenius relation. Furthermore, no flux of the alloying elements is allowed through the boundary not being Γ

$$\frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x},t) = 0, \quad \mathbf{x} \in \partial\Omega_{\text{dp}}(t) \setminus \Gamma(t),\ t > 0,\ i \in \{1,\dots,p\}. \tag{4}$$

Mass conservation for all the chemical species implies that the normal component v_n of the interface velocity is given by

$$(c_i^{\text{part}} - c_i^{\text{sol}}(\mathbf{x},t)) v_n(\mathbf{x},t) = D_i \frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x},t), \quad \mathbf{x} \in \Gamma(t),\ t > 0,\ i \in \{1,\dots,p\}, \tag{5}$$

where $\mathbf{n}$ is the unit normal vector on the interface pointing outward with respect to $\Omega_{\text{part}}(t)$. Note that Eqs. (5) implicitly impose that

$$\frac{D_i}{c_i^{\text{part}} - c_i^{\text{sol}}(\mathbf{x},t)} \frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x},t) = \frac{D_j}{c_j^{\text{part}} - c_j^{\text{sol}}(\mathbf{x},t)} \frac{\partial c_j}{\partial \mathbf{n}}(\mathbf{x},t), \quad \mathbf{x} \in \Gamma(t),\ t > 0,\ i \neq j. \tag{6}$$

The problem as posed by (1)–(6) is commonly referred to as a vector valued Stefan problem, where the concentrations in the diffusive phase c_i, the interface position Γ and the interface concentrations c_i^{sol} are to be determined. We remark that the implementation of cross-diffusion coefficients is straightforward if the diffusion matrix is diagonalizable. The only change in our present code would be the adaptation of Eq. (3), where the eigenvectors of the diffusion matrix would have to be incorporated [34].

2.1. A note on the equilibrium conditions at the interface

According with the definition given in [12], our local equilibrium assumption is equivalent to the so-called orthoequilibrium condition at the interface. The same conclusion can be drawn from the discussion in [14], spite of the controversy that this term raises within different authors.

A simplification to our model can be obtained by imposing the so-called paraequilibrium condition [12] at the interface separating the particle and the surrounding diffusive phase. Under this assumption, only the diffusion of the (faster) interstitial alloying elements is considered and the diffusion of the (slower) substitutional alloying elements is disregarded. Hence, the ternary system is treated as a binary system in which only the faster alloying element is considered. The concentration profile of the substitutional alloying element is assumed not to change during the dissolution process. In a multi-component alloy the number of alloying elements is decreased, hence the complexity of the problem is reduced. Since the slow elements are assumed to diffuse "infinitely" slowly, that is, their diffusion coefficients are set equal to zero, a direct consequence of the assumption of paraequilibrium is that the dissolution rate is predicted to be lower than it actually should be if local equilibrium for all alloying elements (i.e. orthoequilibrium) would be treated in the model.

Consider a ternary alloy for the sake of the illustration. Suppose that the interstitial species has a diffusion coefficient D_1 and the substitutional alloying element diffuses with rate D_2. Then, under the assumption of paraequilibrium, the concentration profile c_2 does not change in time during the entire dissolution process. This implies for the dissolution case that the interface concentration of element 2 is given by $c_2^{\text{sol}} = c_2^{\text{part}}$ at all stages of the dissolution process. Since the interface concentrations satisfy the thermodynamic relationship (3) sketched in Fig. 1(a), c_1^{sol} is defined and stays constant during the entire dissolution process. The dissolution times computed under the assumptions of paraequilibrium and local equilibrium are plotted in Fig. 1(b). We

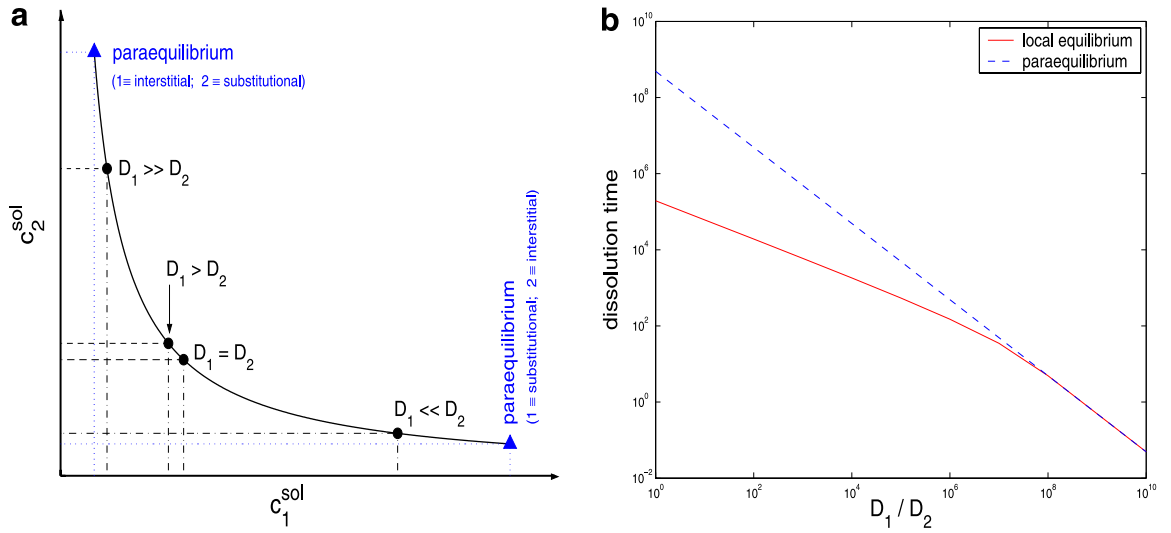

Fig. 1. Differences between the models based on the local equilibrium and paraequilibrium conditions at the interface: (a) interfacial concentrations; (b) dissolution times.

remark that if $D_2 = D_1$, the assumption of paraequilibrium only makes sense as a limit case to show the consequences for small ratios of the diffusion coefficients. The differences between the two models are large when the diffusion coefficients are alike, which shows the advantages of our model. As expected, the results from both models converge as D_1/D_2 tends to infinity. This is reasonable since the assumption of paraequilibrium is common in the case of high ratios of the diffusion coefficients. We finally remark that the curves in Fig. 1(b) have been computed for a planar particle in an unbounded domain of computation for which we used its self-similar exact solution [33].

3. The computational method

The position of the moving interface Γ separating the particle of the diffusive phase is determined using a level set method [25]. Here, the interface Γ is identified by the zero level set of a signed distance function ϕ and its motion follows from a transport equation for ϕ. Furthermore, p diffusion equations have to be solved simultaneously in a domain, the diffusive phase, which is determined by the interface Γ. In addition, due to the interface conditions (3) and (6), the diffusion problems are non-linearly coupled. In our approach, we first approximate the interface position and subsequently we solve the coupled diffusion problems with an iterative method on the interface concentrations.

The numerical solution of the governing equations will be handled as follows. The hyperbolic equations inherited from the level set formulation are discretized with finite difference schemes over a Cartesian mesh, whereas the diffusion problems are solved with standard Galerkin finite element methods using linear elements. This combination allows us to work out the governing equations in a simple fashion, avoiding the stabilization techniques needed to solve hyperbolic equations with finite elements, and the toilsome discretizations of the interface conditions of the finite difference approximations.

3.1. A level set method for the moving interface

A scalar continuous function ϕ is introduced into the governing equations in order to implicitly capture the moving interface Γ as its zero level set:

$$\phi(\mathbf{x}, t) = 0 \Longleftrightarrow \mathbf{x} \in \Gamma(t), \; t \geqslant 0.$$

This scalar indicator function ϕ is commonly referred to as the level set function. The motion of the interface is related to the level set function by

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

where v_n denotes the normal component of the interface velocity given by the Stefan condition Eq. (5). The above equation is only valid at the interface because the velocity v_n is only defined at the front position itself. However, if the front velocity is continuously extended over Ω leading to the artificial vector field $\mathbf{v}$, then the above equation can be generalized to an hyperbolic equation for the level set function on the whole computational domain

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, t) + \mathbf{v}(\mathbf{x}, t) \cdot \nabla \phi(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \Omega, \ t > 0. \tag{7}$$

The technique that we use to compute the velocity field $\mathbf{v}$ is explained in detail in Section 3.3. The numerical approximation of Eq. (7) is done with finite differences on the Cartesian grid. The time integration is carried out with the Euler forward method, whereas the space derivatives are discretized with a first order accurate upwind scheme. Hence, this scheme leads to a CFL stability condition on the time step:

$$\Delta t \max\left(\frac{|v_1|}{\Delta x_1} + \frac{|v_2|}{\Delta x_2} + \frac{|v_3|}{\Delta x_3}\right) < 1. \tag{8}$$

In this framework, geometrical quantities on the interface are easily obtained from the level set function. Thus, the unit normal vector is given by

$$\mathbf{n} = \frac{\nabla \phi}{\|\nabla \phi\|}, \tag{9}$$

which points into the region of $\phi > 0$, and the mean curvature of the interface is defined by $\kappa = \nabla \cdot \mathbf{n}$.

3.1.1. Re-initialization of the level set function

The level set function is initialized as a signed distance function:

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

However, after advecting the interface using Eq. (7), the level set function is in general not a distance function. This may lead to flat/steep gradients of ϕ in the neighbourhood of the interface, resulting into inaccurate approximations of the front velocity. In order to prevent these effects, we re-initialize the level set function to a distance function. This is achieved by solving in pseudo-time τ the hyperbolic equation [26]

$$\frac{\partial \varphi}{\partial \tau} = \widetilde{S}(\varphi)(1 - \|\nabla \varphi\|), \tag{10}$$

with initial condition $\varphi(\mathbf{x}, 0) = \phi(\mathbf{x}, t)$, where $\widetilde{S}$ denotes the smooth sign function

$$\widetilde{S}(\varphi) = \frac{\varphi}{\sqrt{\varphi^2 + \|\nabla \varphi\|^2 \Delta \bar{x}^2}}, \tag{11}$$

$\Delta \bar{x} = \max(\Delta x_1, \Delta x_2, \Delta x_3)$ and φ the reinitialized level set function. Note that the zero level set of φ is the same of ϕ, and in the steady-state $\|\nabla \varphi\| = 1$, which is a characteristic of distance functions. Hence, after Eq. (10) is solved, ϕ is replaced by φ.

In the numerical solution of Eq. (10) we use a third order accurate TVD Runge–Kutta scheme [26] for the time integration and a fifth order WENO scheme [15] for the space derivatives. The term $G(\varphi) = 1 - \|\nabla \varphi\|$ is approximated with

$$G(\varphi_{ijk}) = \begin{cases} 1 - \sqrt{\max(a_+^2, b_-^2) + \max(c_+^2, d_-^2) + \max(e_+^2, f_-^2)}, & \text{if } \varphi_{ijk} > 0, \\ 1 - \sqrt{\max(a_-^2, b_+^2) + \max(c_-^2, d_+^2) + \max(e_-^2, f_+^2)}, & \text{if } \varphi_{ijk} < 0, \\ 0, & \text{otherwise}, \end{cases} \tag{12}$$

where $a = D^-_{x_1}\varphi_{ijk}$, $b = D^+_{x_1}\varphi_{ijk}$, $c = D^-_{x_2}\varphi_{ijk}$, $d = D^+_{x_2}\varphi_{ijk}$, $e = D^-_{x_3}\varphi_{ijk}$ and $f = D^+_{x_3}\varphi_{ijk}$ are used to denote the left/right sided derivatives of φ with respect to x_1, x_2 and x_3. Further, $a_- = \min(a, 0)$ and $a_+ = \max(a, 0)$, and the same definitions apply for $b_\pm$, $c_\pm$, $d_\pm$, $e_\pm$ and $f_\pm$.

In order to improve the performance of our method, we reinitialize the level set function only whenever it deviates more than a prescribed tolerance from a distance function. We measure this deviation with the infinity norm of $\widetilde{S}(\phi)G(\phi)$ along a band around of the interface. We use the same criterion to terminate the reinitialization iteration. The sign function $\widetilde{S}$ is introduced in the termination criteria to smear out large errors occurring in singularity points (such as corners or cusps), since there $G(\phi)$ does not converge to zero. In our applications, we use a band width of $5\Delta\bar{x}$ and we allow a maximum deviation of 0.05. Hence, the number of iterations required to reinitialize the level set function varies during the dissolution process. If the movement of the interface is fast or leads to the breaking up of the interface, more iterations are required. However, if 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.

3.2. The coupled diffusion problems

The hyperbolic relation (3) on the interface concentrations and the Stefan conditions (5) on the interface velocity are reformulated as the zero of the function $\mathbf{f} : \mathbb{R}^p_+ \to \mathbb{R}^p$ given by

$$\begin{cases} f_1(\mathbf{c}^{\mathrm{sol}}) = \prod_{i=1}^{p} (c_i^{\mathrm{sol}})^{\tilde{n}_i} - \mathscr{K}(T), \\ f_i(\mathbf{c}^{\mathrm{sol}}) = \frac{D_i}{c_i^{\mathrm{part}} - c_i^{\mathrm{sol}}} \frac{\partial c_i}{\partial \mathbf{n}} - \frac{D_{i-1}}{c_{i-1}^{\mathrm{part}} - c_{i-1}^{\mathrm{sol}}} \frac{\partial c_{i-1}}{\partial \mathbf{n}}, \quad i = 2, \ldots, p, \end{cases} \tag{13}$$

where the vectorial notation has been embraced for $\mathbf{c}^{\mathrm{sol}}$. Note that $\mathbf{f}$ is defined on the interface, i.e. $\mathbf{c}^{\mathrm{sol}} \equiv \mathbf{c}^{\mathrm{sol}}(\mathbf{x}, t)$ along the interface points $\mathbf{x} \in \Gamma(t)$. Then, provided that the interface location $\Gamma(t)$ is known, the following problem **(P)** has to be solved in order to find the alloying elements concentration on the diffusive phase c_i and at the moving interface c_i^{sol} at a certain time $t > 0$.

$$\begin{cases} \frac{\partial c_i}{\partial t}(\mathbf{x}, t) = D_i \Delta c_i(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_{\mathrm{dp}}(t), \ i = 1, \ldots, p, \\ \frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega_{\mathrm{dp}}(t) \setminus \Gamma(t), \ i = 1, \ldots, p, \\ c_i(\mathbf{x}, t) = c_i^{\mathrm{sol}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \ i = 1, \ldots, p, \\ \mathbf{f}(\mathbf{c}^{\mathrm{sol}}(\mathbf{x}, t)) = \mathbf{0}, \quad \mathbf{x} \in \Gamma(t). \end{cases} \tag{P}$$

In our method, we will use the Euler backward method for the time integration and a standard Galerkin finite element method, with linear elements, for the space derivatives of the diffusion equations. Furthermore, the non-linear coupling given by $\mathbf{f}$ is solved with an iterative method on the interface concentrations. Hence, a proper numerical solution of the coupled diffusion problems **(P)** demands that several issues are treated meticulously. First, the background finite element mesh needs to be adapted to the interface position in order to obtain a triangulation which is conformed to the diffusive phase and to implement the interface conditions easily. Second, the time discretization of the diffusion equations must be locally adapted for the mesh points that have joined the diffusive phase within the current time step because of the jump discontinuity of the concentrations across the interface. And third, a suitable iterative method shall be defined in order to obtain a robust and efficient method of which the extension to higher dimensional problems is affordable.

3.2.1. A simple reconstruction of the particle-diffusive phase interface

We use a cut-cell method to adapt the basic triangulation to the interface position prior to solve the diffusion problem **(P)**. Only linear elements (i.e. triangles in 2D and tetrahedra in 3D) are used in our finite elements approximation. The sign of the level set function on the vertices of an element indicates whether this element is intersected by the interface. The intersection points (i.e. interface points) are computed by linear interpolation of the level set function along the edges of the elements. Hence, these intersections are uniquely defined. Thus, a triangle might have either zero or two intersections, whereas a tetrahedron might have zero, three or four intersections. Each intersected element is subdivided into sub-elements based on the subdivision

of its edges. The division of edges is determined by the intersection points. The division of triangles follows from the division of edges, as sketched in Fig. 2.

The division of faces of tetrahedra is done likewise. However, in this case the subdivision of faces should be done according to a unique rule so that neighbouring elements match. In our method, we base the subdivision of faces in the node numbering, although other criteria are also possible. Once the faces are divided, the division of the element is straightforward. Hence, tetrahedra with three intersection points will be subdivided into four new tetrahedra, see Fig. 3 and tetrahedra with four intersection points will be subdivided into six new tetrahedra, see Fig. 4.

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, the intersection point is moved to that vertex. For instance, if the intersection point I_1 is excessively close to V_3 in Fig. 2, then it is moved to V_3 and the triangle is subdivided into the triangles $V_3I_2V_2$ and $I_2V_1V_2$. A similar procedure is applied for tetrahedra. A tolerance tol_{cc} relative to the mesh size is prescribed by the user to determine whether interface points are too close to the vertices of the elements. The additional error made in this way is of order $\mathcal{O}(h)$, where h denotes the diameter of the triangulation.

It is important to remark that the cut-cell method is only applied to solve the diffusion equation. The interface points obtained with the cut-cell method are used to adapt the triangulation to the interface position, and the level set function is never altered by this procedure. The advantage of the subdivision is that the interface is reconstructed in a relatively simple way, without destroying the original basic mesh. Prescribing boundary conditions on this approximated interface is a straightforward task. Once problem **(P)** is solved, the interface points and interface concentrations are stored in separate vectors, since they will be used to compute the front velocity (5), and the basic mesh is reestablished.

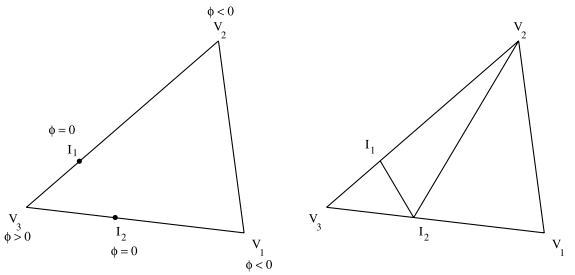

Fig. 2. Subdivision of a two-dimensional element by the cut-cell method.

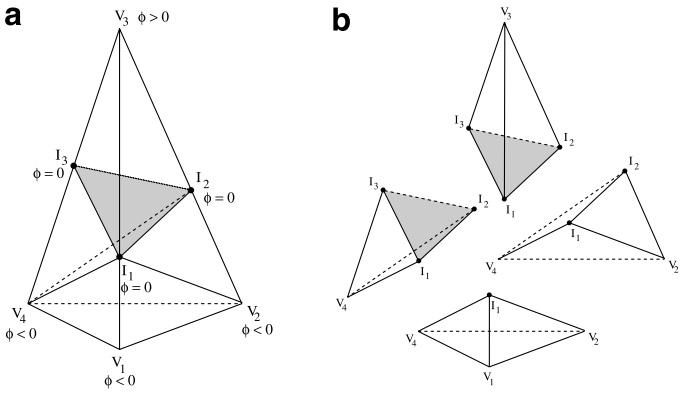

Fig. 3. Subdivision of a three-dimensional element with three intersection points by the cut-cell method. (a) Subdivision of the faces. (b) Subdivision of the element.

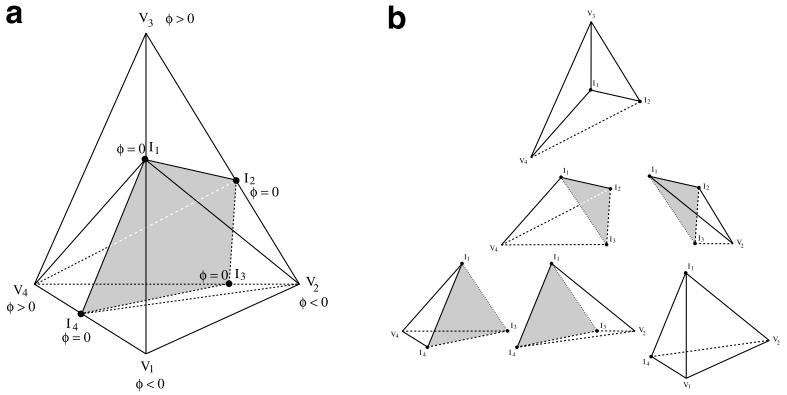

Fig. 4. Subdivision of a three-dimensional element with four intersection points by the cut-cell method. (a) Subdivision of the faces. (b) Subdivision of the element.

3.2.2. Local adaptations of the time integration

Because of the diffusion equations are solved with an implicit Euler method, the time integration at the mesh points that have changed phase within the computational time step must be locally corrected. For the problems considered in this work, this is an important implementational issue due to the jump of the concentrations from c_i^{part} to c_i^{sol} across the interface. The most elegant solution would be to perform the time integration from the time $t^*(\mathbf{x})$ at which the interface passes through the node $\mathbf{x}$ and the new time step t^{n+1}. The time $t^*(\mathbf{x})$ can be easily computed from a linear interpolation of the level set function. Then, one could use

$$\frac{\partial c_i}{\partial t}(\mathbf{x}, t^{n+1}) \approx \frac{c_i(\mathbf{x}, t^{n+1}) - c_i^{\text{sol}}(\mathbf{x}, t^*(\mathbf{x}))}{t^{n+1} - t^*(\mathbf{x})}.$$

This approach can be greatly exploited if $c_i^{\text{sol}}(\mathbf{x}, t^*(\mathbf{x}))$ is known in advance. This holds for instance for long-distance diffusion models of particle dissolution in binary alloys [35], where the interface concentration $c^{\text{sol}}(\mathbf{x}, t)$ is a given constant. However, for the problems under study, the value of $c_i^{\text{sol}}(\mathbf{x}, t^*(\mathbf{x}))$ is unknown, and the only way to compute it is to solve **(P)** at time $t^*(\mathbf{x})$. Since a similar adaptation must be accomplished for all the nodes $\mathbf{x}$ that are crossed by the interface, this technique is unprofitable from a practical point of view. Instead, we control the time-stepping with the tolerance tol_{cc} used in the cut-cell method. Since the diffusion equations are solved with the Euler backward method, the only restriction on the time-stepping is given by the CFL condition (8) for the advection of the interface, which yields

$$\Delta t \leqslant \Delta t_{\text{CFL}} := \frac{\text{CFL}}{\max\left(\frac{|v_1|}{\Delta x_1} + \frac{|v_2|}{\Delta x_2} + \frac{|v_3|}{\Delta x_3}\right)} \tag{14}$$

with $\text{CFL} < 1$. In addition, we impose $\text{CFL} < \text{tol}_{\text{cc}}$. This choice ensures that, before crossing a mesh point, the interface will first lay unavoidably on this mesh point and, consequently, a continuous concentration profile will exist. Normally, in our numerical simulations, we use $\text{tol}_{\text{cc}} = 0.3$ and $\text{CFL} = 0.25$.

3.2.3. The determination of the interface concentrations

The interface concentrations c_i^{sol} are found with an iterative method. The computational cost of such an iteration depends linearly on the number of interface points, which grows linearly in two-dimensional problems and quadratically in three-dimensional problems with the level of refinement of the computational mesh. Hence, the workload per iteration determines the choice of the proper iterative method.

A Newton iteration demands the computation of the Jacobian of $\mathbf{f}$, which in this case should be approximated numerically. Using central differences

$$\frac{\partial f_i}{\partial c_j^{\text{sol}}} = \frac{f_i(c_1^{\text{sol}}, \ldots, c_j^{\text{sol}} + \varepsilon, \ldots, c_p^{\text{sol}}) - f_i(c_1^{\text{sol}}, \ldots, c_j^{\text{sol}} - \varepsilon, \ldots, c_p^{\text{sol}})}{2\varepsilon},$$

with a sufficiently small increment ε, demands $2p^2$ additional evaluations of $\mathbf{f}$. Hence, $2p^2$ problems **(P)** (i.e. $2p^3$ diffusion equations) with modified Dirichlet boundary conditions should be solved in order to approximate the Jacobian in a single interface point and iteration. Broyden's method reduces the computational cost of the Newton method considerably using a sequential approximation of the Jacobian matrix which avoids the additional evaluations of $\mathbf{f}$. However, a $p \times p$ system of equations is still to be solved per interface point and iteration. Therefore, we prefer a fixed-point iteration in spite of its slow convergence rate due to its low computational cost (one evaluation of $\mathbf{f}$ per interface point and iteration).

Let us arrange the chemical species in increasing order of the diffusion coefficients, i.e. $D_1 \leqslant \cdots \leqslant D_p$. Provided $\mathbf{c}_0^{\mathrm{sol}}$, we approximate the zero of $\mathbf{f}$ with a fixed-point iteration

$$\mathbf{c}_k^{\mathrm{sol}} = \mathbf{g}(\mathbf{c}_{k-1}^{\mathrm{sol}}), \tag{15}$$

where the function $\mathbf{g}$ is given by

$$\begin{cases} g_1(\mathbf{c}^{\mathrm{sol}}) = \dfrac{\mathscr{K}(T)}{(c_1^{\mathrm{sol}})^{\tilde{n}_1 - 1}\prod_{j=2}^{p}(c_j^{\mathrm{sol}})^{\tilde{n}_j}}, \\ g_i(\mathbf{c}^{\mathrm{sol}}) = c_i^{\mathrm{sol}} + \delta_i\left(\dfrac{D_i}{c_i^{\mathrm{part}} - c_i^{\mathrm{sol}}}\dfrac{\partial c_i}{\partial \mathbf{n}} - \dfrac{D_{i-1}}{c_{i-1}^{\mathrm{part}} - c_{i-1}^{\mathrm{sol}}}\dfrac{\partial c_{i-1}}{\partial \mathbf{n}}\right), \quad i = 2, \ldots, p. \end{cases} \tag{16}$$

The iteration (15) is terminated when the infinity norm of $\mathbf{f}$ along the interface is smaller than a prescribed tolerance tol_f. The relaxation parameters δ_i are introduced to make $\mathbf{g}$ a contraction and hence to achieve convergence. Furthermore, in order to improve the convergence rate of (15), the parameters δ_i should be tuned according to physical parameters of the chemical species i and $i-1$ involved in the formulation (16). In fact, the choice of $\mathbf{g}$ should be made according to the physical parameters as well. Note that g_i (for $i = 2, \ldots, p$) has two poles at c_i^{part} and c_{i-1}^{part}, and there are situations in which $c_i^{\mathrm{sol}} \to c_i^{\mathrm{part}}$. For instance, if $c_i^{\mathrm{part}} \gg c_i^0 = 0$ for $i = 1, \ldots, p$ and $D_1 \ll D_p$, then $c_1^{\mathrm{sol}} \to c_1^{\mathrm{part}}$ and

$$c_p^{\mathrm{sol}} \to \left(\frac{\mathscr{K}(T)}{\prod_{i=1}^{p-1}(c_i^{\mathrm{sol}})^{\tilde{n}_i}}\right)^{\frac{1}{\tilde{n}_p}} \approx 0$$

in order to satisfy the balance of atoms through the interface (6). In those cases, δ_i should be chosen sufficiently small, which delays convergence considerably. However, the performance of (15) can be substantially improved by defining

$$g_i(\mathbf{c}^{\mathrm{sol}}) = c_i^{\mathrm{sol}} + \delta_i\left((c_{i-1}^{\mathrm{part}} - c_{i-1}^{\mathrm{sol}})D_i\frac{\partial c_i}{\partial \mathbf{n}} - (c_i^{\mathrm{part}} - c_i^{\mathrm{sol}})D_{i-1}\frac{\partial c_{i-1}}{\partial \mathbf{n}}\right). \tag{17}$$

Reversely, in cases where c_i^{sol} deviates sufficiently from c_i^{part}, the terms $c_{[i,i-1]}^{\mathrm{part}} - c_{[i,i-1]}^{\mathrm{sol}}$ in Eq. (17) act as amplification factors, hence δ_i must be reduced and consequently the convergence rate is decreased. In those cases, Eq. (16) defines a more efficient iteration.

The initial guess of the interface concentrations. Providing initial approximations $c_{i,0}^{\mathrm{sol}}$ to the interface concentrations at each time step is a difficult task. At time $t = 0$, we make use of the self-similar solutions [33] defined in unbounded domains for planar interfaces. This approximation does not incorporate the influence of the interface topology, but provides a sufficiently good initial guess. In order to define $c_{i,0}^{\mathrm{sol}}$ at a general time step t^{n+1} we use an average of the interface concentrations at the previous time step, that is

$$c_{i,0}^{\mathrm{sol}}(\mathbf{x}, t^{n+1}) := \frac{1}{n_f}\sum_{k=1}^{n_f} c_i^{\mathrm{sol}}(\mathbf{x}_{f_k}, t^n), \quad \mathbf{x} \in \Gamma(t^{n+1}),\ i = 1, \ldots, p, \tag{18}$$

where n_f denotes the number of interface points $\mathbf{x}_{f_k}$ at time t^n. Note that with this approximation, the spatial variations of $c_i^{\mathrm{sol}}(\cdot, t^n)$ are annihilated. Another possibility, which incorporates the effects of the interface topology, is to define $c_{i,0}^{\mathrm{sol}}(\cdot, t^{n+1})$ from the concentration profiles c_i at the previous time step.

3.3. Extension of the front velocity away of the interface

The front velocity v_n is only defined at the interface location, and an artificial continuous extension of it onto Ω is required to update the level set function. Eqs. (5) impose that all the chemical elements present in the alloy prescribe the same motion to the interface. Hence, v_n will be computed from c_1 and c_1^{sol} without loss of generality. Furthermore, it is important to stress that the interface concentrations, are only known, after solving (**P**), at the interface points computed with the cut-cell method.

Hence, the computational strategy that fits our purposes best is to extend the Cartesian components of the front velocity

$$\mathbf{v}(\mathbf{x},t) = \frac{D_1}{c_1^{\mathrm{part}} - c_1^{\mathrm{sol}}(\mathbf{x},t)} \nabla c_1(\mathbf{x},t)$$

independently in the proper upwind direction. We follow the method presented by Chen et al. [7], where the information is advected from the interface onto Ω by solving in pseudo-time τ the following equations

$$\frac{\partial v_q}{\partial \tau} + S(\phi\phi_{x_q}) \frac{\partial v_q}{\partial x_q} = 0, \quad \text{for } q = 1,2,3, \tag{19}$$

where $\mathbf{v} = (v_1, v_2, v_3)^t$, $\mathbf{x} = (x_1, x_2, x_3)^t$ and S denotes the sharp sign function. The extension procedure has been sketched in Fig. 5 for a two-dimensional example. First, we compute the velocity at the interface points using linear approximations of $\frac{\partial c_1}{\partial x_q}$. Subsequently, the front velocity is attached to the east/west (resp. north/south) neighbours of the interface points. And finally, the information is advected from these points in the proper direction with Eq. (19). In order to reduce the computational cost invested in this procedure, a fixed number of pseudo-time iterations of (19), namely 10, are carried out to extend the front velocity into a sufficiently wide band around the interface.

3.4. The complete algorithm

Next, we give the complete algorithm of our method, where the coupling between the above discussed issues is presented.

1. Set $t = 0$. Initialize the level set function as a signed distance function according to the initial particle geometry, being positive inside the diffusive phase.

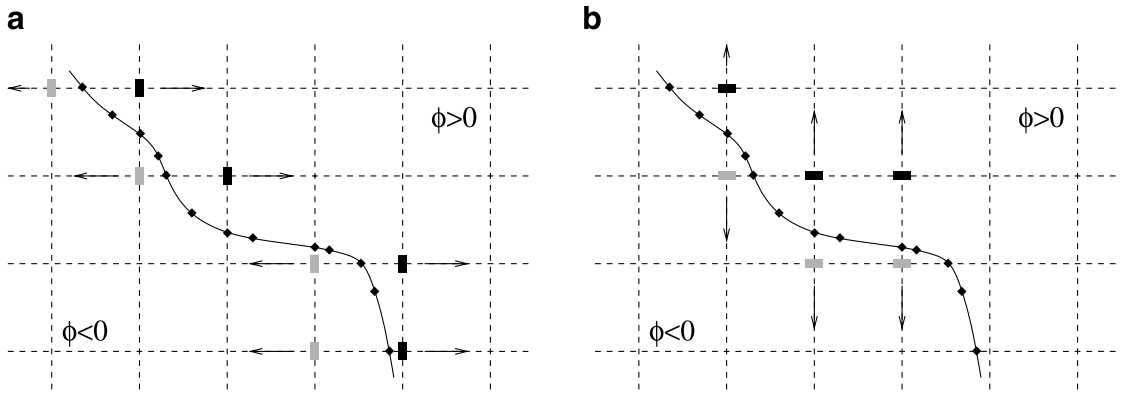

Fig. 5. Extension of the front velocity in the Cartesian directions: neighbours inside the diffusive phase are marked with black rectangles, neighbours inside the particle are marked with grey rectangles, and interface points computed with the cut-cell method are marked with ♦. Information travels off the interface. (a) Extension of the front velocity in the direction of x_1. (b) Extension of the front velocity in the direction of x_2.

2. Initialize the interface concentrations c_i^{sol} and concentrations c_i inside the diffusive phase. For complex particle geometries, for which self-similar solutions [33] can not be defined, use the cut-cell method to adapt the background triangulation to the interface position and solve **(P)** with a time-step $\Delta t_0 \propto \Delta \bar{x}^2$.
3. Compute the velocity field $\mathbf{v}$ according to the procedure presented in Section 3.3, and determine the time step Δt in agreement with (14).
4. Update the level set function by solving Eq. (7).
5. If necessary, solve Eq. (10) to reinitialize the level set function to a distance function.
6. Use the cut-cell method to adapt the background triangulation to the interface position at the new time step.
7. Update the interface concentrations c_i^{sol} and concentrations c_i inside the diffusive phase by solving **(P)**. Perform following iteration:
   7.1 Define the initial approximation $c_{i,0}^{\mathrm{sol}}$ of the interface concentrations with (18). Set $k = 0$.
   7.2 Solve the diffusion equations (1) with Dirichlet boundary conditions $c_i^{\mathrm{sol}} = c_{i,k}^{\mathrm{sol}}$ on the interface and Neumann boundary conditions (4) on the boundary of the diffusive phase not being the interface.
   7.3 Compute $\mathbf{f}$ from the concentration profiles $c_{i,k}$ obtained in 7.2.
   7.4 If the infinity norm of $\mathbf{f}$ along the interface points is larger than tol_f, then update the interface concentrations with (15), set $k = k + 1$ and go back to 7.2.Otherwise, set $c_i^{\mathrm{sol}}(\mathbf{x}, t + \Delta t) = c_{i,k}^{\mathrm{sol}}(\mathbf{x}, t + \Delta t)$, $c_i(\mathbf{x}, t + \Delta t) = c_{i,k}(\mathbf{x}, t + \Delta t)$ and go to 8.
8. Update the time step: $t = t + \Delta t$.
9. If the termination time t_{end} is not reached, go back to 3.

4. Numerical results

This section contains some numerical experiments. First, we consider an academic test problem to study the accuracy of our method. Subsequently, the model is applied to an AlMgSi-alloy to investigate the influence of the particle morphology in the dissolution kinetics. Finally, a topological change is simulated by means of the dissolution of a three-dimensional dumbbell-like particle.

4.1. Accuracy of the numerical method

The influence of the numerical method on the accuracy of the results is investigated here by means of an academic test problem. To this end, consider the dissolution of a particle in an alloy with one, two, three and four diffusive chemical species, respectively. The physical parameters associated with these alloys are listed in Table 1. Note that in the binary alloy (i.e. only one diffusive element), the interfacial concentration is given, whereas in the multi-component alloys, the interfacial concentrations are computed as part of the solution. The performance of our method will be examined by comparison of the numerical results with self-similar solutions and steady-state solutions.

4.1.1. Comparison with self-similar solutions

Self-similar solutions of vector Stefan problems are well known for simple particle geometries. Bourne et al. [4] describe the shape-preserving growth in infinite media of planar, cylindrical, spherical and ellipsoidal particles in ternary alloys. More recently, Vermolen [36] extended these solutions to particle growth in

Table 1
Physical parameters of the alloy with one, two, three and four chemical species, respectively

One species	$c^{\mathrm{part}} = 5$	$c^0 = 0$	$c^{\mathrm{sol}} = 1$	$D = 1$
Two species	$c_i^{\mathrm{part}} = 5$	$c_i^0 = 0$	$\tilde{n}_i = 1$, $\mathscr{K}(T) = 1$	$D = 1$, $D_2 = 5$
Three species	$c_i^{\mathrm{part}} = 5$	$c_i^0 = 0$	$\tilde{n}_i = 1$, $\mathscr{K}(T) = 1$	$D = 1$, $D_2 = 5$, $D_3 = 0.5$
Four species	$c_i^{\mathrm{part}} = 5$	$c_i^0 = 0$	$\tilde{n}_i = 1$, $\mathscr{K}(T) = 1$	$D = 1$, $D_2 = 5$, $D_3 = 0.5$, $D_4 = 20$

multi-component alloys with interface reactions. Analogous solutions for particle dissolution are not always available, since in many cases it is assumed that precipitates are absent at the initial time. However, in this work, we will use the self-similar solutions for dissolution of planar and spherical particles given by Vermolen et al. [33] and Whelan [42], respectively. For planar particles, the interface position s is given by

$$s(t) = s_0 + \alpha\sqrt{t}, \tag{20}$$

where α is related to the interfacial concentrations c_i^{sol} by

$$\frac{\alpha}{2} = \frac{c_i^0 - c_i^{\mathrm{sol}}}{c_i^{\mathrm{part}} - c_i^{\mathrm{sol}}}\sqrt{\frac{D_i}{\pi}}\frac{\exp\left(-\frac{\alpha^2}{4D_i}\right)}{\operatorname{erfc}\left(\frac{\alpha}{2\sqrt{D_i}}\right)}, \quad i = 1, \ldots, p. \tag{21}$$

For spherical particles, the front velocity $\frac{\mathrm{d}s}{\mathrm{d}t}$ is given by

$$\frac{\mathrm{d}s}{\mathrm{d}t}(t) = -\frac{c_i^{\mathrm{sol}} - c_i^0}{c_i^{\mathrm{part}} - c_i^{\mathrm{sol}}}\left(\frac{D_i}{s(t)} + \sqrt{\frac{D_i}{\pi t}}\right), \quad i = 1, \ldots, p. \tag{22}$$

Remember that in both cases, the interfacial concentrations c_i^{sol} satisfy the solubility product Eq. (3). Moreover, it is worth noting the time dependence of the interfacial concentrations in the case of spherical symmetry, which are constant for planar symmetry. This gives evidence of the importance of particle morphology in the dissolution kinetics.

The evolution of the interface is presented in Fig. 6 together with the self-similar solutions for both particle geometries. In both cases, the numerical solutions agree well with the analytic solutions at the begining of the dissolution process. However, as time evolves, the numerical solutions separate from the analytic ones due to the boundedness of our domain domain of computation ($\Omega = [0, 1]$), and converge to the equilibrum states as will be shown in the next section. Similar results are obtained for the systems with three and four diffusive species.

4.1.2. Comparison with steady-state solutions

In this section, we solve the dissolution of a planar interface in a two-dimensional framework in order to establish the influence of the cut-cell method in the accuracy of the numerical results when vector Stefan problems are to be solved in higher dimensions. To this effect, consider a computational domain $\Omega = [0, 1]^2$, and assume that the interface is initially located at $\Gamma(0) = \{\mathbf{x} \in \Omega | x_1 = s_0\}$ with $s_0 = 0.615$. Furthermore, assume that the particle is located at the left side of the interface and the diffusive phase at its right. The parameters of the problem, given in Table 1, have been chosen in such a way that the position of the interface s_∞ at the steady-state solution is the same for all cases under study. For general geometries, the equilibrium interface concentrations $c_{i,\infty}^{\mathrm{sol}}$ and equilibrium particle area (i.e. volume) V_∞ are the solution of system (**E**) based on a conservation of mass argument and the hyperbolic Eq. (3)

$$\left|\begin{array}{l} c_i^{\mathrm{part}} V_0 + c_i^0(V_\Omega - V_0) = c_i^{\mathrm{part}} V_\infty + c_{i,\infty}^{\mathrm{sol}}(V_\Omega - V_\infty), \quad i = 1, \ldots, p, \\ \prod_{i=1}^{p} (c_{i,\infty}^{\mathrm{sol}})^{\tilde{n}_i} = \mathscr{K}(T), \end{array}\right. \tag{E}$$

where V_0 and V_Ω denote the area (i.e. volume) of the particle at $t = 0$ and of the computational domain Ω, respectively. The equilibrium position of the interface s_∞ is derived from the area of the particle at the equilibrium. In this case, $s_\infty = V_\infty = 0.51875$.

Next, we present the relative error in the interface position $\frac{|s_\infty - s_h(t)|}{s_\infty}$ at $t = 0.75$ for the various alloys. First, we present in Table 2 the impact of the cut-cell method in the accuracy of the results for the dissolution in the binary alloy. In this case, the interface concentration is given and the iteration (15) is not necessary to solve (**P**), which allows us to use a small value of $\mathrm{tol}_{\mathrm{cc}}$. This choice might permit narrow elements on the interface, which for dissolution problems in multi-component alloys result in poor approximations of the fluxes $\frac{\partial c_i}{\partial \mathbf{n}}$ and the divergence of (15) in the worse cases. Thus, $\mathrm{tol}_{\mathrm{cc}} = 0.01$ and $\mathrm{tol}_{\mathrm{cc}} = 0.3$ are used as a default tolerances for the dissolution in binary and multi-component alloys, respectively. Surprisingly, a larger difference in

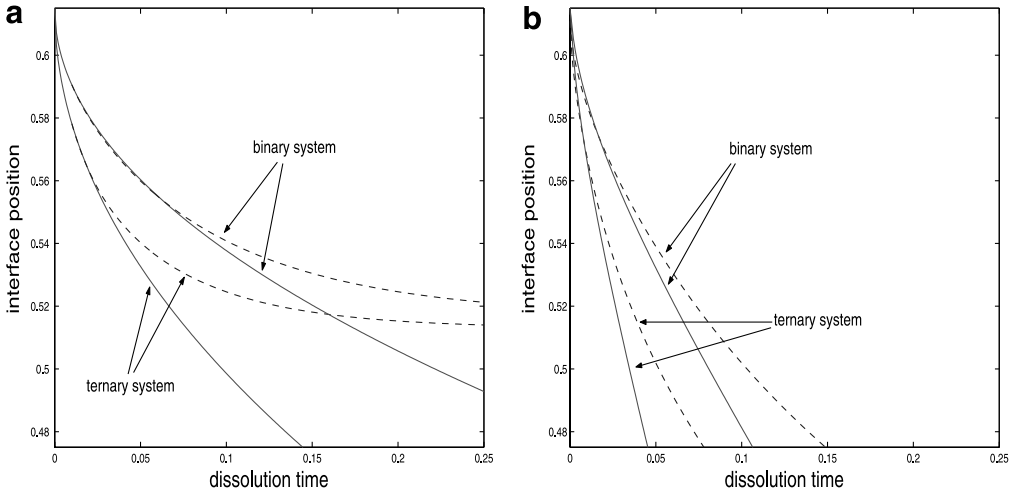

Fig. 6. Early stages of dissolution. Numerical solution (dashed curves) vs the self-similar solution (solid curves) for: (a) planar symmetry; and (b) spherical symmetry.

Table 2
Relative error in the interface position at $t = 0.75$ for the problem with only one diffusive species and the effect of $\mathrm{tol}_{\mathrm{cc}}$ on the experimental order of convergence

Mesh size	Error	EOC	Error	EOC
6.25×10^{-2}	3.46×10^{-2}		4.04×10^{-2}	
3.13×10^{-2}	1.80×10^{-2}	0.95	1.81×10^{-2}	1.16
1.56×10^{-2}	9.22×10^{-3}	0.96	9.09×10^{-3}	0.99
7.81×10^{-3}	4.69×10^{-3}	0.98	5.39×10^{-3}	0.75
	$\mathrm{tol}_{\mathrm{cc}} = 0.01$		$\mathrm{tol}_{\mathrm{cc}} = 0.3$	
α	-0.2444			

the accuracy is observed at the finest mesh. The error due to the shift of the interface points to mesh points is proportional to the mesh size Δx_k, so it does not destroy the overall accuracy of the numerical method. However, as the mesh is refined, this shift is carried out in more occasions for the same displacement of the interface. Hence, the accumulative effect of these errors is what probably causes the differences in the accuracy. Unfortunately, these slight differences in the accuracy have a stronger effect in the experimental order of convergence (EOC)

$$\mathrm{EOC}_{\Delta x} := \frac{\ln\left(\frac{\mathrm{error}_{\Delta x}}{\mathrm{error}_{\Delta x/2}}\right)}{\ln(2)}.$$

The relative errors in the interface position at $t = 0.75$ for the multi-component alloys are presented in Table 3. The coefficient α presented in the last row of Tables 2 and 3 gives a measure of the dissolution rate derived from the self-similar solutions (see Eq. (20)). The numerical results show that the accuracy depends on the dissolution rate. It appears that the accuracy of the method is reduced by increasing the speed of the dissolution.

4.2. The effect of the particle morphology in the dissolution kinetics

In this section, we study the influence of the morphology of the particle in the dissolution kinetics. We consider a family of Mg_2Si particles whose shapes vary from a circle (i.e. cylindrical symmetry) towards a very elongated ellipse. The physical parameters of the problems are shown in Table 4. The diffusion coefficients D_{Mg} and D_{Si} and the solubility product $\mathscr{K}(T)$ are given by the following Arrhenius relationships

Table 3
Relative error in the interface position at $t = 0.75$ for the problems with two, three and four diffusive species

Mesh size	Two species	Three species	Four species
6.25×10^{-2}	4.41×10^{-2}	3.79×10^{-2}	5.15×10^{-2}
3.13×10^{-2}	2.84×10^{-2}	2.05×10^{-2}	3.62×10^{-2}
1.56×10^{-2}	1.71×10^{-2}	1.36×10^{-2}	2.57×10^{-2}
7.81×10^{-3}	1.13×10^{-2}	8.39×10^{-3}	2.02×10^{-2}
α	-0.3682	-0.2876	-0.4081

Table 4
Physical parameters for the dissolution of the Mg_2Si particles

$c^{\text{part}}_{\text{Mg}} = 65$ wt%	$D^0_{\text{Mg}} = 4.9 \times 10^7$ μm^2/s	$n_{\text{Mg}} = 2/3$
$c^{\text{part}}_{\text{Si}} = 35$ wt%	$D^0_{\text{Si}} = 2.02 \times 10^8$ μm^2/s	$n_{\text{Si}} = 1/3$
$c^0_{\text{Mg}} = 0$ wt%	$Q^d_{\text{Mg}} = 1.24 \times 10^5$ J/mol	$k_0 = 89$
$c^0_{\text{Si}} = 0.038$ wt%	$Q^d_{\text{Si}} = 1.36 \times 10^5$ J/mol	$Q_{\text{s}} = 3.197 \times 10^4$ J/mol

$$D_{[\text{Mg,Si}]} = D^0_{[\text{Mg,Si}]} \exp\left(-\frac{Q^d_{[\text{Mg,Si}]}}{RT}\right), \quad \mathcal{K}(T) = k_0 \exp\left(-\frac{Q_{\text{s}}}{RT}\right),$$

where $R = 8.3144$ J/K mol denotes the universal gas constant and $T = 560$ °C stands for the temperature at which the phase transformation takes place.

In order to compare the dissolution kinetics of the different particles, we fix the volume fraction of the initial particle. To achieve this, all the simulations are carried out over a squared cell of area 225 μm^2 and the area of the initial particle is set equal to π μm^2. This corresponds with a circular particle of radius 1 μm (curve I in the figures) or an ellipse which its semimajor and semiminor axes a and b satisfy $ab = 1$ μm^2. In our numerical experiments, we have used $b = 0.8, 0.6, 0.4, 0.2$ μm (curves II, III, IV and V, respectively, in the figures).

Next, the normalized area $A(t)/A(0)$ of the Mg_2Si particles is presented in Fig. 7(a) as a function of the normalized dissolution time t/t_{diss}, where t_{diss} denotes the dissolution time of the circular particle. From our numerical calculations we have that $t_{\text{diss}} = 186.04$ s. As can be observed from the results, the geometry of the particle strongly influences the dissolution kinetics. The more elongated the ellipse is, the faster dissolution takes place. This is due to the fact that, the more elongated the ellipse is, the larger the arc length of the

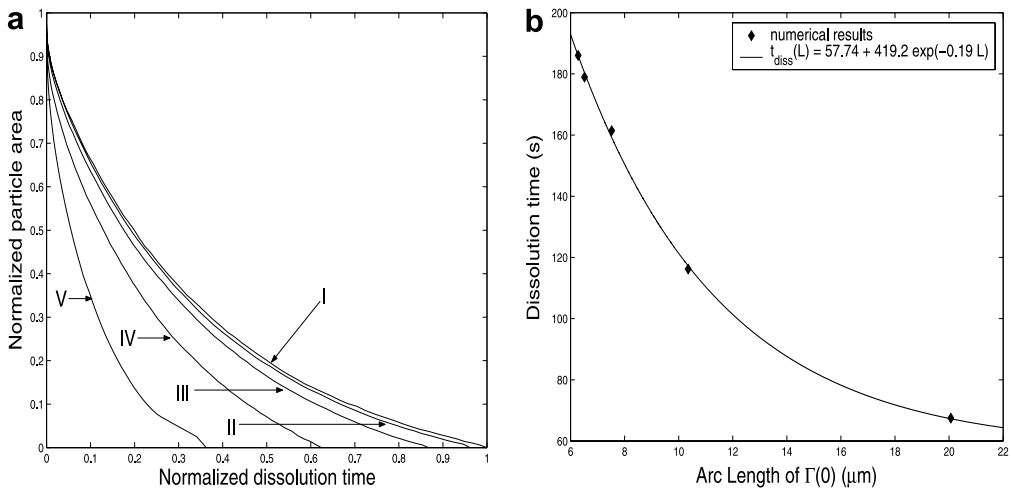

Fig. 7. Dissolution of the Mg_2Si particles: (a) normalized particle area vs normalized dissolution time; (b) dissolution time vs arc length of the initial interface.

interface is, and consequently more atoms are able to cross the interface at the same time. In fact, the numerical results indicate that the dissolution time decays exponentially with the perimeter of the initial particle, as can be observed in Fig. 7(b). The function $t_{\text{diss}}(L)$ approximating the dissolution time as a function of the arc length of the initial interface has been fitted to the numerical results with a least squares method.

Subsequently, we present in Fig. 8 the eccentricity

$$e = \sqrt{1 - \frac{b^2}{a^2}}$$

of the particle during the dissolution process. The numerical results show that the shape of the initial particle is preserved during most of the dissolution process. The largest variations are observed in curves II, III and IV, and occur when the normalized particle area is less than 2%, which corresponds with the area of about 6 or 7 control volumes of our reference mesh. These variations indicate that the particle tends to a more rounded shape when it is close to be fully dissolved. However, since the mesh is not locally refined when the particle is about to dissolve, this outcome might be due to or affected by numerical artifacts.

4.3. Dissolution of a three-dimensional particle involving a topological change

In this section, the dissolution of a three-dimensional dumbbell-like particle in a ternary alloy is simulated. The initial geometry of the particle is given by two spheres, whose centers and radius are $\mathbf{c}_{\text{l}} = (-0.275, 0, 0)^t$, $\mathbf{c}_{\text{r}} = (0.21, 0, 0)^t$, $r_{\text{l}} = 0.15$ and $r_{\text{r}} = 0.2$, connected by a cylindrical bar of radius $r_{\text{bar}} = 0.05$. The domain of computation $\Omega = [-0.5, 0.5]^3$ and the physical parameters of the problem, listed in Table 5, have been chosen to obtain the full dissolution of the particle. In spite of that this is an academic test problem, similar ratios of the diffusion coefficients can be found in the 6xxx-Al alloys under homogenization conditions. The influence of the particle concentrations, which may be notably different in the 6xxx series, in the performance of the method is less relevant because their effect on the performance of the iteration procedure (15) is minor.

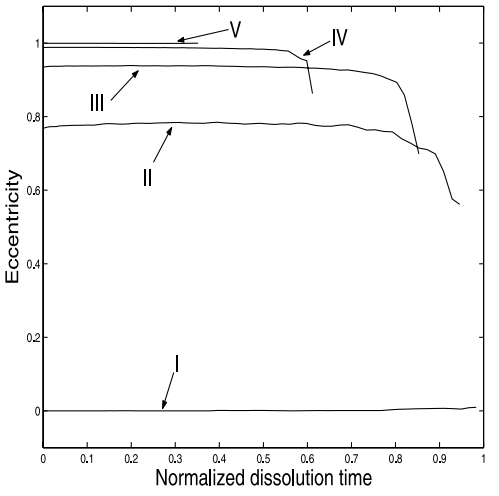

Fig. 8. Dissolution of the Mg_2Si particles: eccentricity of the particle vs normalized dissolution time.

Table 5
Physical parameters for the dissolution of the 3D dumbbell-shaped particle

$c_1^{\text{part}} = 5$	$c_1^0 = 0$	$D_1 = 1$	$n_1 = 1$	$\mathscr{K}(T) = 1$
$c_2^{\text{part}} = 5$	$c_2^0 = 0$	$D_2 = 2$	$n_2 = 1$	

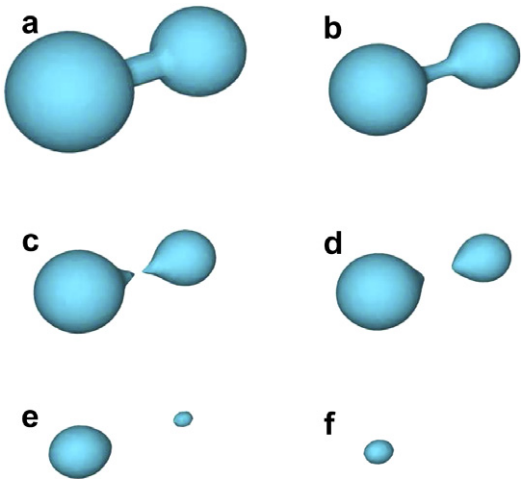

Fig. 9. Dissolution of a three-dimensional dumbbell-like particle in a ternary alloy: (a) $t = 0$; (b) $t = 2.9 \times 10^{-3}$; (c) $t = 7.4 \times 10^{-3}$; (d) $t = 1.05 \times 10^{-2}$; (e) $t = 2.85 \times 10^{-2}$; (f) $t = 4.82 \times 10^{-2}$.

The evolution of the interface is presented in Fig. 9. During the dissolution process, the particle breaks up into two separate particles which dissolve separately. Fig. 9(c) shows clearly the cusps in the interface just after the particle splits, which are smoothed out rapidly as the dissolution evolves. Furthermore, due to the initial topology of the particle, the sub-particle on the right dissolves, as expected, faster.

5. Conclusions

A novel three-dimensional sharp interface model based on the level set method is proposed for simulation of particle dissolution in multi-component alloys. A cut-cell method is presented to adapt the reference mesh to the interface each time step. Due to the complexity of the problem, we recommend a Picard iteration to solve the non-linear coupling on the concentrations of the various alloying elements at the interface. The numerical method is shown to be capable of dealing with complex geometries and topological changes. Perfect agreement with self-similar solutions and steady-state solutions is obtained. Furthermore, it is demonstrated experimentally that the morphology of the particle exerts a strong influence on the dissolution kinetics. Our numerical results indicate an exponential acceleration of the dissolution process with the circumference of the initial particle.

Of course the method is, at this early stage of development, far to be optimized. A noticeable slow down of the convergence of the non-linear iteration is observed for problems in which the diffusion coefficients differ of several, from 4 to 6, orders of magnitude. We believe that a significant improvement in the computational cost can be achieved by optimizing the non-linear solver. Furthermore, even though this method does not have any restriction on the maximum number of species that can be considered simultaneously and works satisfactorily for a wide range of industrial alloys, it also opens the door to many new questions. Further, a close comparison with experimental results currently not available is necessary to successfully comprehend the metallurgical implications of these transformations.

Acknowledgments

The authors thank Dr. Lie Zhao and one of the referees for their remarks, which improved the manuscript. Financial support for this research was provided by the Dutch Technology Foundation (STW).

References

[1] H.B. Aaron, G.R. Kotler, Second phase dissolution, Met. Trans. 2 (1971) 393–407.
[2] D. Adalsteinsson, J.A. Sethian, The fast construction of extension velocities in level set methods, J. Comp. Phys. 148 (1999) 2–22.

[3] U.L. Baty, R.A. Tanzilli, R.W. Heckel, Dissolution kinetics of CuAl2 in Al–4Cu alloy, Met. Trans. 1 (1970) 1651–1656.
[4] J.P. Bourne, C. Atkinson, R.C. Reed, Diffusion-controlled growth in ternary systems, Metall. Mater. Trans. 25A (1994) 2683–2694.
[5] E. Burman, M. Picasso, J. Rappaz, Analysis and computation of dendritic growth in binary alloys using a phase-field model, Numerical Mathematics and Advanced Applications, Springer, Berlin, 2004.
[6] G. Caginalp, Stefan and Hele-Shaw type models as asymptotics of the phase-field equations, Phys. Rev. A 39 (1989) 5887–5896.
[7] S. Chen, B. Merriman, S. Osher, P. Smereka, A simple level set method for solving Stefan problems, J. Comp. Phys. 135 (1997) 8–29.
[8] J. Crank, Free and Moving Boundary Problems, Oxford Science Publications, The Clarendon Press, New York, 1984.
[9] D. Dumont, A. Deschamps, Y. Brechet, A model for predicting fracture mode and toughness in 7000 series aluminium alloys, Acta Mater. 52 (2004) 2529–2540.
[10] M. Dumont, A. Steuwer, A. Deschamps, M. Peel, P.J. Withers, Microstructure mapping in friction stir welds of 7449 aluminium alloy using SAXS, Acta Mater. 54 (2006) 4793–4801.
[11] C.M. Elliott, J.R. Ockendon, Weak and Variational Methods for Moving Boundary Problems, Pitman Press, Boston, 1982.
[12] G. Ghosh, G.B. Olson, Simulation of paraequilibrium growth in multicomponent systems, Metall. Mater. Trans. 32A (2001) 455–467.
[13] U. Grafe, B. Böttger, J. Tiaden, S.G. Fries, Coupling of multicomponent thermodynamics to a phase field model: application to solidification and solid-state transformations of superalloys, Scripta Mater. 42 (2000) 1179–1186.
[14] M. Hillert, J. Ågren, On the definitions of paraequilibrium and orthoequilibrium, Scripta Mater. 50 (2004) 697–699.
[15] G.-S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (2000) 2126–2143.
[16] D. Juric, G. Tryggvason, A front-tracking method for dendritic solidification, J. Comp. Phys. 123 (1996) 127–148.
[17] Y.-T. Kim, N. Goldenfeld, J. Dantzig, Computation of dentritic microstructures using a level set method, Phys. Rev. E 62 (2000) 2471–2474.
[18] I. Kovačević, B. Šarler, Solid–solid phase transformations in aluminiun alloys described by a multiphase-field model, Mater. Sci. Forum 508 (2006) 579–584.
[19] Y.C. Lam, J.C. Chai, P. Rath, H. Zheng, V.M. Murukeshan, A fixed-grid method for chemical etching, Int. Commun. Heat Mass Transf. 31 (2004) 1123–1131.
[20] 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.
[21] W.D. Murray, F. Landis, Numerical and machine solutions of transient heat- conduction problems involving melting or freezing, Trans. ASME (C) 81 (1959) 106–112.
[22] B. Nedjar, An enthalpy-based finite element method for nonlinear heat problems involving phase change, Comput. Struct. 80 (2002) 9–21.
[23] F.V. Nolfi Jr., P.G. Shewmon, J.S. Foster, The dissolution and growth kinetics of spherical particles, Trans. Metall. Soc. AIME 245 (1969) 1427–1433.
[24] S. Osher, R. Fedkiw, Level set methods and dynamic implicit surfaces, Applied Mathematical Sciences, vol. 153, Springer-Verlag, New York, 2003.
[25] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comp. Phys. 79 (1988) 12–49.
[26] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comp. Phys. 155 (1999) 410–438.
[27] O. Reiso, N. Ryum, J. Strid, Melting and dissolution of secondary phase particles in AlMgSi-alloys, Metall. Trans. 24A (1993) 2629–2641.
[28] 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.
[29] J.A. Sethian, Level set methods and fast marching methods, Cambridge Monographs on Applied and Computational Mathematics, second ed., vol. 3, Cambridge University Press, Cambridge, 1999.
[30] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comp. Phys. 114 (1994) 146–159.
[31] K. Thornton, J. Ågren, P.W. Voorhees, Modelling the evolution of phase boundaries in solids at the meso- and nano-scales, Acta Mater. 51 (2003) 5675–5710.
[32] U.H. Tundal, N. Ryum, Dissolution of particles in binary alloys: Part I, Computer simulations, Met. Trans. 23A (1992) 433–449.
[33] F.J. Vermolen, C. Vuik, A mathematical model for the dissolution of particles in multi-component alloys, J. Comp. Appl. Math. 126 (2000) 233–254.
[34] F.J. Vermolen, C. Vuik, Solution of vector Stefan problems with cross-diffusion, J. Comput. Appl. Math. 176 (2005) 179–201.
[35] F.J. Vermolen, C. Vuik, E. Javierre, S. van der Zwaag, Review on some Stefan problems for particle dissolution in solid metallic alloys, Nonlinear Anal.: Modell. Control 10 (2005) 257–292.
[36] F.J. Vermolen, On self-similar solutions and interface reactions for a vector-valued Stefan problem, Nonlinear Anal.: Modell. Control (submitted for publication).
[37] F.J. Vermolen, C. Vuik, S. van der Zwaag, The dissolution of a stoichiometric second phase in ternary alloys: a numerical analysis, Mater. Sci. Eng. A 246 (1998) 93–103.
[38] F.J. Vermolen, C. Vuik, S. van der Zwaag, Particle dissolution and cross-diffusion in multi-component alloys, Mater. Sci. Eng. A 347 (2003) 265–279.
[39] V.R. Voller, An implicit enthalpy solution for phase change problems: with application to a binary alloy solidification, Appl. Math. Modell. 11 (1987) 110–116.

[40] V.R. Voller, M. Cross, Accurate solutions of moving boundary problems using the enthalpy method, Int. J. Heat Mass Transf. 24 (1981) 545–556.
[41] A.A. Wheeler, W.J. Boettinger, G.B. McFadden, Phase-field model for isothermal phase transitions in binary alloys, Phys. Rev. A 45 (1992) 7424–7439.
[42] M.J. Whelan, On the kinetics of particle dissolution, Met. Sci. J. 3 (1969) 95–97.
[43] H.K. Zhao, A fast sweeping method for Eikonal equations, Math. Comp. 74 (2004) 603–627.