

Numerical methods for vector Stefan models  
of solid-state alloys

PROEFSCHRIFT

ter verkrijging van de graad van doctor  
aan de Technische Universiteit Delft,  
op gezag van de Rector Magnificus prof.dr.ir. J.T. Fokkema,  
voorzitter van het College voor Promoties,  
in het openbaar te verdedigen op dinsdag 21 november 2006 om 10:00 uur

door

Etelvina JAVIERRE PÉREZ  
Licenciada en Matemáticas, Universidad de Zaragoza

geboren te Almudafar, provincie Huesca, Spanje

Dit proefschrift is goedgekeurd door de promotor:

Prof.dr.ir. P. Wesseling

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof.dr.ir. P. Wesseling,	Technische Universiteit Delft, promotor
Dr.ir. C. Vuik,	Technische Universiteit Delft, toegevoegd promotor
Prof.dr.ir. A.W. Heemink,	Technische Universiteit Delft
Prof.dr.ir. J.J.W. van der Vegt,	Universiteit Twente
Prof.dr.ir. S. van der Zwaag,	Technische Universiteit Delft
Dr.rer.nat. S. Luding,	Technische Universiteit Delft
Dr.ir. F.J. Vermolen,	Technische Universiteit Delft

Numerical methods for vector Stefan models of solid-state alloys.

Dissertation at Delft University of Technology.

Copyright © 2006 by E. Javierre Pérez



The work described in this thesis was financially supported by STW.

ISBN 90-8559-240-2

---

# Summary

Aluminium is a widely used metal in various branches of industry. Packing, construction, aerospace and automotive industries are some of its applications. The molten aluminium manufactured from bauxite, by means of sequential application of the Bayer and Hall-Heroult processes, is enriched with other chemical elements (magnesium, manganese, silicon and iron among others) to give the so-called aluminium-based alloys. Subsequently, the obtained alloy is cast into a mould where it solidifies. During the solidification process, segregation of the alloying elements takes place and small precipitates (*i.e.* particles), that are rich in alloying elements, are formed. These precipitates result in local variations of the mechanical properties, which limit the applicability and efficiency of further mechanical treatments. Thus, for instance, direct extrusion after solidification leads to the appearance of cracks at the surface of the extruded material and/or an orange-like skin as an undesired effect. In order to remove these compositional variations and hence to improve its mechanical properties, an intermediate annealing treatment, referred to as homogenization, is applied prior to extrusion. In this thermal treatment the alloy is heated up to just below its eutectic temperature and kept at this temperature for several hours. As a result, a more homogeneous micro-structure is obtained, due to diffusional transport of the alloying elements. Some of the metallurgical processes of major importance that can take place during homogenization are:

1. dissolution of precipitates being present as isolated particles,
2. morphological changes of the precipitates, in particular the transformation of metastable  $\beta$ -particles of plate-like structure into arrays of stable  $\alpha$ -particles with sphere-like structure,
3. nucleation and growth of precipitates formed from alloying elements that are in a supersaturated solution.

Homogenization is a very expensive process because of its energy requirements. An underestimate of the homogenization time results into bad properties or failure during the manufacturing process leading to extremely high cost, whereas an overestimate of this time leads to high energy cost. Therefore, predictions of homogenization times are of great value. Computer simulations can provide fast, cheap and valuable insight into the homogenization process.

This work is focused on dissolution processes, although nucleation and growth processes are also briefly discussed in Chapter 7. The dissolution kinetics are determined by many factors, such as morphology, effective diffusion coefficients and thermodynamic boundary conditions at the particle/matrix interface. The mathematical model used to describe these processes is of the vector Stefan type: diffusion equations describing the transport of the alloying elements have to be solved in a domain, the aluminium-rich phase, which changes in time. A mass conservation argument leads to an equation of motion for the moving interface separating the particle and

the aluminium-rich matrix. Furthermore, the balance of atoms of each alloying element crossing the interface yields a strong nonlinear coupling between the concentration gradients and the interface concentrations on the interface. Our goal is to develop a robust numerical method to perform computer simulations of particle dissolution, which may or may not involve morphological changes, in one, two or three spatial dimensions. Furthermore, even though this work is focused on aluminium-based alloys, most of the mathematical insights can also be used in other metallic alloys, and even in the field of ceramic materials.

The first question that arises when dealing with a moving boundary problem is how to represent the moving interface. Front-capturing methods have shown to be the most adequate to solve moving boundary problems which involve topological changes. Among these, phase field and level set methods are the most widely used within the scientific community to simulate phase transformations. Phase field methods introduce a diffusive interface, where the phase transformation occur, and avoid direct implementation of the interface conditions. A great difficulty of phase field methods is, in the opinion of the author, that the accurate resolution of the interface region demands adaptive mesh techniques, which result in severe bounds on the time step and hence in large computational cost. Thus, a level set method has been chosen to describe the moving interface. The initial front coincides with the zero level set of a signed distance function, and its motion is rewritten as an advection equation for the level set function, whose velocity field is a continuous extension of the front velocity. Furthermore, the level set function is re-initialized to a signed distance function whenever necessary, to prevent flat/sharp gradients near the interface. With this formulation, the computational domain is parameterized by the level set function, and the moving interface is treated as a 'sharp' interface.

A simple combination of Finite Difference and Finite Element methods is presented in this work. Hyperbolic equations (*i.e.* advection of the interface and re-initialization of the level set function) inherited from the level set formulation are solved on a Cartesian grid with Finite Difference schemes. The continuous extension of the front velocity to the whole computational domain is achieved by advecting information from the interface in the proper upwind direction. The diffusion problems are solved with the SEPRAN package, which uses the Finite Element method. A novel technique, referred to as the cut-cell method, provides a linear reconstruction of the moving interface and realizes the adaptation of the background Finite Element mesh to the interface position. Thus, implementation of the interface conditions becomes a simple task. Finally, an iterative algorithm is implemented in order to deal with the nonlinear coupling on the interface for the vector-valued Stefan problem which occurs in multi-component alloys. Such an iteration should be carried out along all the interface points, whose number grows linearly in two-dimensional problems and quadratically in three-dimensional problems with the level of refinement of the computational mesh. This motivates us to implement a fixed-point iteration due to its reduced computational cost per iteration in spite of its slow convergence rate.

The innovative aspects of this work reside in the application of the level set method to vector Stefan problems, in the simulation of dissolution of three-dimensional precipitates, involving topological changes of their geometry, and in the FD/FE combination which leads to the implementation of the cut-cell method. The accuracy of the numerical results is checked by comparison with self-similarity or steady-state solutions. A survey on the computational cost of the numerical algorithm is provided. Both single particle dissolution and morphological changes, mainly induced by cracks on the surface of the particle, are simulated.



---

# Samenvatting van het proefschrift getiteld ”Numerieke methoden voor vector Stefan modellen van legeringen in vaste toestand”

Aluminium is een veel gebruikt metaal in verschillende takken van de industrie. Als voorbeelden noemen we de verpakkingsindustrie, bouw, luchtvaart- en automobielandustrie. Voor het verkrijgen van aluminiumlegeringen wordt gesmolten aluminium, dat vervaardigd wordt uit bauxiet door toepassing van achtereenvolgens de Bayer- en Hall-Heroult processen, verrijkt met legeringselementen (zoals magnesium, mangaan, silicium en ijzer). Vervolgens wordt de legering gegoten in een mal waar hij stolt. Tijdens het stollen komen legeringselementen vrij, die in oververzadigde toestand in oplossing zaten, om kleine precipitaten (ofwel deeltjes) te vormen. Deze deeltjes zijn rijk aan legeringselementen. Deze precipitaten veroorzaken plaatselijke variaties van de mechanische eigenschappen, die de toepasbaarheid en efficiëntie van verdere mechanische behandelingen beperken. Bijvoorbeeld, extrusie van een zojuist gegoten legering leidt tot haarscheurtjes aan het metaaloppervlak of tot een *zg.* sinaasappelhuid als ongewenste eigenschappen. Daarom past men vóór de extrusie een warmtebehandeling, ook wel homogenisatie genoemd, toe om deze variaties in de samenstelling te verwijderen en zo de mechanische eigenschappen te verbeteren. Tijdens deze warmtebehandeling wordt de legering opgewarmd tot een temperatuur net onder de eutectische temperatuur en een aantal uur op deze temperatuur gehouden. Hierdoor wordt een homogener micro-structuur verkregen door diffusie-transport van de legeringselementen. Een aantal belangrijke processen die tijdens het homogeniseren plaatsvinden zijn:

1. het oplossen van aanwezige geïsoleerde deeltjes,
2. morfologische veranderingen van de precipitaten, in het bijzonder de transformatie van metastabiele plaatvormige  $\beta$ -deeltjes naar stabiele bolvormige  $\alpha$ -deeltjes,
3. nucleatie en groei van precipitaten gevormd door legeringselementen uit oververzadigde oplossing.

De grote hoeveelheid benodigde energie maakt het homogeniseren erg kostbaar. Een te korte homogenisatiebehandeling geeft een slechte legering of anderzijds falen tijdens het vervaardigingsproces, wat dan weer leidt tot zeer hoge kosten. Aan de andere kant geeft een te lange homogenisatiebehandeling een onnodig hoge extra kostenpost. Daarom zijn goede voorspellingen voor de benodigde homogenisatietijd van grote waarde. Computersimulaties kunnen snel en goedkoop waardevolle schattingen geven voor de benodigde homogenisatietijd.

Dit werk is grotendeels geconcentreerd op het oplossen van deeltjes, al worden nucleatie en deeltjesgroei kort bestudeerd in hoofdstuk 7. De oploskinetiek wordt bepaald door vele factoren, zoals morfologie, diffusiecoëfficiënten en randvoorwaarden aan het deeltjes/matrix grensvlak uit de thermodynamica. Het wiskundige model om deze processen te beschrijven is een vector Stefan probleem waarin de diffusievergelijkingen, die het transport van legeringselementen beschrijven, opgelost worden in het tijdsafhankelijke aluminium-rijke gebied. Behoud van massa leidt tot een bewegingsvergelijking voor het bewegende grensvlak tussen het deeltje en de aluminium-rijke matrix. Verder geeft de massabalans van ieder legeringselement dat het grensvlak oversteeft een sterke niet-lineaire koppeling tussen de concentratiegradiënten en grensvlakconcentraties. Ons doel is een robuuste numerieke methode te ontwikkelen om computersimulaties van het deeltjes oplosproces uit te voeren. Hierbij kunnen morfologische veranderingen voorkomen in één, twee of drie dimensies. Al is het werk overwegend voor aluminiumlegeringen uitgevoerd, toch kunnen de meeste wiskundige inzichten ook gebruikt worden voor andere metaallegeringen en zelfs voor keramische materialen.

Wanneer we werken aan een diffusieprobleem met een bewegende rand, is de eerste vraag hoe we met het bewegende grensvlak omgaan. "Front-capturing" methoden blijken het meest geschikt te zijn om bewegende randwaardeproblemen, waarin topologische veranderingen voorkomen, op te lossen. In deze klasse van methoden worden de fase-veld methode en level-set methode het vaakst gebruikt om aan faseformaties te rekenen. Fase-veld methoden gaan uit van een diffuse grenslaag, waar de faseformatie plaatsvindt. De randvoorwaarden worden hier niet direct opgelegd. Een groot bezwaar, in de ogen van de schrijfster, is dat een nauwkeurige behandeling van het gebied rond de rand adaptieve gridtechnieken vereist. Dit geeft een lage bovengrens van de tijdstap en dus een langdurige berekening. Daarom wordt de level-set methode gebruikt om de bewegende rand te beschrijven. De rand in de begintoestand valt samen met de verzameling van de punten in het rekengebied waarvoor de level-set functie de nulwaarde oplevert. De overige functiewaarden leveren de afstand tot aan de rand, waarbij het teken afhangt van aan welke kant van de rand het punt zich bevindt. De beweging van de rand wordt beschreven met een transportvergelijking voor de level-set functie, waarbij de snelheid een continue uitbreiding is van de snelheid van de rand. Verder wordt de level-set functie gereïntialiseerd, zodat de functie weer een afstandsfunctie (tot de bewegende rand) wordt. Het teken hangt dan weer af van de kant van de rand het punt zich bevindt. Dit laatste wordt gedaan om vlakke of steile gradiënten in de buurt van de bewegende rand te vermijden. Hiermee wordt het rekengebied geparametriseerd door de level-set functie en het bewegende grensvlak wordt behandeld als een 'scherpe' rand.

Een eenvoudige combinatie van de eindige differentie en eindige elementen methode wordt beschreven in dit werk. De hyperbolische vergelijkingen (te weten de transport vergelijking voor de rand en de reïntialisatie van de level-set functie) verkregen uit de level-set methode worden opgelost op een Cartesisch rooster met eindige differenties. De continue uitbreiding van de frontnelheid over het gehele rekengebied wordt verkregen door het adveceren van informatie vanaf de bewegende rand. De diffusieproblemen worden opgelost met SEPRAN, gebruikmakend van de eindige elementen methode. Een nieuwe techniek, de cut-cell methode, geeft een lineaire reconstructie van het bewegende grensvlak en realiseert de aanpassing van het achtergrond rooster aan de grensvlakpositie. Hiermee wordt de behandeling van de randvoorwaarden vrij eenvoudig. Tenslotte wordt een iteratieve procedure gebruikt om de niet-lineaire koppeling voor een vector Stefan probleem voor legeringen met meerdere elementen aan te kunnen. Zo'n iteratie wordt uitgevoerd over alle punten op de bewegende rand, waarvan het aantal achtereenvolgens lineair en kwadratisch voor twee- en driedimensionale problemen toeneemt met de resolutie van het rekenrooster. Daarom wordt een vaste-punten iteratie, die welliswaar traag convergeert maar wel per iteratie goedkoop is, gebruikt.

De innovaties in dit werk zijn het oplossen van het vector Stefan probleem met de level-set methode, het simuleren van deeltjesoplossen in drie ruimtedimensies, behandeling van topologische veranderingen van de geometrie, en de combinatie van de eindige differentie en eindige elementen

methoden die leidt tot het gebruik van de cut-cell methode. De nauwkeurigheid van de uitkomsten is onderzocht door te vergelijken met gelijkvormigheidsoplossingen of stationaire oplossingen. Er is ook aandacht gegeven aan de rektijden van het algoritme. Verder worden zowel enkelvoudige deeltjes als morfologische veranderingen, veroorzaakt door scheurtjes aan het deeltjesoppervlak, doorgerekend.



---

# Acknowledgments

The work presented in this thesis was conducted during my four years appointment at the Delft Institute of Applied Mathematics (DIAM), section of Scientific Computing, of the Delft University of Technology. This research was possible thanks to the financial support of the Dutch Technology Foundation (STW), project DWI.5983. My gratitude goes likewise to the committee members of this project for all the fruitful discussions from which I benefited.

First of all, I would like to thank Prof. Francisco J. Lisbona and Dr. Francisco J. Gaspar from the University of Zaragoza (Spain), and Dr. Kees Oosterlee from the Delft University of Technology for giving me the opportunity to come to Delft, as well as to Prof. Piet Wesseling for taking me as a PhD student into his group. I would like to express my most sincere gratitude to my supervisors Dr. Kees Vuik and Dr. Fred Vermolen for introducing me into the fascinating world of the Stefan problems, and for all the encouragement and valuable discussions which I enjoyed throughout these years. I am very thankful to Piet Wesseling, Kees Vuik and Fred Vermolen for their critical reviews of the early manuscript which led to an improvement of this thesis. Their kind help with the *samenvatting* and the *stellingen* is also appreciated. In addition, I feel deeply grateful to ir. Guus Segal for his assistance with the finite element package Sepran during the implementation of the code developed for this work. Without him, this thesis would not have been the same.

It has been a pleasure to collaborate across research disciplines. I especially thank Prof. Sybrand van der Zwaag for bringing his valuable insight into this research, who certainly motivated me to look at the problems studied here from a different perspective, hopefully closer to that of the material scientists. My appreciation goes also to Dr. Niels Kuijpers, Dr. Lie Zhao and Dr. Pedro Rivera for all the very inspiring discussions, from which I was prompted to study different properties of various metallurgical systems.

I would like to thank all my colleagues throughout these years at the Scientific Computing group for creating an excellent atmosphere, especially to Ariel Almendral, John Brusche and Sander van Veldhuizen for the moments of joy and/or discussion at the HB 07.250. I truly took delight in sharing the office with them. I deeply appreciate the help of our secretaries Dori Steeneken and Diana Droog who made bureaucracy seem almost invisible. To our system administrator Kees Lemmens I am very thankful for all the valuable help concerning computers and laptops. If within these years I have learned something about Linux, it is indeed thanks to him.

Finally, I would like to take this opportunity to express my gratitude to all these people who have made these years far from home easier. I have been lucky enough to find them both inside and outside of the university environment in Delft, as well as having some of them back home.

All of them are gratefully acknowledged for their friendships and the time we have spent and still have to spend together. To conclude, I would like to express my deepest gratitude to my brother, whose kind help has been used in many occasions, and to my parents, who have supported me in the most generous way all my life. To them I feel enormously indebted.

Etelvina Javierre Pérez  
Delft, October 2006

---

# CONTENTS

<b>Summary</b>	<b>iii</b>
<b>Samenvatting</b>	<b>v</b>
<b>Acknowledgments</b>	<b>ix</b>
<b>List of Figures</b>	<b>xv</b>
<b>List of Tables</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Technological background . . . . .	1
1.2 Mathematical models for particle dissolution . . . . .	3
1.3 Outline of the thesis . . . . .	5
<b>2 The physical problem</b>	<b>7</b>
2.1 Introduction . . . . .	7
2.2 The scalar Stefan problem . . . . .	7
2.3 The vector Stefan problem . . . . .	8
2.4 Non-equilibrium interface conditions for the scalar Stefan problem . . . . .	9
2.4.1 Gibbs-Thomson effect . . . . .	9
2.4.2 Interface reactions . . . . .	10
2.4.3 The modified Stefan problem . . . . .	10
<b>3 A comparison of numerical models for one-dimensional scalar Stefan problems</b>	<b>11</b>
3.1 Introduction . . . . .	11
3.2 The physical problem . . . . .	13
3.2.1 A solid-liquid phase transformation . . . . .	13
3.2.2 A solid-solid phase transformation . . . . .	14
3.2.3 The modified Stefan problem . . . . .	14
3.3 Survey of numerical methods . . . . .	14
3.3.1 The moving grid method . . . . .	15
3.3.2 The level set method . . . . .	15
3.3.3 The phase field method . . . . .	16
3.4 Numerical results . . . . .	18
3.4.1 Classical Stefan problems . . . . .	18
3.4.2 Modified Stefan problems . . . . .	22

3.5	Conclusions . . . . .	24
<b>4</b>	<b>A Level Set Method for higher-dimensional scalar Stefan problems: Particle dissolution in binary alloys</b>	<b>27</b>
4.1	Introduction . . . . .	27
4.2	The Level Set Method . . . . .	29
4.2.1	Extension of the front velocity . . . . .	30
4.2.2	Reinitialization . . . . .	32
4.3	The diffusion problem . . . . .	32
4.4	The cut-cell approach . . . . .	34
4.5	Numerical results . . . . .	36
4.5.1	Extension of the front velocity . . . . .	37
4.5.2	Convergence to the steady-state solutions and mass conservation . . . . .	38
4.5.3	Comparison with a Moving Grid Method . . . . .	40
4.5.4	A 3D test problem involving topological changes . . . . .	41
4.6	Conclusions . . . . .	44
<b>5</b>	<b>Higher-dimensional vector-valued Stefan problems: Particle dissolution in multi-component alloys</b>	<b>45</b>
5.1	Introduction . . . . .	45
5.2	Extension of the front velocity for varying $c^{sol}$ . . . . .	46
5.3	The coupled diffusion problems . . . . .	47
5.3.1	Determination of the interface concentrations . . . . .	48
5.4	Numerical results . . . . .	51
5.4.1	Comparison with the similarity and steady-state solutions . . . . .	52
5.4.2	Evolution of the interface concentrations for a two-dimensional test problem . . . . .	54
5.4.3	Dissolution of a three-dimensional dumbbell shaped particle . . . . .	55
5.4.4	Evaluation of the computational cost of the fixed point iteration . . . . .	60
5.5	Conclusions . . . . .	60
<b>6</b>	<b>Metallurgical applications</b>	<b>65</b>
6.1	Introduction . . . . .	65
6.2	Applications . . . . .	65
6.2.1	Dissolution of spherical cementite . . . . .	65
6.2.2	Spheroidization process in lamellar structures . . . . .	68
6.2.3	The effect of the particle morphology on the dissolution kinetics . . . . .	72
6.3	Conclusions . . . . .	76
<b>7</b>	<b>Models for nucleation and growth</b>	<b>77</b>
7.1	Introduction . . . . .	77
7.2	Models of precipitation . . . . .	77
7.2.1	Particle size distribution models . . . . .	78
7.2.2	Monte Carlo models . . . . .	80
7.3	Growth of precipitates . . . . .	83
7.4	Conclusions . . . . .	84
<b>8</b>	<b>Conclusions</b>	<b>87</b>
8.1	Conclusions . . . . .	87
8.2	Recommendations for future research . . . . .	88
<b>A</b>	<b>Convergence analysis of the fixed-point iteration in the vector Stefan problem using the similarity solution of the planar interface</b>	<b>91</b>
	<b>Bibliography</b>	<b>95</b>



---

List of Publications

101

Curriculum Vitae

103



---

# LIST OF FIGURES

1.1	Time-Temperature diagram of the processing of aluminium alloys. . . . .	1
1.2	Different types of extruded profiles. . . . .	2
1.3	Optical microscopy pictures of the microstructure before and after homogenization. . . . .	2
2.1	The representative cell $\Omega$ . . . . .	7
3.1	The solid-solid phase transformation problem with the moving grid (MGM) and the level set (LSM) methods. . . . .	19
3.2	The solid-liquid phase transformation problem with the moving grid (MGM), the level set (LSM) and the phase field (PFM) methods. . . . .	20
3.3	Mesh trajectories of (a) the moving grid method, (b) the level set method, and (c) the phase field method. . . . .	21
3.4	Evolution of the front for two different initial radius. . . . .	23
3.5	Evolution of the interface for various undercooling rates. . . . .	24
4.1	Extension of the front velocity in the Cartesian directions: neighbours inside the diffusive phase marked with black rectangles, and neighbours inside the particle marked with grey rectangles. Information travels off the interface. . . . .	31
4.2	Subdivision of a two-dimensional element by the cut-cell method. . . . .	35
4.3	Subdivision of a three-dimensional element with three intersection points by the cut-cell method. . . . .	35
4.4	Subdivision of a three-dimensional element with four intersection points by the cut-cell method. . . . .	36
4.5	The initial position of the interface with the Cartesian extension (left) and the normal extension (right) of the front velocity for the square interface problem. . . . .	38
4.6	Circular interface: snapshots at intervals of 0.1s, using the Cartesian extension of the front velocity. . . . .	39
4.7	Circular interface. 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). . . . .	41
4.8	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) for the square interface problem. . . . .	42
4.9	Interface position and mesh with the moving grid method at times $t = 0.005$ , $t = 0.205$ and $t = 0.405$ for the square interface problem. . . . .	42
4.10	Dissolution of an initially perturbed cylindrical particle. Time evolution is from left to right and from top to bottom. . . . .	43

5.1	The Cartesian mesh with the interface. Interface points denoted by $\blacklozenge$ are those obtained with the cut-cell method, whereas $\mathbf{x}_f$ denotes the interface point associated with the mesh point $\mathbf{x}$ . . . . .	46
5.2	The background FE mesh with the interface $\Gamma$ at two consecutive times, the interface points computed with the cut-cell method and a reconstruction $\Gamma'(t^{n+1})$ (dashed curve) of $\Gamma(t^{n+1})$ by advection in the normal direction the points on $\Gamma(t^n)$ . . . . .	51
5.3	Interface position vs time for the dissolution of a planar interface in an alloy with one, two, three and four chemical species respectively. Numerical solutions presented with solid curves and similarity solutions presented with dashed curves. . . . .	53
5.4	Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom): $t = 3.3 \times 10^{-3}$ , $9 \times 10^{-3}$ , $1.4 \times 10^{-2}$ . Interface position (left) and $c_1^{sol}$ (right). . . . .	56
5.5	Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom): $t = 3.3 \times 10^{-3}$ , $9 \times 10^{-3}$ , $1.4 \times 10^{-2}$ . $c_2^{sol}$ (left) and $c_3^{sol}$ (right). . . . .	57
5.6	Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom): $t = 3 \times 10^{-2}$ , $5 \times 10^{-2}$ , $5 \times 10^{-1}$ . Interface position (left) and $c_1^{sol}$ (right). . . . .	58
5.7	Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom): $t = 3 \times 10^{-2}$ , $5 \times 10^{-2}$ , $5 \times 10^{-1}$ . $c_2^{sol}$ (left) and $c_3^{sol}$ (right). . . . .	59
5.8	Dissolution of a three-dimensional dumbbell-like particle in a ternary alloy. . . . .	61
5.9	The computational cost of the dissolution of a cylindrical particle. Processor Intel <sup>®</sup> Pentium <sup>®</sup> 4 (3.00GHz) . . . . .	62
6.1	Carbon concentration profiles at consecutive times. . . . .	66
6.2	The interfacial positions of the cementite-austenite and austenite-ferrite interfaces as a function of time. . . . .	67
6.3	The interfacial positions of the cementite-austenite interface as a function of time for several values of the initial position of the austenite-ferrite interface. . . . .	67
6.4	Dissolution of a cementite plate with a knucklebone shape. . . . .	70
6.5	Dissolution of a cementite plate with a sinusoidal perturbation. . . . .	71
6.6	The volume of the cementite particle as a function of time for several perturbations on its surface. . . . .	72
6.7	The configuration and numbering of the dissolving isolated particles at a cross-section at the plane $z = 0.25\mu\text{m}$ . The bounding box denotes the boundary of the computational domain. . . . .	73
6.8	The volume of several individual cementite particles as a function of time for the case of 20 particles (T1) and for 16 particles in which particles III, VIII, XIII and XVIII have been removed (T2). . . . .	73
6.9	Interface profile at the cross section parallel to the $xy$ -plane at $z = 0.25$ for the 20 platelets (T1). Time evolution goes from the exterior towards the interior. . . . .	74
6.10	Interface profile at the cross section parallel to the $xy$ -plane at $z = 0.25$ for the 16 platelets (T2). Time evolution goes from the exterior towards the interior. . . . .	74
6.11	Dissolution of the $\text{Mg}_2\text{Si}$ particles: (a) Normalized particle area vs normalized time, (b) dissolution time vs arc length of the initial interface. . . . .	75
6.12	Dissolution of the $\text{Mg}_2\text{Si}$ particles: eccentricity of the particle vs normalized dissolution time . . . . .	76
7.1	Results for the particle size distribution model . . . . .	80
7.2	Arrangement of the lattice for two consecutive states $\mu$ and $\nu$ . . . . .	81
7.3	Initial arrangement of the atoms. . . . .	82
7.4	Snapshots of the precipitation process: nucleation, growth and coarsening respectively. . . . .	82
7.5	Number of particles and volume with the Monte Carlo method. . . . .	83
7.6	Growth of a diamond-like particle. Snapshots of the interface position at times 0, 0.01, 0.025, 0.05, 0.075 and 0.1. . . . .	85

---

# LIST OF TABLES

3.1	Parameters for the solid-solid phase transformation test problem. . . . .	18
3.2	Convergence behaviour of the moving grid and the level set methods for the solid-solid phase transformation problem. Similarity solution $s(0.1) = 0.276815$ . . . . .	19
3.3	Parameters for the solid-liquid phase transformation test problem. . . . .	19
3.4	Values of $\alpha$ for different values of $K_{liq}$ . . . . .	20
3.5	Parameters for the critical radius instability solidification test problem. . . . .	22
3.6	Parameters for the undercooling test problem. . . . .	23
4.1	Average CPU-time (seconds) in the extension of the front velocity for the squared interface problem. Processor Intel <sup>®</sup> Pentium <sup>®</sup> 4 (2.80GHz). . . . .	38
4.2	Relative errors in the interface position and mass at time $t_{end} = 0.75$ with the level set method for the planar interface problem. Initial mass of the system $m(0) = 0.39225$ and final position of the interface $s_{\infty} = 0.4225$ . . . . .	39
4.3	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 $m(0) = 0.3446$ . . . . .	40
4.4	Relative errors in the mass for the squared interface problem. Initial mass: $m(0) = 0.3458$ . . . . .	40
4.5	Relative errors in the interface position and mass at time $t_{end} = 0.75$ when the moving grid method is used for the planar interface test problem. . . . .	41
5.1	Asymptotic behaviour of the interface concentrations for different values of $D_1$ . . .	49
5.2	Parameters for the test problem of Section 5.4.1. . . . .	52
5.3	Effective particle concentrations, solid solubility, diffusion coefficient and dissolution rate for the quasi-binary alloy, and the dissolution rate coefficient for the multi-component alloy. . . . .	53
5.4	Relative error in the interface position at $t = 0.75$ for the problem with only one diffusive species. Effect of $tol_{cc}$ on the accuracy: $tol_{cc}=0.01$ is the default value for dissolution in binary alloys and $tol_{cc}=0.3$ is the default value for dissolution in multi-component alloys. . . . .	54
5.5	Relative error in the interface position at $t = 0.75$ for the problems with two, three and four diffusive species. . . . .	54
6.1	Physical parameters for the dissolution of the Mg <sub>2</sub> Si particles. . . . .	72
7.1	Parameters for the particle distribution model. . . . .	79

A.1 Performance of the fixed-point iteration using the similarity solution of a planar interface . . . . .	93
--	----

---

---

# CHAPTER 1

---

## Introduction

### 1.1 Technological background

Production of industrial aluminium alloys involves a number of intermediate steps after bauxite is extracted from earth until the commercial product is realized. First, the aluminium metal is derived from bauxite by means of a wet chemical, caustic leach method (the Bayer process), followed by an electrolytic process (the Hall-Heroult process). The molten aluminium obtained in this way is enriched with other metallic elements (magnesium, manganese, silicon and iron, among others), in order to optimize the mechanical properties. Subsequently, the aluminium-based alloy is cast into a mould, where it solidifies. The solid material is then ready for further processing. Figure 1.1 shows the time-temperature diagram of different heat treatments used in the production of aluminium alloys.

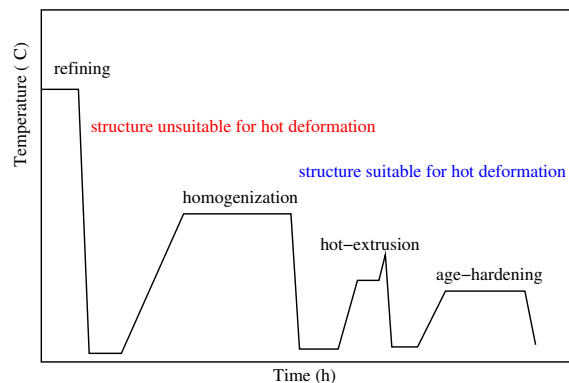


Figure 1.1: Time-Temperature diagram of the processing of aluminium alloys.

Hot-extrusion is a mechanical process in which the preheated billets are pushed through shape dies to produce extruded profiles, such as those shown in Figure 1.2, that can be used in mechanical constructions such as engines, etc. This process is one of the most important steps in the production of aluminium alloys, since it determines the properties and the applicability of the final product. Furthermore, extruded materials may need to undergo further processing (such as surface treatments, age-hardening, etc), depending on their final use. Age-hardening is a heat treatment

in which a fine distribution of precipitates is produced in order to enhance the strength of the alloy.

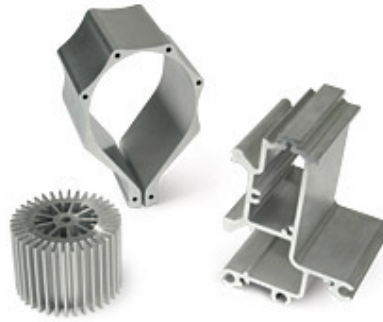
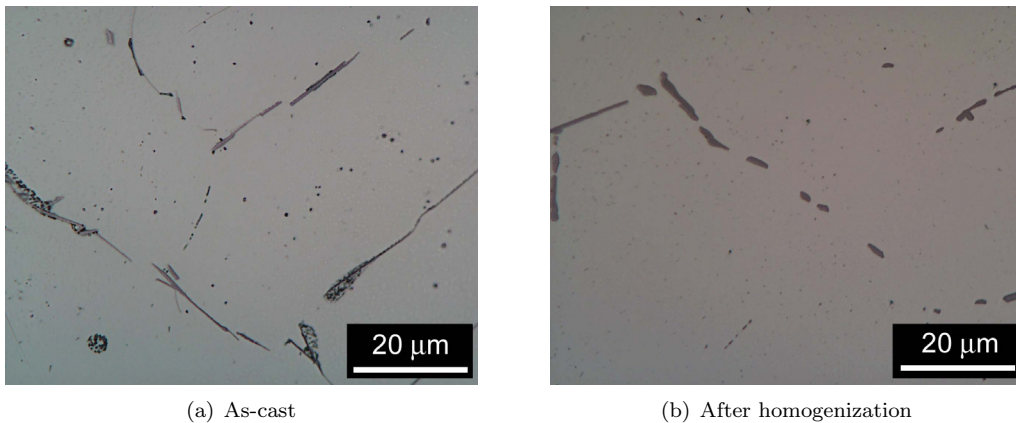


Figure 1.2: Different types of extruded profiles.

However, as-cast alloys are not suited for hot-extrusion. During the solidification process, segregation of the alloying elements takes place and small precipitates are formed, see Figure 1.3(a). These precipitates result into local variations of the mechanical properties, which limit the applicability and efficiency of further mechanical treatments. In order to remove these compositional variations, and hence to improve the mechanical properties, an intermediate annealing treatment, referred to as homogenization, is applied prior to extrusion. During homogenization, the temperature of the as-cast billets is gradually increased until just below their eutectic temperature, and kept at this temperature for several hours. After this time, the billets are quenched rapidly to avoid precipitation. Because of this process, a more homogeneous microstructure is obtained due to diffusional transport of the alloying elements. Figure 1.3(b) shows the homogenized microstructure after 32 hours at 590°C.



(a) As-cast

(b) After homogenization

Figure 1.3: Optical microscopy pictures of the microstructure before and after homogenization.

Homogenization is a very expensive process, because of its energy requirements. An underestimate of the homogenization time results in bad properties or failure during the extrusion process, whereas an overestimate of this time leads to extra energy consumption. Therefore, accurate predictions of the homogenization time are of great value. Until recently, it was determined by simple rules of thumb. However, computer simulations can provide better insight in the homogenization times for different alloy compositions, precipitate volume fractions and temperatures, as well as in the metallurgical processes involved in it. Among these processes, of major importance are:



1. the dissolution of precipitates that are present as isolated particles,
2. the morphological changes of the precipitates, especially the transformation of metastable  $\beta$ -particles of plate-like structure into arrays of stable  $\alpha$ -particles with sphere-like structure [1]
3. the nucleation and growth of precipitates formed from alloying elements that are in a super-saturated solution.

This thesis is mainly focused on dissolution processes, although nucleation and growth processes are also considered in Chapter 7. Dissolution (and growth) kinetics are determined by many factors such as morphology, effective diffusion coefficients and thermodynamic boundary conditions at the interface between the particle and the aluminium-rich phase. Our aim is to develop a mathematical model to perform computer simulations of dissolution (and/or eventually growth) of isolated particles for different thermodynamic regimes and especially for different particle morphologies.

## 1.2 Mathematical models for particle dissolution

Dissolution of particles is controlled by two consecutive processes:

1. the interface reactions, which involve the mechanisms that result in the migration of atoms across the interface from the particle towards the aluminium-rich phase,
2. the long-distance diffusion, in which the atoms separated from the particle are redistributed along the aluminium-rich phase by diffusional transport.

Long-distance diffusion models [2, 3, 4] assume that the interface is always in thermodynamic equilibrium with the surrounding phase, implying that the interface reactions proceed infinitely fast. In these models the interface concentration is the solid solubility derived from the phase diagram. Non-equilibrium conditions at the interface were incorporated in [5, 6]. Nolfi *et al.* [5] use a Robin condition which relates the interfacial concentration with the flux of atoms through the interface. Aaron *et al.* [6] introduce the influence of the curvature on the interface concentration by means of the Gibbs-Thomson relation [7]. More recently, Svododa *et al.* [8] use a thermodynamically based method to predict non-equilibrium kinetics when mechanical and chemical forces are exerted on the interface. Later modelling of particle dissolution has been extended to multi-component alloys. Reiso *et al.* [9] investigated the dissolution of  $\text{Mg}_2\text{Si}$  particles in aluminium alloys mainly experimentally. Transformations in steel were studied in [10, 11]. All the analyses indicate that the addition of secondary alloying elements can influence the dissolution kinetics strongly. Vermolen *et al.* [12] considered the dissolution of a stoichiometric particle in a ternary alloy. They formulated a hyperbolic relation on the interfacial concentrations based on the concept of local equilibrium between phases. In a series of articles, Vermolen and Vuik [13, 14, 15, 16] realize an intensive analysis on the following aspects of particle dissolution in multi-component alloys:

- (a) existence, uniqueness and monotonicity of the solution,
- (b) effects of the particle geometry on the dissolution process,
- (c) metallurgical implications of cross-diffusion on the rate of dissolution,
- (d) stability and efficiency of the numerical solution obtained with a moving grid method to tackle the moving interface and an iterative method to solve the hyperbolic coupling of the interface concentrations.

In all these references, particle dissolution is modelled as a Stefan problem: the diffusion equations are to be solved in a domain, namely the aluminium-rich phase, which changes in time, and hence has to be found as part of the solution. Several numerical methods have been developed to solve Stefan problems. Crank [17] provides a good introduction to Stefan problems and presents an extensive collection of classical numerical techniques.

Front-tracking methods represent the interface explicitly by a set of points which may or may not be connected to the background computational mesh. Murray and Landis [18] introduced the moving grid method for one-dimensional solidification problems. The computational mesh was such that it was uniform at both sides of the interface, and the moving interface is always located on the same mesh point. Segal *et al.* [19] extend this method to two-dimensional particle dissolution problems. The movement of the mesh was incorporated into the governing equations by means of the so-called Arbitrary Lagrangian Eulerian approach in both articles. Juric and Tryggvason [20] combine a stationary mesh technique with a nonstationary curve or surface which represents the moving interface to solve solidification problems in two and three dimensions.

Front-capturing methods use an implicit representation of the moving interface, adding an artificial unknown to the governing equations. Level set methods, firstly introduced by Osher and Sethian [21], identify the interface with the zero level set of a continuous function, and reformulate its motion as a transport equation of this function. Although the level set method has been used to solve Stefan problems for solidification by several authors [22, 23, 24], it has never been applied to single-phase vector Stefan problems, such as particle dissolution, according to the author's knowledge. From another point of view, phase field methods introduce a diffuse interface region where the phase transformations occur. Points of the computational domain are parameterized as being inside or outside the interface region, by means of a function that is constant away from the interface region and has a steep gradient within. The evolution of the system is derived from the minimization of a free energy functional. Phase transitions in binary alloys are considered in [25, 26], whereas the extension to multi-component alloys is done by Grafe *et al.* [27].

Deciding which numerical method is most suitable depends on the problem that is to be solved as well as on the available resources. Both moving grid and phase field methods as presented in the abovementioned references deal with the boundary conditions at the interface easily. For the moving grid method this is because a set of nodal points lies on the interface. For the phase field method this is because the diffusive interface avoids direct implementation of the boundary conditions. However, this task requires more work with the level set method, for which interpolation strategies are normally used. The accuracy of the moving grid solutions depends on the quality of the computational mesh. Thus, remeshing will eventually be necessary. Furthermore, topological changes in the particle shape are very difficult to tackle, although major improvements have been made [28] in this respect. On the other hand, front-capturing methods deal with topological changes in a natural fashion, because of the implicit representation of the interface. Level set methods generally identify the interface with the zero level set of a signed distance function. This representation simplifies the computation of geometrical quantities on the interface, such as the normal vector or the curvature. In order to maintain this representation, a nonlinear hyperbolic equation is usually solved after the interface is advected. High order schemes should be applied to avoid an unwanted shift of the interface due to the discretization errors. Furthermore, in Stefan problems, the front velocity is only defined at the front position itself. Thus, it must be continuously extended onto the whole computational domain before the interface is advected. On the other hand, phase field methods require adaptive mesh strategies [29, 30, 31] in order to resolve the interfacial region accurately. Note that the sharp interface problem is to be recovered in the limit of vanishing interface thickness [32]. These fine meshes consequently impose a very limiting time-step criterion, resulting in large computational times. Another drawback of phase field methods is that physically justifiable parameter values are usually obtained from large and expensive thermodynamic databases. Otherwise, fitting techniques coupling experiments and simulations are unavoidable.

Therefore, since morphological changes are to be simulated, a front-capturing method is to be chosen. Moreover, as it will be shown in Chapter 3, the level set method is preferred to the phase field method because it provides a simpler way of modelling of single-phase vector Stefan problems.

## 1.3 Outline of the thesis

In Chapter 2, the physical problem is described. Particle dissolution in both binary and multi-component alloys are presented as scalar and vector-valued Stefan problems respectively.

Chapter 3 provides a comparison of three numerical methods (moving grid, level set and phase field) for two well known scalar Stefan problems in one-spatial dimension, the melting problem and the solid state phase transformation in binary metallic alloys, in order to determine which of them is more suitable for solving Stefan problems in higher dimensions.

Chapter 4 gives a detailed description of a level set method for solving particle dissolution in binary alloys in two- and three-spatial dimension. Special attention is paid to the extension of the front velocity and to the adaptation of the background mesh to the interface location. The performance of this method is compared with the moving grid method presented in [19].

Chapter 5 gives a generalization of the level set method presented in Chapter 4 to solve particle dissolution in multi-component alloys. This chapter is focused on the solution of the nonlinear coupled diffusion problems.

Chapter 6 is devoted to the application of the level set method developed in Chapters 4 and 5 to cementite dissolution in steel. Numerical simulation of phase transformations with multiple phases and spheroidization process are the targets of this chapter.

Chapter 7 gives two different nucleation models and describes the adaptations that should be carried out on the level set method described in Chapter 4 in order to introduce the Gibbs-Thomson effect on the interface concentration.

Finally, Chapter 8 summarizes the conclusions of this work and brings some remarks and suggestions for future research.



---

---

# CHAPTER 2

---

## The physical problem

### 2.1 Introduction

After manufacturing, an alloy is cast into a mould. The state of the alloy is then referred to as the as-cast state. The as-cast microstructure is simplified to a representative cell  $\Omega$  containing a diffusive phase  $\Omega_{dp}$  and a particle  $\Omega_{part}$ , as sketched in Figure 2.1. The phases  $\Omega_{dp}$  and  $\Omega_{part}$  are disjoint, and the interface separating them is denoted by  $\Gamma$ . The alloy consists of aluminum and one or more chemical species, leading to a binary or multicomponent alloy respectively. Aluminum is the solvent in which the particle is allowed to dissolve or grow. The dissolution or growth is governed by the diffusional transport of atoms from or towards the particle, giving as a result the movement of the interface. In binary alloys, only one diffusion problem is to be solved, resulting in a so-called *scalar* Stefan problem. However, in multicomponent alloys, several diffusion problems are to be solved, one for each chemical species in the alloy. Consequently, these problems are referred to as *vector-valued* Stefan problems. The governing equations that describe the dissolution of particles under the assumption of long-distance diffusion are given in detail in Section 2.2 and 2.3 respectively. Next, the non-equilibrium conditions on the interface are presented in Section 2.4 for scalar Stefan problems. Their generalization for vector Stefan problems is straightforward and hence omitted.

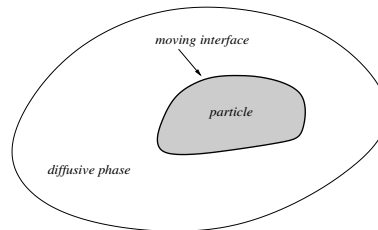


Figure 2.1: The representative cell  $\Omega$ .

### 2.2 The scalar Stefan problem

Consider a binary alloy, formed of aluminum and a secondary alloying element. The particle is assumed to remain stoichiometric during the entire process, hence the concentration  $c$  ( $mol/m^3$ )

of the secondary species inside the particle  $\Omega_{part}$  is given by

$$c(\mathbf{x}, t) = c^{part}, \quad \mathbf{x} \in \Omega_{part}(t), \quad t \geq 0, \quad (2.1)$$

where  $\mathbf{x}$  denotes a point in the computational domain  $\Omega$  and  $t$  denotes time. The initial concentration  $c(\mathbf{x}, 0)$  inside the diffusive phase is given. The particle dissolves/grows due to Fickian diffusion in the diffusive phase

$$\frac{\partial c}{\partial t}(\mathbf{x}, t) = D\Delta c(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_{dp}(t), \quad t > 0, \quad (2.2)$$

where  $D$  ( $m^2/s$ ) denotes the diffusion coefficient, which is supposed to be independent of the concentration. The concentration at the interface  $\Gamma$ , separating  $\Omega_{part}$  and  $\Omega_{dp}$ , is assumed to be given by the constant value  $c^{sol}$

$$c(\mathbf{x}, t) = c^{sol}, \quad \mathbf{x} \in \Gamma(t), \quad t \geq 0, \quad (2.3)$$

which is the equilibrium concentration as predicted by thermodynamics. No flux of the alloying element is allowed through the boundary not being  $\Gamma$ , hence

$$D\frac{\partial c}{\partial \mathbf{n}}(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega_{dp}(t) \setminus \Gamma(t), \quad t > 0. \quad (2.4)$$

Since the mass of the whole system must be conserved, the concentration gradient on the side of  $\Omega_{dp}$  at  $\Gamma$  causes its displacement. The normal component of the interface velocity  $v_n$  is given by [19]

$$(c^{part} - c^{sol})v_n(\mathbf{x}, t) = D\frac{\partial c}{\partial \mathbf{n}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad (2.5)$$

where  $\mathbf{n}$  is the unit normal vector on the interface pointing outward with respect to  $\Omega_{part}(t)$ . The problem as posed by equations (2.1)-(2.5), in which the solution of the diffusion equation and the domain in which this solution holds are unknown, belongs to a category of moving interface problems called Stefan problems.

## 2.3 The vector Stefan problem

Consider a multi-component alloy formed of aluminum and  $p$  secondary chemical species. The particle grows/dissolves due to Fickian diffusion of the chemical species in the diffusive phase

$$\frac{\partial c_i}{\partial t}(\mathbf{x}, t) = \sum_{j=1}^p D_{ij}\Delta c_j(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_{dp}(t), \quad t > 0, \quad i \in \{1, \dots, p\}, \quad (2.6)$$

where  $c_i$  denotes the concentration of the  $i$ th 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. If  $D_{ij} = 0$  for  $i \neq j$ , then the classical diffusion equations are recovered for each chemical species. The initial concentration  $c_i(\mathbf{x}, 0)$  in the diffusive phase is given for  $i = 1, \dots, p$ . The particle is assumed to remain stoichiometric during the process. Hence, the particle concentrations remain constant

$$c_i(\mathbf{x}, t) = c_i^{part}, \quad \mathbf{x} \in \Omega_{part}(t), \quad t \geq 0, \quad i \in \{1, \dots, p\}. \quad (2.7)$$

Furthermore, the concentrations  $c_i^{sol}$  on the moving interface  $\Gamma$ , separating  $\Omega_{part}$  and  $\Omega_{dp}$ , satisfy the hyperbolic relation [12]

$$\prod_{i=1}^p (c_i^{sol}(\mathbf{x}, t))^{\tilde{n}_i} = \mathcal{K}(T), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad (2.8)$$

which is derived from the Gibbs-free-energy of the stoichiometric elements, where  $\tilde{n}_i$  denotes the stoichiometric number of the  $i$ th chemical species and  $\mathcal{K}(T)$  denotes the solubility product which depends on the temperature  $T$  through an Arrhenius relation [1] (page 12). Thus, for isothermal processes,  $\mathcal{K}(T)$  is a positive constant. No flux of the alloying elements is allowed through the boundary not being  $\Gamma$ :

$$\sum_{j=1}^p D_{ij} \frac{\partial c_j}{\partial \mathbf{n}}(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega_{dp}(t) \setminus \Gamma(t), \quad t > 0, \quad i \in \{1, \dots, p\}. \quad (2.9)$$

When cross-diffusion terms are neglected (*i.e.*  $D_{ij} = 0$  for  $i \neq j$ ) or the matrix  $\mathbf{D} = (D_{ij})_{1 \leq i, j \leq p}$  is not singular, then Eq. (2.9) is equivalent to

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

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

$$(c_i^{part} - c_i^{sol}(\mathbf{x}, t))v_n(\mathbf{x}, t) = \sum_{j=1}^p D_{ij} \frac{\partial c_j}{\partial \mathbf{n}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad i \in \{1, \dots, p\}, \quad (2.10)$$

where  $\mathbf{n}$  is the unit normal vector on the interface pointing outward with respect to  $\Omega_{part}(t)$ . Note that Eqs. (2.10) implicitly impose that

$$\sum_{k=1}^p \frac{D_{ik}}{c_i^{part} - c_i^{sol}(\mathbf{x}, t)} \frac{\partial c_k}{\partial \mathbf{n}}(\mathbf{x}, t) = \sum_{k=1}^p \frac{D_{jk}}{c_j^{part} - c_j^{sol}(\mathbf{x}, t)} \frac{\partial c_k}{\partial \mathbf{n}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad i \neq j. \quad (2.11)$$

The problem as posed by equations (2.6)-(2.11) is commonly referred to as a vector valued Stefan problem.

## 2.4 Non-equilibrium interface conditions for the scalar Stefan problem

In this section we assume that the interface is not in thermodynamic equilibrium with the surrounding phase. Hence, Eq. (2.3) is no longer valid. The Gibbs-Thomson relation, as presented in Section 2.4.1, incorporates the influence of the curvature in the motion of the interface. Interface reactions are described in Section 2.4.2. Finally, the modified Stefan problem which incorporates both curvature effects and interface reactions into the motion of the interface is presented in Section 2.4.3.

### 2.4.1 Gibbs-Thomson effect

The maximum solubility at the interface  $c_{\max}^{sol}$  is related to the local curvature of the interface by the Gibbs-Thomson effect [7]:

$$c_{\max}^{sol}(\mathbf{x}, t) = c^{sol} \exp(2\vartheta\kappa(\mathbf{x}, t)), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad (2.12)$$

where  $\vartheta$  ( $m$ ) denotes the capillarity constant and  $\kappa$  the curvature of the interface. Note that  $\vartheta = \frac{\gamma V_m}{RT}$ , where  $\gamma$  ( $J/m^2$ ),  $V_m$  ( $m^3/mol$ ) and  $R$  ( $J/Kmol$ ) denote the interfacial energy, molar volume of the particle and universal gas constant respectively.

Note that the Gibbs-Thomson relation has a stabilizing influence. Regions with positive curvature have a higher interface concentration than the regions with negative curvature. Therefore, regions of positive curvature move faster in case of dissolution and slower in case of growth than the regions of negative curvature, which eventually results in the extinction of finger-like patterns.

### 2.4.2 Interface reactions

When interface reactions are taken into account, the interface concentration has to be found as part of the solution as well. Assuming a first order reaction at the interface [33] yields

$$K_R(c(\mathbf{x}, t) - c^{sol}) = D \frac{\partial c}{\partial \mathbf{n}}(\mathbf{x}, t) + c(\mathbf{x}, t)v_n(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0, \quad (2.13)$$

where  $K_R$  ( $m/s$ ) is a measure of the rate of the interface reaction. This, with the equation of motion

$$(c^{part} - c(\mathbf{x}, t))v_n(\mathbf{x}, t) = D \frac{\partial c}{\partial \mathbf{n}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0 \quad (2.14)$$

provides the two interface conditions that are needed to determine the interface concentration and interface velocity. Substituting equation (2.14) into equation (2.13) gives

$$c(\mathbf{x}, t) = c^{sol} + \frac{c^{part}}{K_R}v_n(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0.$$

Therefore, for  $K_R$  large the problem is diffusion controlled, whereas for  $K_R$  small the problem is reaction controlled. In the limiting case of  $K_R = \infty$ , the interface concentration is the solid solubility  $c^{sol}$  and hence equations (2.14) and (2.13) simplify to equation (2.5). Note that for aluminium alloys with  $c^{part} > c^0$ , where  $c^0$  denotes the initial concentration inside the diffusive phase, the interface reactions effectively delay the dissolution or growth kinetics, whereas for alloys with  $c^{part} < c^0$  the interface reactions accelerate the transformation.

### 2.4.3 The modified Stefan problem

If both curvature effects and interface reactions are considered, then the interface concentration is given by

$$c(\mathbf{x}, t) = c^{sol} \exp(2\vartheta\kappa(\mathbf{x}, t)) + \frac{c^{part}}{K_R}v_n(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \quad t > 0. \quad (2.15)$$

and the front velocity is computed from Eq. (2.14). The problem given by equations (2.1), (2.2), (2.4), (2.15) and (2.14) is known as the modified Stefan problem, and has been deeply studied for solidification problems. Some references on this problem are [20, 22, 23, 24, 30, 31, 34].



---

---

# CHAPTER 3

---

## A comparison of numerical models for one-dimensional scalar Stefan problems

### 3.1 Introduction

In Stefan problems, the position of an interface between different phases has to be found as part of the solution. These problems describe several phenomena in nature, science and society, among others the melting of the polar ice caps, originally studied by J. Stefan (1835-1893), the dendritic solidification problem [20, 22, 30], the decrease of oxygen in a muscle in the vicinity of a clotted bloodvessel [17], the etching problem [35], the American option pricing problem [36], or the phase transformations in metallic alloys [19]. This chapter deals with a survey of existing numerical techniques for solving one-dimensional Stefan problems. In particular, the melting problem and a solid state phase transformation are considered in parallel because of the resemblance in their governing equations, that will be shown afterwards. Existence of solution was proved by Evans [37], while uniqueness was proved by Douglas [38]. Moreover, the solution of the Stefan problems considered here satisfies the maximum principle in each phase. Furthermore, it is possible to derive analytical expressions for the solution of these problems in an infinite or semi-infinite one-dimensional space. Under these hypotheses, the solution is a function of  $\frac{x-s_0}{\sqrt{t}}$  as proved by Hill [39], and it is often called the similarity solution.

Several numerical methods have been developed to solve various Stefan problems. Crank [17] provides a good introduction to Stefan problems and presents an extensive collection of numerical methods used for these problems. Front-tracking methods use an explicit representation of the interface, given by a set of points lying on the interface, which should be updated at each time step. Murray and Landis [18] compare an adaptive grid procedure with a fixed grid method for a one-dimensional melting problem, and show that the adaptive grid method captures more accurately the interface position, whereas the fixed grid algorithm gives a more precise heat distribution in the whole domain. Segal *et al.* [19] extend this method to two-dimensional particle dissolution problems. In references [18, 19], the interface points are connected to the computational grid, and the movement of the grid is incorporated into the governing equations by means of the total time derivative (also called Arbitrary Lagrangian Eulerian -ALE- approach). Juric and Tryggvason [20] used a fixed grid in space where some variables of the problem (*i.e.* temperature) were calculated, and a moving grid on the interface where the interface heat sources were computed. Information from the interface to the fixed grid was transferred via the immersed boundary method.

On the other hand, front-capturing methods are the natural alternative to front-tracking methods. The interface is identified with a specific property of a mark function added to the model. Within this class of methods, the most widely used are the enthalpy method, the level set method and the phase field method.

Enthalpy methods, see [17] and Chapter 9 of [40], introduce an enthalpy or energy density function to measure the energy of the system. This function has a jump discontinuity at the interface given by the heat released (or absorbed) during the phase change. The heat equation is replaced by an energy balance equation which relates the enthalpy function with the heat distribution. The Stefan condition on the interface is therefore implicitly incorporated in the energy equation. The discretized system is highly nonlinear and for its solution advanced techniques [41] are required. The interface location is identified in a post-processing step as a corresponding isotherm of the heat distribution. However, 'stair-casing' effects on the temperature seem to be inherent to enthalpy methods, which may result in inaccurate approximations to the interface, especially near steady-state regimes. Furthermore, although this method has been successfully applied to solid-liquid phase transformations by Voller *et al.* [42, 43] and Nedjar [44], it has only recently been generalized to solid state phase transformations with a simple condition on the moving boundary, see Lam *et al.* [45] for further details.

The level set method, firstly introduced by Osher and Sethian [21], has already been generalized to many moving boundary problems [46, 47]. The interface is captured as the zero level set of the so-called level set function, and its motion is described by a hyperbolic equation for the level set function that is added to the governing set of equations. The velocity field used to update the level set function is quite different in the published applications of this method. Sussman *et al.* [48] use the fluid velocity to simulate incompressible two-phase flows. Chen *et al.* [22] use advection equations to extend the interface velocity onto the whole domain in a solidification problem, and Adalsteinsson and Sethian [49] use a procedure based on the fast marching methods to extend the front velocity in such a way that it does not destroy the distance function attribute of the level set function. In references [22, 49], the velocity field is only introduced for a numerical purpose.

The phase field method is widely used for phase transformation problems. The domain is parameterized by the phase field function which equals a fixed constant in each phase, and varies rapidly, but smoothly, between these two values in the interface region. The phase transformation occurs inside this interface region, whose thickness is an artificial parameter of the model. Several phase field models have been studied in the literature, although the most used are the Kobayashi potential [50] (see Wheeler *et al.* [25] for phase transitions in binary alloys), the Caginalp potential (see Caginalp and Socolovsky [51] for classical and modified Stefan problems) and the thermodynamically-consistent models [52]. The Kobayashi potential is based on a double-well potential with fixed minima at  $\pm 1$  coupled with a monotonically increasing function of the temperature. The Caginalp potential is based on a double-well potential measured by a parameter and a linear coupling with the temperature. The thermodynamically-consistent models are based on local positive entropy production. Fabbri and Voller [34] give a comparison of Caginalp and Kobayashi potentials for a one-dimensional solidification problem. Their numerical results show that, using the same thickness of the interfacial region, the solutions computed with the Kobayashi potential exhibit a closer agreement with the solution of the sharp interface problem than the Caginalp potential. Furthermore, a skilled asymptotic analysis of the phase field formulation is required in order to check whether the phase field solution converges to the sharp interface problem and to determine the physical value of the parameters that appear in the formulation of the model. This asymptotic analysis is already done in [32], where Caginalp proves the convergence to the Stefan and Hele-Shaw problems by taking the limit in the parameters in a suitable way.

The numerical solution of the phase field models is complex due to the steep gradients of the phase field function in the interfacial region. Fabbri and Voller [34] use a uniform fixed grid method which demands a fine grid resolution to capture the interface region. Lin [53] uses a coarse grid for

the heat equation and a fine grid for the phase field equation. These approaches imply excessive computational effort to solve the governing equations away of the interfacial region, where the phase field variable is constant. Furthermore, the use of such fine meshes makes these numerical methods unsuitable to solve higher-dimensional problems. More recent phase field computations [29, 30, 31] implement adaptive mesh techniques with fine resolutions in the interface region.

The aim of this chapter is to determine the numerical method that is most suitable for our scalar Stefan problem. A critical comparison of the moving grid, level set and phase field methods will be made. In order to apply the phase field method, our one-phase Stefan problem will be embedded in a two-phases Stefan problem, in which the thermal diffusivity of the artificial phase should be sufficiently small. Furthermore, we will limit ourselves to one-dimensional problems. This chapter is organized as follows. First, the governing equations for the one-phase and two-phases Stefan problems will be presented in Section 3.2. The numerical methods will be described in Section 3.3. Some numerical results will be given in Section 3.4 and the conclusions will be presented in Section 3.5.

## 3.2 The physical problem

In the present chapter, two classical Stefan problems are compared: the melting problem and the solid state phase transformation problem in binary metallic alloys. In the first problem, an interface separates a liquid phase and a solid phase. At the interface, the temperature is the melting temperature. This problem is also referred to as a solid-liquid transformation. In the second problem, a volume of constant composition is surrounded by a diffusive phase. In the interface between the particle and the diffusive phase a constant concentration is assumed. This problem is also referred to as a solid-solid transformation. Furthermore, the kinetic effects of surface tension and undercooling will be investigated for the melting problem.

The computational domain will be denoted by  $\Omega = [0, l]$ , where  $l$  denotes the length. This domain will be split into two phases, and the interface separating these phases will consist of only one point. Therefore, a function  $s : \mathbb{R}^+ \rightarrow [0, l]$  will assign each time  $t$  the position of the interface at this time  $s(t)$ .

### 3.2.1 A solid-liquid phase transformation

The computational domain  $\Omega$  is split into a liquid phase  $\Omega_{liq}(t) = [0, s(t))$  and a solid phase  $\Omega_{sol}(t) = (s(t), l]$ . The point separating both phases determines the position of the interface  $s(t)$ . The temperature in the point  $x$  at time  $t$  is denoted by  $u(x, t)$ .

The governing equations for this problem are

$$\frac{\partial u}{\partial t}(x, t) = \frac{1}{x^g} \frac{\partial}{\partial x} (K_{liq} x^g \frac{\partial u}{\partial x}), \quad x \in \Omega_{liq}(t), \quad t > 0, \quad (3.1a)$$

$$\frac{\partial u}{\partial t}(x, t) = \frac{1}{x^g} \frac{\partial}{\partial x} (K_{sol} x^g \frac{\partial u}{\partial x}), \quad x \in \Omega_{sol}(t), \quad t > 0, \quad (3.1b)$$

where  $g$  stands for the geometry of the problem (*i.e.*  $g = 0, 1, 2$  for planar, cylindrical and spherical symmetry respectively),  $K_{sol}$  and  $K_{liq}$  denote the thermal diffusivities in the solid and the liquid phases respectively, which involve heat capacity, density and heat conduction coefficients of the materials, which are assumed to be constant in time and position. At the interface, the temperature is the melting temperature

$$u(s(t), t) = u_m, \quad t > 0, \quad (3.2)$$

chosen here to be 0 without loss of generality, *i.e.*  $u_m = 0$ . Under these hypotheses, the velocity of the interface is given by the jump condition

$$L \frac{ds}{dt}(t) = K_{sol} \frac{\partial u}{\partial x}(x, t)|_{x \downarrow s(t)} - K_{liq} \frac{\partial u}{\partial x}(x, t)|_{x \uparrow s(t)}, \quad (3.3)$$

where  $L$  denotes the latent heat of solidification. Equation (3.3) is frequently called the Stefan condition. Furthermore, the domain is thermally insulated

$$\frac{\partial u}{\partial x}(x, t) = 0, \quad x \in \partial\Omega, \quad t > 0. \quad (3.4)$$

A piecewise constant initial temperature distribution

$$u(x, 0) = \begin{cases} u_{liq}, & \text{if } x \in \Omega_{liq} = [0, s_0), \\ 0, & \text{if } x = s_0, \\ u_{sol}, & \text{if } x \in \Omega_{sol} = (s_0, l], \end{cases}$$

is considered here, where  $u_{liq}$  and  $u_{sol}$  are constants, generally positive and negative respectively, and  $s_0$  denotes the initial position of the interface, *i.e.*,  $s_0 = s(0)$ .

### 3.2.2 A solid-solid phase transformation

The domain  $\Omega = [0, l]$  consists of a particle  $\Omega_{part}(t) = [0, s(t))$  and a diffusive phase  $\Omega_{dp}(t) = (s(t), l]$ . The point separating the particle and the diffusive phase represents the interface  $s(t)$ . The governing equations of this problem are the one-dimensional version of equations (2.1)-(2.5), with the appropriate symmetry specified by  $g$ .

If we take the thermal diffusivity of the liquid phase zero in the melting problem and assign  $L = c^{part} - c^{sol}$ , then equations (3.1a), (3.1b) and (3.3) are equivalent to (2.1), (2.2) and (2.5). The only difference between the two problems is the value prescribed at the interface, which merely implies a constant shift.

### 3.2.3 The modified Stefan problem

Due to the similarity in the governing equations, we only study the modified Stefan problem for the solid-liquid transformations. Surface tension and undercooling effects are incorporated in the model with the extended Gibbs-Thomson relation (3.5), which determines the interface temperature:

$$u(s(t), t) = -\frac{\sigma}{\Delta s}(\kappa(s(t)) + \nu \frac{ds}{dt}(t)), \quad (3.5)$$

where  $\sigma$  denotes the surface tension,  $\Delta s$  the entropy difference between phases,  $\kappa$  the curvature of the interface and  $\nu$  is identified with the kinetic undercooling coefficient. Note that a first order approximation of the exponential term in the Gibbs-Thomson relation has been used in Eq. (3.5).

## 3.3 Survey of numerical methods

For the sake of simplicity, the presentation of the numerical methods is restricted to the solid-liquid transformations. Generalization of these procedures to the solid-solid transformations is straightforward, except for the phase field model.

### 3.3.1 The moving grid method

The computational domain  $\Omega$  is divided in  $N$  intervals,  $r$  of those in the liquid phase  $\Omega_{liq}$  and  $N - r$  in the solid phase  $\Omega_{sol}$ . The grid is uniform in each phase and the interface is always located in the  $r^{th}$  mesh point. Due to the movement of the interface, the grid must be adapted each time step. Information is transferred to consecutive meshes by interpolation, see [17] for details. For higher-dimensional problems, however, interpolation is a time consuming process, that can be avoided by including the velocity of the mesh in the governing equations, by the so-called Arbitrary Lagrangian Eulerian (ALE) approach,

$$\frac{\partial u}{\partial t} - \frac{dx}{dt} \frac{\partial u}{\partial x} = \frac{1}{x^g} \frac{\partial}{\partial x} \left( K_{ph} x^g \frac{\partial u}{\partial x} \right), \quad (3.6)$$

where  $K_{ph} = K_{liq}$  in the liquid phase and  $K_{ph} = K_{sol}$  in the solid phase. This technique was used in [18, 19].

### 3.3.2 The level set method

In this section the main features of the level set method will be summarized. Reductions of the formulation will be made when possible. For a detailed presentation of this method the reader is referred to Chapter 4 of this thesis. The interface is captured as the zero level set of a continuous function  $\phi$ , the so-called level set function. Hence,  $\phi(x, t) = 0 \iff x = s(t), \forall t \geq 0$ . Furthermore,  $\phi$  is initialized as a signed distance function,

$$\phi(x, 0) = \begin{cases} -|x - s_0|, & \text{if } x < s_0, \\ 0 & \text{if } x = s_0, \\ |x - s_0| & \text{if } x > s_0, \end{cases}$$

which has been arbitrarily selected positive in the solid phase (resp. in the diffusive phase). The movement of the interface can be rewritten in terms of the level set function by taking the total time derivative of  $\phi$  at the interface position:

$$\frac{\partial \phi}{\partial t}(s(t), t) + \frac{ds}{dt}(t) \frac{\partial \phi}{\partial x}(s(t), t) = 0, \quad t > 0.$$

This local transport equation can be generalized onto  $\Omega$  if a continuous extension  $v$  of the front velocity  $ds/dt$  is provided, leading to

$$\frac{\partial \phi}{\partial t}(x, t) + v(x, t) \frac{\partial \phi}{\partial x}(x, t) = 0, \quad x \in \Omega, \quad t > 0. \quad (3.7)$$

Eq. (3.7) is discretized with a forward Euler scheme for the time integration and a first order upwind scheme in space. Therefore, the time stepping has to be chosen according to the stability condition:

$$\Delta t^n \leq \Delta t_{CFL} = \frac{\text{CFL} \Delta x}{|\max_{x \in \Omega} v(x, t^n)|},$$

where  $0 < \text{CFL} < 1$  and  $\Delta x$  denotes the mesh width, given by  $\Delta x = \frac{l}{N}$ . In order to prevent excessively large time steps, the following criteria is used:  $\Delta t^n = \min(\Delta t_{CFL}, \Delta t_{min})$ , where  $\Delta t_{min}$  is chosen proportional to  $\Delta x$ . The velocity field  $v$  is obtained by advection of the front velocity in the proper upwind direction [22]:

$$\frac{\partial v}{\partial \tau}(x, \tau) + S\left(\phi(x, t) \frac{\partial \phi}{\partial x}(x, t)\right) \frac{\partial v}{\partial x}(x, \tau) = 0, \quad (3.8)$$

where  $\tau > 0$  denotes a fictitious time not related to the physical time  $t$  and  $S$  denotes the sign function. This equation is degenerate in the interface location, and the initial condition  $v(x, 0) =$

$\frac{ds}{dt}(t)\mathbf{1}_{\{x=s(t)\}}(x)$  is prescribed, where  $\mathbf{1}$  denotes the characteristic function (*i.e.*  $\mathbf{1}_A(x) = 1$  if  $x \in A$ ,  $\mathbf{1}_A(x) = 0$  if  $x \notin A$ ).

After advecting the interface using Eq. (3.7), the level set function may not be a distance function at the new time step. Flat or steep gradients of  $\phi$  might appear, leading to inaccurate approximations in the neighborhood of the interface. In order to prevent them, the level set function is re-initialized to a signed distance function by solving in pseudo-time  $\tau$  the hyperbolic equation

$$\frac{\partial \tilde{\phi}}{\partial \tau}(x, \tau) = S(\phi(x, t)) \left( 1 - \left| \frac{\partial \tilde{\phi}}{\partial x}(x, \tau) \right| \right), \quad (3.9)$$

where the initial condition  $\tilde{\phi}(x, 0) = \phi(x, t)$  is used. Note that Eq. (3.9) is degenerate in the interface position, therefore it does not alter the interface location, *i.e.*  $\tilde{\phi}(x, \tau) = 0 \iff x = s(t) \forall \tau > 0$ , and  $\left| \frac{\partial \tilde{\phi}}{\partial x} \right| = 1$ , which characterizes distance functions, in the steady-state solution.

The heat equation (3.1) is discretized using a backward Euler scheme in time and central differences in space. For the nodes neighbouring to the interface, the second order derivatives of the quadratic Lagrangian interpolation polynomials that approximate the solution in the vicinity of the interface from the appropriate side of the interface are used. When a grid node changes phase (*i.e.* the interface crosses it) the discretization of the heat equation should be adapted. The node in question is not included in the discretization, and the solution at this node is obtained by interpolation from the neighboring nodes within the same phase. This procedure is slightly different from the method presented recently by Gibou *et al.* in [23], where the temperature in the conflictive node is adapted to the interface position before the heat equation is solved.

### 3.3.3 The phase field method

The phase field method uses a function  $\psi(x, t)$  which characterizes the phase of the system at each point  $x$  and time  $t$ . This function, the so-called phase field function, assumes an interface region of thickness  $\varepsilon$  where phase transitions occur. This clearly differs from the moving grid and level set methods where a sharp interface is considered. The phase field function  $\psi$  is defined as

$$\psi(x, t) = \begin{cases} 1 & \text{if } x \text{ is in the liquid phase at time } t, \\ -1 & \text{if } x \text{ is in the solid phase at time } t, \end{cases}$$

and  $-1 < \psi(x, t) < 1$  within the interface region. The evolution of the system is described by the following system of two coupled PDEs

$$\nu \xi^2 \frac{\partial \psi}{\partial t} = - \frac{\delta \mathcal{F}}{\delta \psi}, \quad (3.10a)$$

$$\frac{\partial u}{\partial t} + \frac{L}{2} \frac{\partial \psi}{\partial t} = \frac{1}{x^g} \frac{\partial}{\partial x} \left( K_{ph} x^g \frac{\partial u}{\partial x} \right), \quad (3.10b)$$

where  $K_{ph}$  is the appropriate diffusivity coefficient in each phase (*i.e.*,  $K_{ph} = K_{liq}$  where  $\psi = 1$  and  $K_{ph} = K_{sol}$  where  $\psi = -1$ ),  $L$  denotes the latent heat,  $\mathcal{F}$  denotes a free energy functional, which is a function of  $\psi$  as well as other variables of the problem, and  $\frac{\delta \mathcal{F}}{\delta \psi}$  denotes the variational derivative of  $\mathcal{F}$  with respect to  $\psi$ . The parameter  $\nu$  is a relaxation time and  $\xi$  is related to the microscopic physics and provides a length scale related to the thickness of the interface region.

The Caginalp model is used here. The free energy functional  $\mathcal{F}$  is expressed by

$$\mathcal{F}(\psi, u) = \int_{\Omega} \left[ \frac{1}{2} \xi^2 \left( \frac{\partial \psi}{\partial x} \right)^2 + f(\psi, u) \right] dx,$$

where  $f$  is the so-called free energy density, which consists of a double-well potential measured by a parameter  $a$  and a term coupling  $u$  with  $\psi$

$$f(\psi, u) = \frac{1}{8a}(\psi^2 - 1)^2 - 2u\psi.$$

The two minima of  $f$  establish the stable states of the problem (*i.e.* the liquid and solid states), which are slightly displaced from its physical values  $\phi = \pm 1$  due to the influence of the parameter  $a$ . Hence, to minimize its influence, the parameter  $a$  should be chosen small. Further, the interface thickness is given by the relation  $\varepsilon = \xi\sqrt{a}$ .

An adaptive mesh procedure is the most adequate to successfully resolve the interface region in a reasonable computing time, especially when higher-dimensional problems are to be solved. The computational mesh used here is based on an equidistribution principle [29]

$$\int_{x_i(t)}^{x_{i+1}(t)} M(\tilde{x}, t) d\tilde{x} = \frac{1}{N} \int_0^l M(\tilde{x}, t) d\tilde{x}, \quad \text{for } i = 0, 1, \dots, N-1, \quad (3.11)$$

where  $N$  is the number of space intervals considered in the spatial domain  $\Omega$  and  $M$  is a monitor function related with the thickness of the interfacial region. The choice of the function  $M$  should provide a measure of the computational error in the solution of the physical PDE, and can involve the mesh velocity [54]. In this case

$$M(x, t) = \gamma\beta(t) + \operatorname{sech}\left(\frac{x-s(t)}{2\varepsilon}\right), \quad \beta(t) = \int_0^l \operatorname{sech}\left(\frac{x-s(t)}{2\varepsilon}\right) dx, \quad (3.12)$$

where  $\gamma > 0$  is a parameter chosen by the user. The parameter  $\gamma$  must be chosen positive to ensure that the monitor function  $M$  is positive and not zero to avoid the clustering of all the mesh points inside the interface region. The number of mesh points placed within the interface region is approximately  $\frac{N}{1+\gamma}$ .

Finally, the use of the Caginalp potential and the adaptive mesh procedure leads to the following system of differential equations:

$$\nu\xi^2\left(\frac{\partial\psi}{\partial t} - \frac{dx}{dt}\frac{\partial\psi}{\partial x}\right) = \frac{\xi^2}{x^g}\frac{\partial}{\partial x}\left(x^g\frac{\partial\psi}{\partial x}\right) - \frac{1}{2a}(\psi^3 - \psi) + 2u, \quad (3.13a)$$

$$\frac{\partial u}{\partial t} - \frac{dx}{dt}\frac{\partial u}{\partial x} + \frac{L}{2}\left(\frac{\partial\psi}{\partial t} - \frac{dx}{dt}\frac{\partial\psi}{\partial x}\right) = \frac{1}{x^g}\frac{\partial}{\partial x}\left(K_{ph}x^g\frac{\partial u}{\partial x}\right), \quad (3.13b)$$

where the ALE approach has been used to incorporate the mesh movement into the governing equations. These equations are solved separately. Note that the mesh at the new time step is required to solve system (3.13). Hence, the following algorithm is used.

1. Estimate the interface position at time  $t^{n+1}$  by  $s_*^{n+1} = s^n + \Delta t^{n+1} \frac{s^n - s^{n-1}}{\Delta t^n}$ .
2. Compute the mesh at the new time step with Eqs. (3.11)-(3.12) using  $s_*^{n+1}$ .
3. Use the Newton method to solve the nonlinear phase field equation (3.13a), and compute the interface position  $s^{n+1}$  by linear interpolation of  $\psi^{n+1}$ .
4. Solve Eq. (3.13b) using  $\psi^{n+1}$  to find the temperature distribution at time  $t^{n+1}$ .

The interested reader is referred to [55] for a detailed presentation of the discretizations. The time-stepping condition  $\Delta t < 2\nu a \xi^2 = 2\nu \varepsilon^2$  was found sufficient [29] to ensure the existence of a numerical solution to Eq. (3.13a). This condition reveals the numerical difficulties that arise when the sharp interface problem has to be solved [32] (*i.e.*  $a, \xi, \varepsilon \rightarrow 0$ ). However, for the numerical test cases considered here, it has been possible to use a larger time step, although its selection is very sensitive to the other parameters in the phase field model.

## 3.4 Numerical results

### 3.4.1 Classical Stefan problems

Both the single-phase and the two-phases problems are solved in this section. The aim is to mimic the solid-solid phase transformation with a (super-heated) solid-liquid transformation. This approach has been chosen since the phase field method cannot be applied to the solid-solid phase transformation directly. Consider the computational domain  $\Omega = [0, 1]$  ( $l = 1$ ) in which a planar ( $g = 0$ ) solid-solid (resp. solid-liquid) phase transformation as presented in Section 3.2 takes place. The initial position of the interface is  $s_0 = 0.2$ .

#### Solid-solid phase transformation

The moving grid (MGM) and level set (LSM) methods are used to solve to compute the interface position and concentration profiles at each time step. The parameters for this test problem are presented in Table 3.1.

	Moving Grid	Level Set
$c^{part} = 0.53, c^0 = 0.1$	$N = 200$	$N = 200$
$c^{sol} = 0$	$\Delta t = 5 \times 10^{-4}$	$\Delta t_{min} = 5 \times 10^{-4}$
$D = 1$		CFL= 0.1

Table 3.1: Parameters for the solid-solid phase transformation test problem.

Figure 3.1(a) shows the evolution of the interface as a function of time, together with the similarity solution [55] defined in  $\Omega^{(s)} = \mathbb{R}$ . A good agreement between the numerical solutions and the similarity solution is observed at the beginning of the phase transformation. However, at the later stages the numerical solutions diverge from the similarity solution because of the boundedness of the computational domain  $\Omega$ . At the end of the simulation, the steady state solution is reached. Figure 3.1(b) shows the concentration history at  $x = 0.25$ . Excellent agreement with the similarity solution of both the moving grid and the level set methods is observed, with the only difference in the time at which the interface passes through  $x = 0.25$ .

The convergence to the similarity solution of both numerical models is examined in Table 3.2, with final time for the numerical integration  $t_{end} = 0.1$ . The time step was halved as  $N$  was doubled in both methods. First-order convergence is observed in both cases, with slightly higher accuracy for the moving grid method. This is probably due to the differences in the grid spacing and time steppings for both methods.

#### Solid-liquid phase transformation

The solid-solid phase transformation studied above is mimicked here with a solid-liquid transformation, in which the solid phase is super-heated. The parameters for this test problem are presented in Table 3.3. Note that  $L = c^{part} - c^{sol}$ ,  $K_{sol} = D$  and  $K_{liq} \rightarrow 0$ .



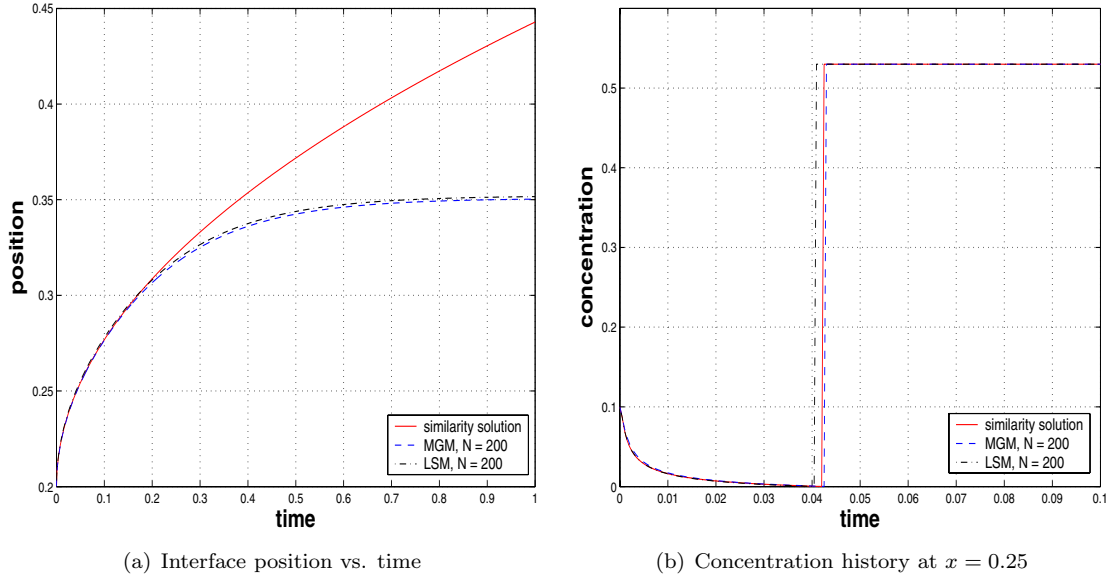


Figure 3.1: The solid-solid phase transformation problem with the moving grid (MGM) and the level set (LSM) methods.

$N$	Moving Grid Method		Level Set Method	
	$s_h(0.1)$	$\ s - s_h\ _\infty$	$s_h(0.1)$	$\ s - s_h\ _\infty$
100	0.276223	0.000727	0.278445	0.001630
200	0.276470	0.000495	0.277812	0.000997
400	0.276609	0.000343	0.277405	0.000594
800	0.276687	0.000239	0.277144	0.000342
1600	0.276730	0.000168	0.276988	0.000194

Table 3.2: Convergence behaviour of the moving grid and the level set methods for the solid-solid phase transformation problem. Similarity solution  $s(0.1) = 0.276815$ .

	Moving Grid	Level Set	Phase Field
$u_{liq} = 0.53, u_{sol} = 0.1$	$N = 200, r = 100$	$N = 200$	$N = 200, \gamma = 1$
$L = 0.53$	$\Delta t = 5 \times 10^{-4}$	$\Delta t_{min} = 5 \times 10^{-4}$	$\Delta t_{min} = 5 \times 10^{-4}$
$K_{liq} = 0.005, K_{sol} = 1$		CFL = 0.1	$\xi = 0.0001, a = 0.0625$

Table 3.3: Parameters for the solid-liquid phase transformation test problem.

Under these assumptions, the interface position in the similarity solution [55] is given by

$$s(t) = s_0 + 2\alpha\sqrt{t},$$

where  $\alpha$  is the solution to

$$\alpha = \frac{\sqrt{K_{sol}}}{\sqrt{\pi}L} \frac{u_{sol}}{\operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{sol}}}\right)} \exp\left(-\frac{\alpha^2}{K_{sol}}\right) + \frac{\sqrt{K_{liq}}}{\sqrt{\pi}L} \frac{u_{liq}}{2 - \operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{liq}}}\right)} \exp\left(-\frac{\alpha^2}{K_{liq}}\right).$$

Table 3.4 gives the values of  $\alpha$  when different values of  $K_{liq}$  are used. The similarity solution of the solid-solid phase transformation problem is recovered when  $K_{liq} = 0$ .

$K_{liq}$	0.05	0.01	0.005	0
$\alpha$	0.169082	0.127968	0.122595	0.121455

Table 3.4: Values of  $\alpha$  for different values of  $K_{liq}$ .

Figure 3.2(a) shows the evolution of the interface as a function of time for the moving grid (MGM), level set (LSM) and phase field (PFM) methods, together with the similarity solution. The initial temperature distribution was obtained from the similarity solution [55]

$$u^{(s)}(x, t) = \begin{cases} -\frac{u_{liq} \operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{liq}}}\right)}{2 - \operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{liq}}}\right)} + \frac{u_{liq} \operatorname{erfc}\left(\frac{x-s_0}{2\sqrt{K_{liq}t}}\right)}{2 - \operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{liq}}}\right)}, & \text{if } x < s(t), \\ u_{sol} - \frac{u_{sol} \operatorname{erfc}\left(\frac{x-s_0}{2\sqrt{K_{sol}t}}\right)}{\operatorname{erfc}\left(\frac{\alpha}{\sqrt{K_{sol}}}\right)}, & \text{if } x \geq s(t), \end{cases} \quad (3.14)$$

using an initial time  $t_0 = 0.01$ , *i.e.*  $u(x, 0) = u^{(s)}(x, t_0)$ , since it was seen that a discontinuous initial temperature distribution caused instabilities in the phase field solution. Comparison with the similarity solution is only valid for small times, and hence  $t_{end} = 0.25$  is used as final time. Figure 3.2(b) shows the temperature history at  $x = 0.25$ . The numerical solutions are consistent with the similarity solution (3.14). Note that the temperature history presented in Figure 3.2(b) resembles the concentration history presented in Figure 3.1(b), the only difference being in the temperature profile after the interface has passed  $x = 0.25$ , because of  $K_{liq} > 0$ . Note that decreasing  $K_{liq}$  will require an increase of  $t_0$  in order to obtain an initial temperature distribution that is sufficiently smooth inside the interface region. From the numerical simulations it has been observed that decreasing  $K_{liq}$  requires the reduction of the interface thickness  $\varepsilon$ , which implies reduction of the time-step too.

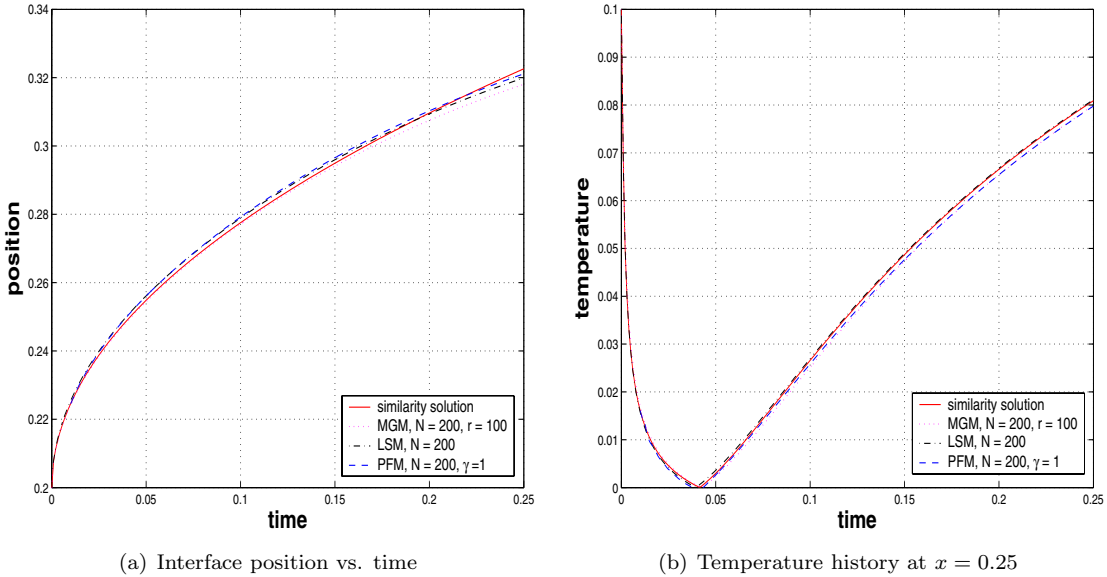


Figure 3.2: The solid-liquid phase transformation problem with the moving grid (MGM), the level set (LSM) and the phase field (PFM) methods.

Subsequently, we compare the mesh trajectories of the three numerical methods. In the moving grid method, Figure 3.3(a), the mesh is uniform on both sides of the interface. Since solidification

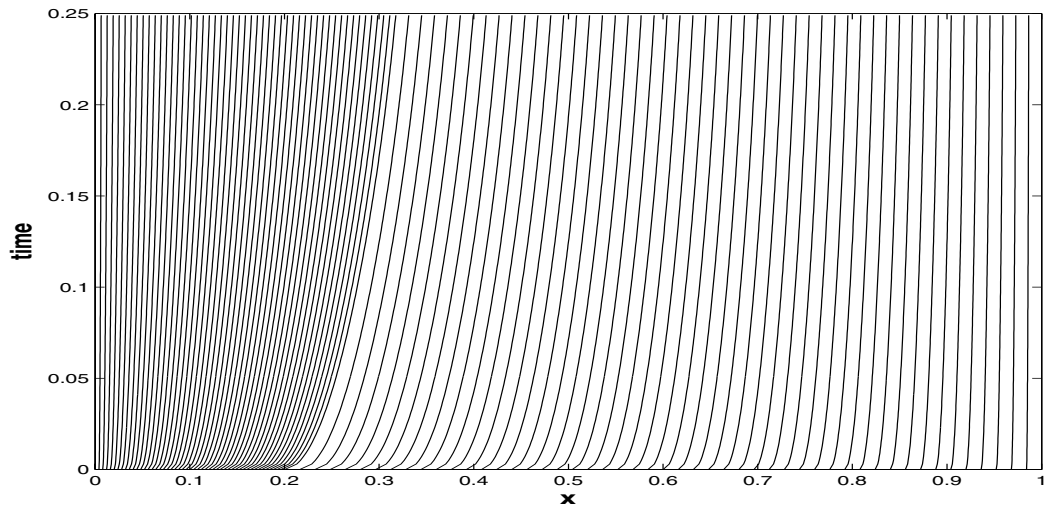
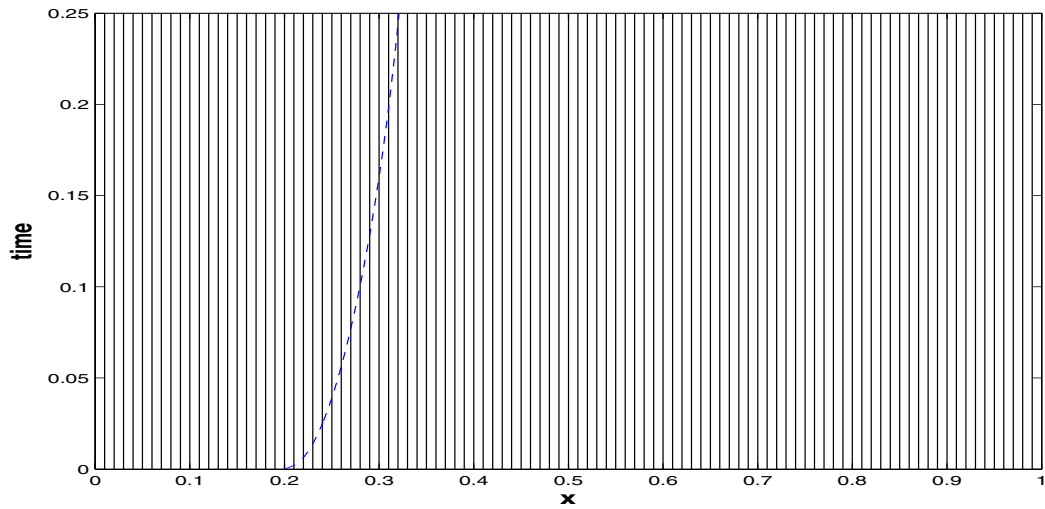
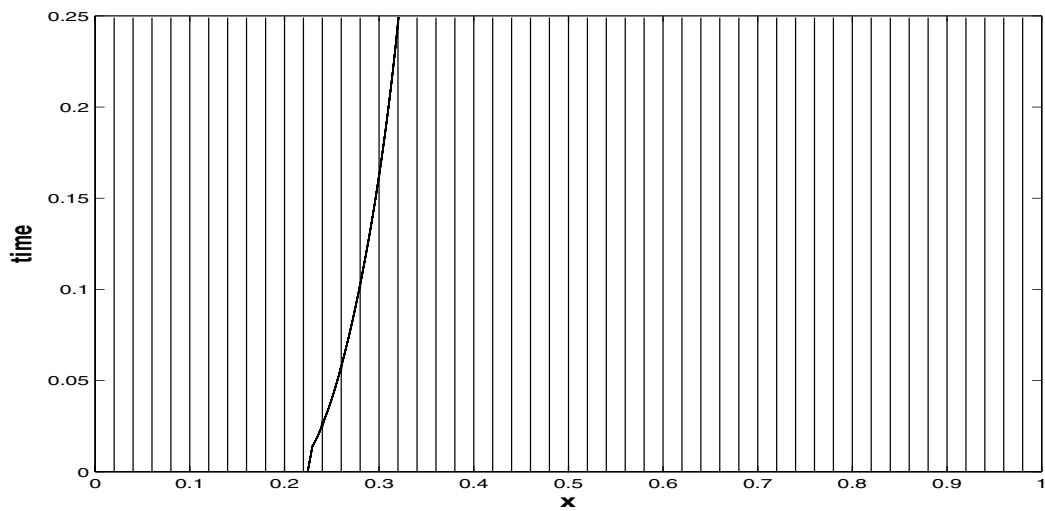
(a) Moving grid method with  $N = 100$  and  $r = 50$ .(b) Level set method with  $N = 100$ . The dashed line represents the moving interface.(c) Phase field method with  $N = 100$  and  $\gamma = 1$ .

Figure 3.3: Mesh trajectories of (a) the moving grid method, (b) the level set method, and (c) the phase field method.

takes place, the mesh width in the solid phase increases whereas the mesh width in the liquid phase decreases. The level set method, Figure 3.3(b), uses a uniform mesh that does not change. In the phase field method, Figure 3.3(c), half of the mesh points are located inside the interface region, of thickness  $\varepsilon = 2.5 \times 10^{-5}$ , since  $\gamma = 1$ .

### 3.4.2 Modified Stefan problems

In this section we will evaluate the performance of the numerical methods with two benchmark problems in which the interface temperature is defined by the modified Gibbs-Thomson relation Eq. (3.5). First, the critical radius of solidification as presented in Section VI of [51] will be studied. Next, the kinetic undercooling effects as presented in Section 4 of [34] will be investigated. In both cases, the phase field solutions will be compared with the 'sharp' interface solutions provided by the moving grid and the level set methods.

#### Curvature effects

In this section the critical radius of solidification [29, 51], which is the infimum of the radius for which the sphere will grow, is investigated. Consider a solid sphere of radius  $s_0 = 1.5$  in equilibrium with its melt. Hence, if  $\kappa_0 = \frac{2}{r_0}$  denotes the sum of its principal curvatures and both liquid and solid phases are at constant temperature

$$u_0 = -\frac{\sigma}{\Delta s} \kappa_0,$$

then the interface is in equilibrium. However, this equilibrium is unstable [51]. Consider the computational domain  $\Omega = [1, 2]$  and  $g = 2$  which mimics spherical symmetry, where the following Dirichlet boundary conditions are used:  $u(1, t) = u(2, t) = u_0$  for  $t > 0$ . In addition,  $u_0$  is used as initial temperature distribution. The remaining parameters of the problem are listed in Table 3.5. According to [32],  $\nu \rightarrow 0$  in the phase field formulation gives the sharp interface problem with the Gibbs-Thomson relation

$$u(s(t), t) = -\frac{\sigma}{\Delta s} \kappa(s(t)),$$

on the moving interface. For this purpose, we use  $\nu = 0.01$  in the phase field formulation, whereas  $\nu = 0$  is set in both the moving grid and level set formulations. A similar test problem (but with  $\nu = 1$ ) is solved in Section VI of [51] and in Section 5.4 of [29]. Figure 3.4(a) shows the evolution of the interface for  $s_0 = 1.499$ , which gives melting of the sphere. Figure 3.4(b) shows the evolution of the interface for  $s_0 = 1.501$ , which yields solidification. The solutions with the moving grid method (MGM) and the level set method (LSM) are indistinguishable. The solution with the phase field method is affected by  $\nu > 0$  and the diffusive interface, although a good agreement with the sharp interface models is found at early stages.

	Moving Grid	Level Set	Phase Field
$L = 0.1$	$N = 200, r = 100$	$N = 200$	$N = 100, \gamma = 1$
$K_{liq} = K_{sol} = 10$	$\nu = 0$	$\nu = 0$	$\nu = 0.01$
$\sigma = 0.15$	$\Delta t = 10^{-3}$	$\Delta t_{min} = 10^{-3}$	$\Delta t = 5 \times 10^{-4}$
$\Delta s = 4$		CFL= 0.25	$\xi = 0.045, a = 0.04$

Table 3.5: Parameters for the critical radius instability solidification test problem.

#### Undercooling effects

Consider a planar interface (*i.e.*  $g = 0$ ) on the domain  $\Omega = [0, 1]$ . The initial temperature distribution is given by  $u(x, 0) = u^{(s)}(x, t_0)$ , where

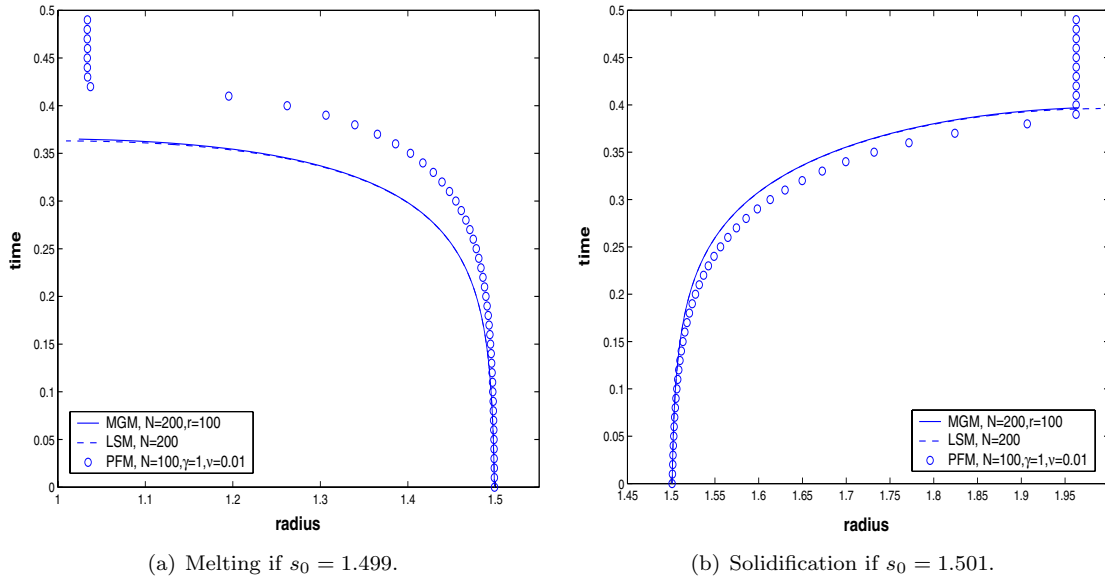


Figure 3.4: Evolution of the front for two different initial radius.

$$u^{(s)}(x, t) = \begin{cases} C_1 \frac{\operatorname{erf}(b/2) - \operatorname{erf}(x/(2\sqrt{t}))}{\operatorname{erf}(b/2)}, & \text{if } x \leq s(t), \\ C_2 \frac{\operatorname{erf}(b/2) - \operatorname{erf}(x/(2\sqrt{t}))}{1 - \operatorname{erf}(b/2)}, & \text{if } x > s(t), \end{cases} \quad (3.15)$$

$C_1 = -0.085$ ,  $C_2 = -0.015$ ,  $b = 0.396618$ ,  $t_0 = 0.15$  and  $s(t) = b\sqrt{t}$ . The initial interface position is given by  $s_0 = s(t_0)$ . Furthermore, the Dirichlet boundary conditions  $u(0, 1) = -0.085$  and  $u(1, t) = u^{(s)}(1, t + t_0)$  are used.

At time  $t_0$  the undercooling temperature  $u_0 < 0$  is prescribed on the moving interface. The undercooling coefficient  $\nu$  is given by

$$\nu = -\frac{u_0 \Delta s}{\sigma v_0},$$

where  $v_0 = \frac{b}{2\sqrt{t}}$ . Note that in this geometry the curvature  $\kappa$  vanishes. The remaining parameters of the problem are given in Table 3.6. A similar test problem was solved in Section 4 of [34] and in Section 5.2 of [29].

	Moving Grid	Level Set	Phase Field
$L = 1$	$N = 200, r = 100$	$N = 200$	$N = 100, \gamma = 1$
$\sigma = 0.00533$	$\Delta t = 10^{-3}$	$\Delta t_{min} = 10^{-3}$	$\Delta t = 10^{-3}$
$\Delta s = 4$		CFL = 0.25	$\xi = 0.002, a = 0.0625$
$K_{liq} = K_{sol} = 1$			

Table 3.6: Parameters for the undercooling test problem.

Figure 3.5 presents the evolution of the interface for various undercooling rates. When the initial undercooling temperature is small,  $u_0 = -0.001$ , the problem resembles a classical Stefan problem. There is hardly any difference between the numerical solutions computed with the

moving grid (MGM), level set (LSM) and phase field (PFM) methods, as shown in Figure 3.5(a). However, for more severe undercooling rates, only the phase field method is capable of handling the movement of the interface, see Figure 3.5(b). Instabilities were observed in the beginning of the simulation with both the moving grid and level set methods. Providing an initial temperature distribution such that the gradient of the temperature becomes smooth in a neighbourhood of the interface improves the performance of both methods.

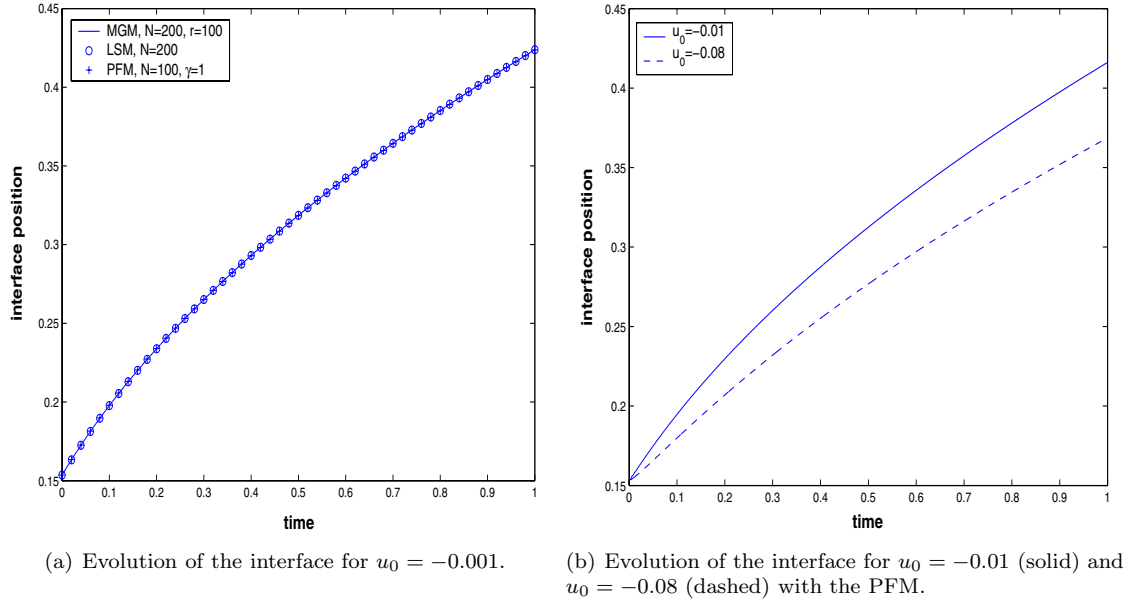


Figure 3.5: Evolution of the interface for various undercooling rates.

### 3.5 Conclusions

Two Stefan problems resulting from phase transformations have been considered in this chapter: the melting problem and a diffusional phase transformation in binary alloys where only in one of the two phases the solution of the diffusion equation is determined, whereas the solution is constant in the other phase. Numerical solutions of those problems have been obtained with the moving grid, the level set and the phase field methods.

The formulation of the problems and the existing similarity solutions show the resemblance between the two problems. In fact, the diffusional one phase transformation is a special case of the melting problem. From the numerical computations several conclusions can be obtained:

- Both the moving grid and the level set methods are suitable numerical models for the solid state phase transformation, and their accuracy is comparable.
- The same can be concluded for the melting problem. Furthermore, for this problem the phase field method is also suitable. From the reported results (Figure 3.2) one might conclude that the phase field method gives a better approximation of the interface position and temperature profile. However, the grid resolution within the interfacial region is much higher with the phase field method than with the moving grid and the level set methods, which leads to greater computational cost.
- The phase field method is found to be applicable for the melting problem, even when the difference of the thermal properties in each phase is appreciable. Unfortunately, it is hard

to derive the appropriate phase field parameter values from the physical parameters of the phases involved, which limits the true predictive power of this technique. The moving grid and the level set methods do not suffer from this high dependence on the physical parameters of the problem. Their input values can be obtained rather easily from tables and phase-diagrams.

- Surface tension and undercooling effects can be easily introduced in the model, using the (extended) Gibbs-Thomson relation. The phase field model, due to the diffusive interface, avoids explicit computation of the curvature and the interface temperature. Furthermore, it is capable of handling large undercooling rates, which is a severe test for the moving grid and level set methods.

The interpolative approach presented for the moving grid method can be replaced by the ALE approach, which introduces the displacement of the grid into the governing equations, leading to solving a convection-diffusion equation, which is more convenient for higher-dimensional problems. Topological changes, which involve merging or breaking of the interface, are difficult to model with the moving grid method. However, this is easily handled in the level set method and this will be a topic in Chapter 4. The difficulties in the level set method are the extension of the front velocity and the reinitialization, although these difficulties can be overcome without much effort. Hence, the level set method is found to be the most adequate method to tackle higher-dimensional Stefan problems.





---

---

# CHAPTER 4

---

## A Level Set Method for higher-dimensional scalar Stefan problems: Particle dissolution in binary alloys

### 4.1 Introduction

In the present chapter the numerical simulation of particle dissolution in binary alloys is discussed. The mathematical model used to describe the physical problem is based on long-distance diffusion [2, 3, 4], in which the concentration of the solute at the interface between the adjacent phases is the solid solubility derived from the binary phase diagram (see Section 2.2). The long-distance diffusion models imply that the processes at the interface proceed infinitely fast.

The level set method, first introduced by Osher and Sethian [21], is applied here to determine the interface location. This method allows topological changes, such as breaking up and merging of interfaces, in a natural way. Furthermore, numerical computation of geometrical quantities (as for instance the local curvature of the interface) is done easily. The moving interface is identified with the zero level set (a set of points, curves or surfaces in one-, two- and three-dimensional problems respectively) of a continuous function, the so-called level set function, and denoted by  $\phi$ . The front velocity, only defined at the front position, has to be (continuously) extended into the computational domain in order to advect the interface. Furthermore, in most applications of this method, the level set function is unavoidably maintained as a signed distance function, at least in a band around the interface, during the complete computation. This additional constraint demands re-setting the values of the level set function without altering its zero level set (*i.e.* the interface location). This process is referred to as reinitialization.

Several techniques have been developed to extend information from the interface position. Chen *et al.* [22] use a set of advection equations to define a continuous extension of the front velocity. Information is transported off the interface location in each Cartesian direction independently. Osher and Fedkiw propose in Chapter 8 of [47] an equivalent advection equation to extend a general scalar quantity  $q$  defined on the interface outwards in the normal direction. Kim *et al.* [24] use a constant extrapolation in the normal direction to extend the front velocity. Sethian and Adalsteinson [46, 49, 56] develop a numerical strategy based on the fast marching method for the same purpose. Both approaches due to Kim *et al.* and Sethian and Adalsteinson solve the steady-state of the equation proposed by Osher and Fedkiw. However, by construction, these

methods [24, 46, 49, 56] maintain the level set function a distance function after updating the interface, and so avoid reinitialization, except at the 'center' of the region enclosed by the interface for expanding (*i.e.* growing) interfaces where a flat profile will appear.

In order to reinitialize the level set function, Susmann *et al.* [48] solve a nonlinear hyperbolic equation which converges to a signed distance function in the stationary regime. In this equation, the information is transported from the interface in the normal direction with speed  $S(\phi)$ , where  $\phi$  denotes the level set function and  $S$  the sign function or a smooth representation of it. Russo *et al.* [57] use a second order accurate approximation of a distance function as a Dirichlet condition in the mesh points adjacent to the interface together with a truly upwind scheme in Susmann's reinitialization equation, in order to reduce the shift of the interface location due to the reinitialization. Peng *et al.* [58] propose a modification of Susmann's equation in such a way that the sign of the level set is never changed during reinitialization. Hence, the displacement of the interface due to the reinitialization is upper bounded by the mesh size. All these approaches have a complexity of  $\mathcal{O}(N^4)$ , where  $N$  denotes the number of mesh points in each direction. This complexity can be reduced to  $\mathcal{O}(kN^3)$  if a total of  $k$  pseudo-time iterations are used.

The fast marching method [46] is used to solve the Eikonal equation, which is the steady-state solution of the hyperbolic equations solved in [48, 57, 58]. The solution in the regions of  $\phi > 0$  and  $\phi < 0$  is computed separately. Information propagates from the interface into the downwind direction. The point of smallest  $\phi$  (resp. smallest  $|\phi|$  for the region of  $\phi < 0$ ) is efficiently computed by means of the heap sort algorithm. In this way, the complexity of the reinitialization is reduced to  $\mathcal{O}(N^3 \log N)$ . The fast sweeping method [59, 60, 61] represents an improvement to the fast marching method. Characteristics having the same direction are covered within the same sweep. Hence  $2^n$  sweeps (where  $n$  denotes the spatial dimension of the problem) are required to cover all possible directions. The complexity of this method is then  $\mathcal{O}(N^3)$ . Both fast marching and fast sweeping methods rely on accurate approximations of the solution on the nodes adjacent to the interface, since both methods solve a boundary value problem. In opinion of the author, these approximations might become arduous to achieve if the geometry of the interface is complicated. However, the approaches described in [48, 58] avoid this, in spite of suffering of eventual shift of the interface location.

Chessa *et al.* [62] use an enriched finite element method to solve a solidification problem. This method is able to deal with discontinuities across the interface on the temperature gradients, and requires an 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.

The numerical approach employed here uses a combination of finite difference and finite element methods. Finite difference schemes are used to solve the hyperbolic equations inherited from the level set formulation. These equations correspond to the advection of the interface, extension of the front velocity and reinitialization of the level set. Finite elements are used to solve the diffusion problem Eqs. (2.2), (2.3) and (2.4). The cut-cell approach is a novel technique used here to adapt the finite element triangulation according to the interface position at each time step, and hence to discretize Eq. (2.3) correctly. Interpolation between the Cartesian mesh and the finite element triangulation is not necessary, since both meshes are based on the same mesh points. A similar combination has been presented recently by Tan and Zabaras in [63] for the simulation of dendritic growth. The following algorithm is used in this chapter:

1. Initialize the concentration  $c$  and the level set function  $\phi$  according to the initial conditions of the problem. Set  $t = 0$ .
2. From the concentration and level set (*i.e.* interface position) at time  $t$ , compute a continuous extension  $\mathbf{v}$  of the front velocity  $v_n$  and determine the time-step  $\Delta t$ .
3. Update the level set function with  $\mathbf{v}$  and  $\Delta t$ .

4. If necessary, reinitialize the level set function to a signed distance function.
5. Use the cut-cell approach to adapt the underlying triangulation to the interface position at time  $t + \Delta t$ .
6. Solve the diffusion problem inside the diffusive phase.
7. Update time  $t = t + \Delta t$ .
8. If the termination time is not reached, go back to 2.

In the following sections, the details of this algorithm are given. For the sake of presentation, only two-dimensional problems are considered. The extension of the schemes and ideas to three-dimensional problems is straightforward, and hence not described. Only the presentation of the cut-cell method will be extended to three-dimensional problems.

This chapter is organized as follows. The level set method is described in detail in Section 4.2. The extension of the front velocity is studied in Section 4.2.1 and the reinitialization of the level set function in Section 4.2.2. Next, the solution to the diffusion problem is presented in Section 4.3, and the cut-cell method is described in detail in Section 4.4. A number of numerical results are given in Section 4.5 and the conclusions are formulated in Section 4.6.

## 4.2 The Level Set Method

In the level set method, a scalar continuous function  $\phi$  is introduced into the governing equations in order to implicitly capture the moving interface  $\Gamma$  as its zero level set:

$$\phi(\mathbf{x}, t) = 0 \iff \mathbf{x} \in \Gamma(t), \quad t \geq 0.$$

The scalar indicator function  $\phi$  is commonly referred to as the level set function or simply level set. The level set method was first introduced by Osher and Sethian [21] and has been successfully applied to two-phase Stefan problems (*i.e.* solidification problems) by Chen *et al.* [22] and Osher *et al.* [64]. 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. (2.5) for scalar Stefan problems. The above equation is only valid at the interface, because the velocity  $v_n$  is only defined at the front position itself. If the front velocity is continuously extended over  $\Omega$  leading to the vector field  $\mathbf{v}$ , then the above equation can be generalized to an hyperbolic equation for the level set function

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, t) + \mathbf{v}(\mathbf{x}, t) \cdot \nabla \phi(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \Omega, \quad t > 0. \quad (4.1)$$

The numerical approximation of Eq. (4.1) is done in a Cartesian grid. The time integration is carried out with the forward Euler method, whereas the space derivatives are discretized with a first order accurate upwind scheme. This scheme leads to a CFL stability condition on the time step:

$$\Delta t \max \left( \frac{|v_1|}{\Delta x_1} + \frac{|v_2|}{\Delta x_2} \right) < 1. \quad (4.2)$$

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

$$\mathbf{n} = \frac{\nabla\phi}{\|\nabla\phi\|}, \quad (4.3)$$

which points into the region of  $\phi > 0$ . Likewise, the mean curvature of the interface is defined by

$$\kappa = \nabla \cdot \mathbf{n}, \quad (4.4)$$

so that  $\kappa > 0$  for convex regions,  $\kappa < 0$  for concave regions and  $\kappa = 0$  for a plane. Further, if  $\phi$  is a signed distance function (*i.e.*  $\|\nabla\phi\| = 1$ ), then  $\kappa = \Delta\phi$ , where  $\Delta$  represents the Laplace operator.

### 4.2.1 Extension of the front velocity

The front velocity  $v_n$  is only defined at the interface location, and an artificial continuous extension of it onto  $\Omega$  is required. Two different extension procedures are presented below in detail. Both approaches require the solution of hyperbolic equations, which define a velocity field in the whole domain  $\Omega$ . In the first approach, the Cartesian components of the front velocity are decoupled during the extension [22]. In the second approach, the normal velocity is extrapolated in the normal direction [47].

#### Extension in the Cartesian directions

Considering the Stefan condition Eq. (2.5), and the fact that the concentration equals a constant at the interface Eq. (2.3), the normal velocity vector  $\mathbf{v} = v_n \mathbf{n}$  of the interface can be rewritten as

$$\mathbf{v}(\mathbf{x}, t) = \frac{D}{c^{part} - c^{sol}} \nabla c(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), t > 0.$$

An extension of the front velocity might be obtained by solving separately for each Cartesian direction

$$\begin{cases} \frac{\partial v_k}{\partial \tau} + S \left( \phi \frac{\partial \phi}{\partial x_k} \right) \frac{\partial v_k}{\partial x_k} = 0, \\ v_k(\mathbf{x}, 0) = \frac{D}{c^{part} - c^{sol}} \frac{\partial c}{\partial x_k}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t), \end{cases} \quad \text{for } k = 1, 2, \quad (4.5)$$

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 (4.5) point away from the interface location, which implies that a Dirichlet condition is needed on  $\Gamma(t)$  only.

Therefore, a suitable implementation of the interface conditions is required. Let  $\mathbf{x}_{ij} = (x_{1,i}, x_{2,j})^t$  be a node in the diffusive phase which has one of its left/right neighbors inside the particle. Such a node is marked with a vertical black rectangle in Figure 4.1(a). Then  $v_{1,ij}$  is computed by the following discretization

$$v_{1,ij} = \frac{D}{c^{part} - c^{sol}} \frac{c_{ij} - c^{sol}}{d_{1,ij}},$$

where

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

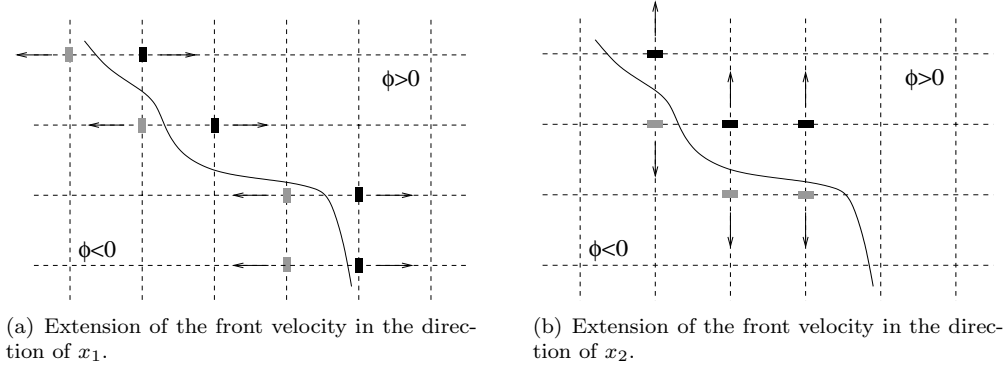


Figure 4.1: Extension of the front velocity in the Cartesian directions: neighbours inside the diffusive phase marked with black rectangles, and neighbours inside the particle marked with grey rectangles. Information travels off the interface.

is an approximation of the signed 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 given to the left/right neighbor that lies inside the particle, denoted with a gray vertical rectangle in Figure 4.1(a). These values replace the Dirichlet boundary condition on the interface in the discretized version of Eq. (4.5) with  $k = 1$ . The velocity  $v_2$  in the adjacent nodes to the interface, *i.e.*  $\phi_{ij-1} < 0 < \phi_{ij}$  or  $\phi_{ij} > 0 > \phi_{ij+1}$ , marked with horizontal rectangles in Figure 4.1(b), is obtained analogously.

To solve Eqs. (4.5) numerically, the forward Euler method and first order upwind discretizations are used. The CFL condition for stability yields  $\Delta\tau < \Delta x_1$  (resp.  $\Delta\tau < \Delta x_2$  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  $\Gamma(t)$ , only a small number of pseudo-time steps is realized. Thus,  $\Delta\tau = 0.8\Delta x_i$  and 10 pseudo-time-steps (for  $i = 1, 2$ ) are used in the simulations presented in Section 4.5.

### Extension in the normal direction

Extension of the front velocity in the normal direction is an alternative. If  $\phi$  is a distance function, the normal velocity  $v_n = \frac{D}{c^{part} - c^{sol}} \frac{\partial c}{\partial \mathbf{n}}$  can be calculated with

$$v_{n,ij} = \frac{D}{c^{part} - c^{sol}} S \left( \frac{c^0 - c^{sol}}{c^{part} - c^{sol}} \right) \frac{|c_{ij} - c^{sol}|}{\phi_{ij}}, \quad (4.6)$$

for the nodes  $\mathbf{x}_{ij}$  within the diffusive phase. Note that  $S \left( \frac{c^0 - c^{sol}}{c^{part} - c^{sol}} \right)$  has been included to give the correct sign to  $v_{n,ij}$ . To extend this velocity into the particle domain the following equation is solved:

$$v_{n,\tau} + \min(S(\phi), 0) \mathbf{n} \cdot \nabla v_n = 0, \quad \mathbf{x} \in \Omega, \quad \tau > 0, \quad (4.7)$$

where  $\tau$  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  $\phi$ . The CFL condition is now given by  $\Delta\tau \max \left( \frac{|n_1|}{\Delta x_1} + \frac{|n_2|}{\Delta x_2} \right) < 1$ , 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. Thus,  $\Delta\tau = 0.8 \left( \frac{1}{\Delta x_1} + \frac{1}{\Delta x_2} \right)^{-1}$  and 10 pseudo-time-steps are used in Section 4.5.

## 4.2.2 Reinitialization

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

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

After advecting the interface using Eq. (4.1), the level set function is in general not a distance function at the new time step. It might lead to flat/steep gradients of  $\phi$ , resulting in inaccurate approximations of the normal velocity  $v_n$ . However, the level set function can be reinitialized by solving in pseudo-time  $\tau$  the hyperbolic equation

$$\frac{\partial \varphi}{\partial \tau} = S(\varphi)(1 - \|\nabla \varphi\|), \quad (4.8)$$

where  $S$  denotes the smooth sign function

$$S(\varphi) = \frac{\varphi}{\sqrt{\varphi^2 + \|\nabla \varphi\|^2 \Delta x^2}}, \quad (4.9)$$

and  $\varphi$  the reinitialized level set function. It is initialized by  $\varphi(\mathbf{x}, 0) = \phi(\mathbf{x}, t)$  as the level set function at time  $t$ . Note that the zero level set of  $\varphi(\mathbf{x}, \tau)$  is the same as that of  $\phi(\mathbf{x}, t)$ , and the steady-state solution of (4.8) is given by  $\psi$  satisfying  $\|\nabla \psi\| = 1$ , which is a characteristic of a distance function. After Eq. (4.8) is solved,  $\phi = \psi$ .

The monotone Godunov upwind scheme Eq. (4.10) is used in the numerical solution of (4.8). The term  $G(\varphi) := 1 - \|\nabla \varphi\|$  is approximated by (see [58])

$$G(\varphi_{ij}) = \begin{cases} 1 - \sqrt{\max(a_+^2, b_-^2) + \min(c_+^2, d_-^2)}, & \text{if } \varphi_{ij} > 0, \\ 1 - \sqrt{\max(a_-^2, b_+^2) + \min(c_-^2, d_+^2)}, & \text{if } \varphi_{ij} < 0, \\ 0, & \text{otherwise,} \end{cases} \quad (4.10)$$

where  $a = D_{x_1}^- \varphi_{ij}$ ,  $b = D_{x_1}^+ \varphi_{ij}$ ,  $c = D_{x_2}^- \varphi_{ij}$  and  $d = D_{x_2}^+ \varphi_{ij}$  are used to denote the left/right sided derivatives of  $\varphi$  with respect to  $x_1$  and  $x_2$  respectively,  $a_- = \min(a, 0)$  and  $a_+ = \max(a, 0)$ . The same definitions apply for  $b_{\pm}$ ,  $c_{\pm}$  and  $d_{\pm}$ . The time integration is done by a third order accurate TVD Runge-Kutta scheme [58], and the space derivatives are approximated by a fifth order WENO scheme [65]. In this way, the order of the overall accuracy of the numerical solution of the complete Stefan problem is not adversely affected because of discretization errors in the reinitialization problem.

The reinitialization procedure is terminated when  $|S(\varphi_{ij})G(\varphi_{ij})|$  is smaller than a prescribed tolerance in a band around the interface location. The smooth sign function is used in the termination criterion to smear out large errors occurring in singularity points. The same criterion is 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. If the movement of the interface is fast or leads to the breaking of the interface, more iterations are required to satisfy 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.

## 4.3 The diffusion problem

The diffusion equation (2.2) is solved with a standard Galerkin finite element method using linear elements. The finite element mesh consists of triangles and tetrahedra for two- and three-dimensional problems respectively. The underlying triangulation must be adapted to the interface position each time step in order to apply the interface condition Eq. (2.3). This is done with

the cut-cell approach, which is presented in detail in Section 4.4. A temporary triangulation is created as a result of the cut-cell method, with a set of additional nodes on the interface.

Let  $\mathbf{x}_{kl}$  denote a general mesh point of the triangulation located either on the interface, *i.e.* obtained with the cut-cell method, or inside the diffusive phase. The concentration  $c$  inside the diffusive phase  $\Omega_{dp}$  is approximated by a linear combination of the so-called basis functions  $\psi_{kl}$

$$c(\mathbf{x}, t) = \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) > 0}} c_{kl}(t) \psi_{kl}(\mathbf{x}) + \tilde{c}(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_{dp}(t) \cup \Gamma(t), \quad t > 0, \quad (4.11)$$

where  $c_{kl}(t)$  are to be determined at each time step and  $\tilde{c}$  deals with the conditions at the interface. The basis functions  $\psi_{kl}$  are piecewise linear functions which satisfy

$$\psi_{kl}(\mathbf{x}_{ij}) = \begin{cases} 1, & \text{if } i = k \text{ and } j = l, \\ 0, & \text{otherwise.} \end{cases}$$

The interface condition Eq. (2.3) is satisfied if

$$\psi_{kl}(\mathbf{x}) = 0, \quad \mathbf{x} \in \Gamma(t), \quad t > 0,$$

for  $k$  and  $l$  such that  $\phi(\mathbf{x}_{kl}, t) \geq 0$ , and

$$\tilde{c}(\mathbf{x}, t) = c^{sol}, \quad \mathbf{x} \in \Gamma(t), \quad t > 0,$$

are imposed. Therefore, the diffusion equation is discretized as follows

$$\begin{aligned} \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) > 0}} \frac{dc_{kl}}{dt}(t) \int_{\Omega} \psi_{kl} \psi_{ij} d\Omega + \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) > 0}} c_{kl}(t) \int_{\Omega} D \nabla \psi_{kl} \cdot \nabla \psi_{ij} d\Omega = \\ - \int_{\Omega} \frac{\partial \tilde{c}}{\partial t} \psi_{ij} d\Omega - \int_{\Omega} D \nabla \tilde{c} \cdot \nabla \psi_{ij} d\Omega, \end{aligned} \quad (4.12)$$

for all indexes  $i, j$  such that  $\phi(\mathbf{x}_{ij}, t) \geq 0$ . Note that, in this case, one can choose

$$\tilde{c}(\mathbf{x}, t) = \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) = 0}} c^{sol} \psi_{kl}(\mathbf{x})$$

without loss of generality. Then, equation (4.12) yields

$$\begin{aligned} \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) > 0}} \frac{dc_{kl}}{dt}(t) \int_{\Omega} \psi_{kl} \psi_{ij} d\Omega + \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) > 0}} c_{kl}(t) \int_{\Omega} D \nabla \psi_{kl} \cdot \nabla \psi_{ij} d\Omega = \\ - \sum_{\substack{k,l \\ \phi(\mathbf{x}_{kl}, t) = 0}} c^{sol} \int_{\Omega} D \nabla \psi_{kl} \cdot \nabla \psi_{ij} d\Omega, \end{aligned} \quad (4.13)$$

for all indexes  $i, j$  such that  $\phi(\mathbf{x}_{ij}, t) \geq 0$ . The time integration is performed with a backward Euler method, which finally leads to

$$\begin{aligned} \sum_{\substack{k,l \\ \phi_{kl}^{m+1} > 0}} \frac{c_{kl}^{m+1} - c_{kl}^m}{\Delta t^m} \int_{\Omega} \psi_{kl} \psi_{ij} d\Omega + \sum_{\substack{k,l \\ \phi_{kl}^{m+1} > 0}} c_{kl}^{m+1} \int_{\Omega} D \nabla \psi_{kl} \cdot \nabla \psi_{ij} d\Omega = \\ - \sum_{\substack{k,l \\ \phi_{kl}^{m+1} = 0}} c^{sol} \int_{\Omega} D \nabla \psi_{kl} \cdot \nabla \psi_{ij} d\Omega, \end{aligned} \quad (4.14)$$

for all indexes  $i, j$  such that  $\phi_{ij}^{m+1} \geq 0$ , where the superscript  $m$  denotes the time level.

It is important to remark that the time-stepping and concentration has to be adapted locally in dissolution problems whenever the interface crosses a node. To illustrate this point, let  $\mathbf{x}_{kl}$  be a node which was inside the particle at time  $t^m$  and lies in the diffusive phase at time  $t^{m+1}$ , hence  $\phi_{kl}^m < 0 < \phi_{kl}^{m+1}$ . Note that there is a jump of the concentration from  $c^{part}$  to  $c^{sol}$  in such a node. The time  $t_{kl}^*$  at which the interface was exactly at the node  $\mathbf{x}_{kl}$  can be approximated by linear interpolation of the level set function:

$$t_{kl}^* = t^m - \phi_{kl}^m \frac{t^{m+1} - t^m}{\phi_{kl}^{m+1} - \phi_{kl}^m}.$$

Hence, the time derivative in equation (4.14) is replaced by

$$\sum_{\substack{k,l \\ \phi_{kl}^{m+1} > 0}} \frac{c_{kl}^{m+1} - \hat{c}_{kl}^m}{\Delta \hat{t}_{kl}^{m+1}} \int_{\Omega} \psi_{kl} \psi_{ij} d\Omega,$$

where

$$\hat{c}_{kl}^m = \begin{cases} c_{kl}^m, & \text{if } \phi_{kl}^m \geq 0 \text{ and } \phi_{kl}^{m+1} > 0, \\ c^{sol}, & \text{if } \phi_{kl}^m < 0 \text{ and } \phi_{kl}^{m+1} > 0, \end{cases}$$

and

$$\Delta \hat{t}_{kl}^{m+1} = \begin{cases} \Delta t^m, & \text{if } \phi_{kl}^m \geq 0 \text{ and } \phi_{kl}^{m+1} > 0, \\ t^{m+1} - t_{kl}^*, & \text{if } \phi_{kl}^m < 0 \text{ and } \phi_{kl}^{m+1} > 0. \end{cases}$$

The discretized system (4.14) forms a linear system of equations  $\mathbf{M}\mathbf{c}' + \mathbf{S}\mathbf{c} = \mathbf{F}$ , with  $\mathbf{M}$  the mass matrix,  $\mathbf{S}$  the stiffness matrix and  $\mathbf{F}$  stands for the terms in the right hand side of (4.14).

## 4.4 The cut-cell approach

The cut-cell approach is used to adapt the basic triangulation to the interface position at each time step. Only linear elements (*i.e.* triangles and tetrahedra for two- and three-dimensional problems respectively) are considered here. In each vertex a value of the level set function is given. If in an element the level set function changes sign, then this 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, there may be either zero or two intersections. In a tetrahedron, there may be zero, three or four intersections. Each intersected element is subdivided into subelements, based on the number of intersection points on the edges. Hence, a triangle with two intersection points is subdivided into three new triangles, see Figure 4.2. In 3D the situation is somewhat more complicated than in 2D. The subdivision is still based on the subdivision of edges. The subdivision of faces must be done in a unique way, otherwise neighbouring elements do not match. This problem only occurs if two edges of a face are



intersected. In this work, the subdivision of faces is based on the node numbering. However, other criteria are also possible. Then, when a face is subdivided, it is marked as subdivided according to a certain rule. Once the faces are subdivided, the subdivision of the tetrahedron is simple. Thus, a tetrahedron with three intersection points is subdivided into four new tetrahedra, see Figure 4.3, whereas a tetrahedron with four intersection points is subdivided into six new tetrahedra, see Figure 4.4.

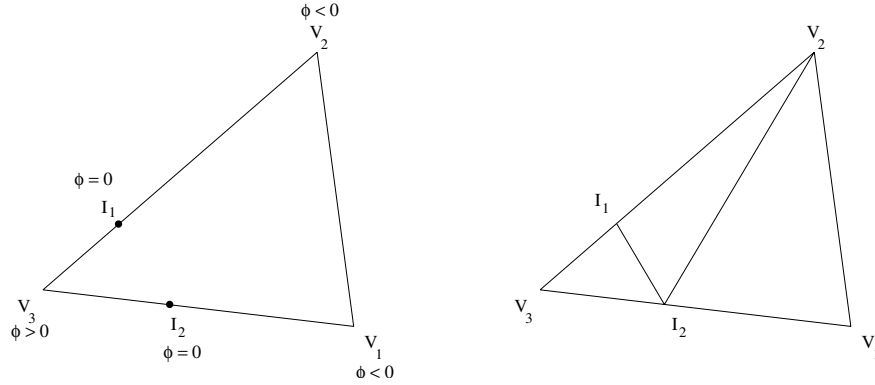
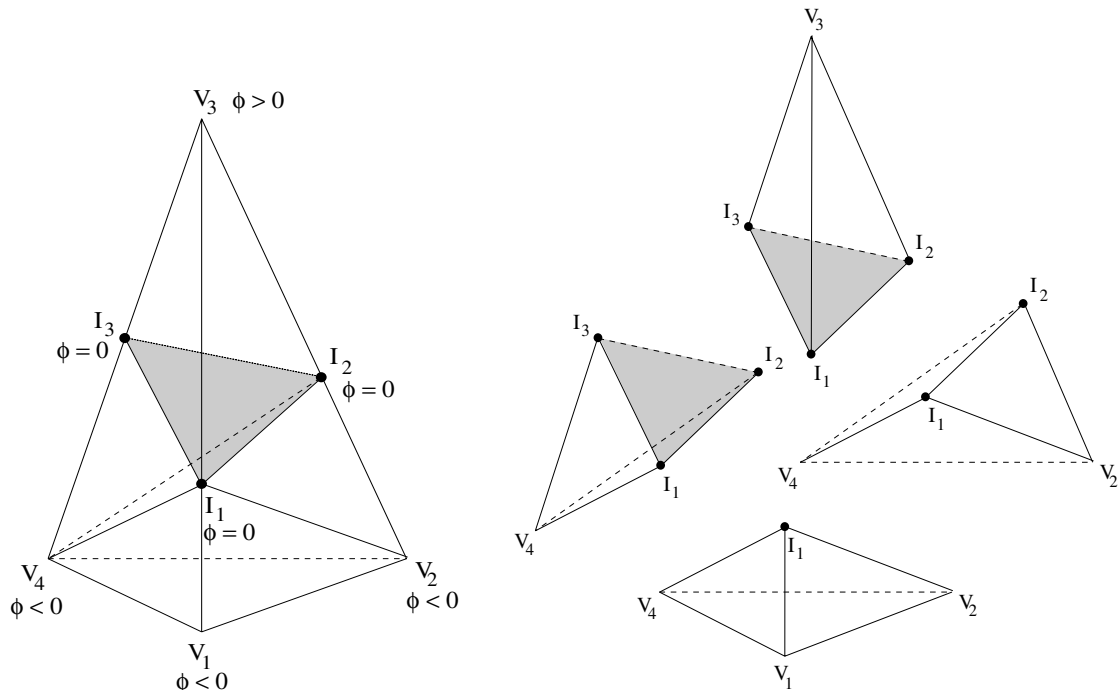


Figure 4.2: Subdivision of a two-dimensional element by the cut-cell method.



(a) Tetrahedron with three intersection points. Subdivision of the faces.

(b) Tetrahedron with three intersection points. Subdivision of element.

Figure 4.3: Subdivision of a three-dimensional element with three intersection points 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, the intersection point is moved to that vertex. For instance, if the intersection point  $I_1$  is excessively close to  $V_3$  in Figure 4.2, then it is

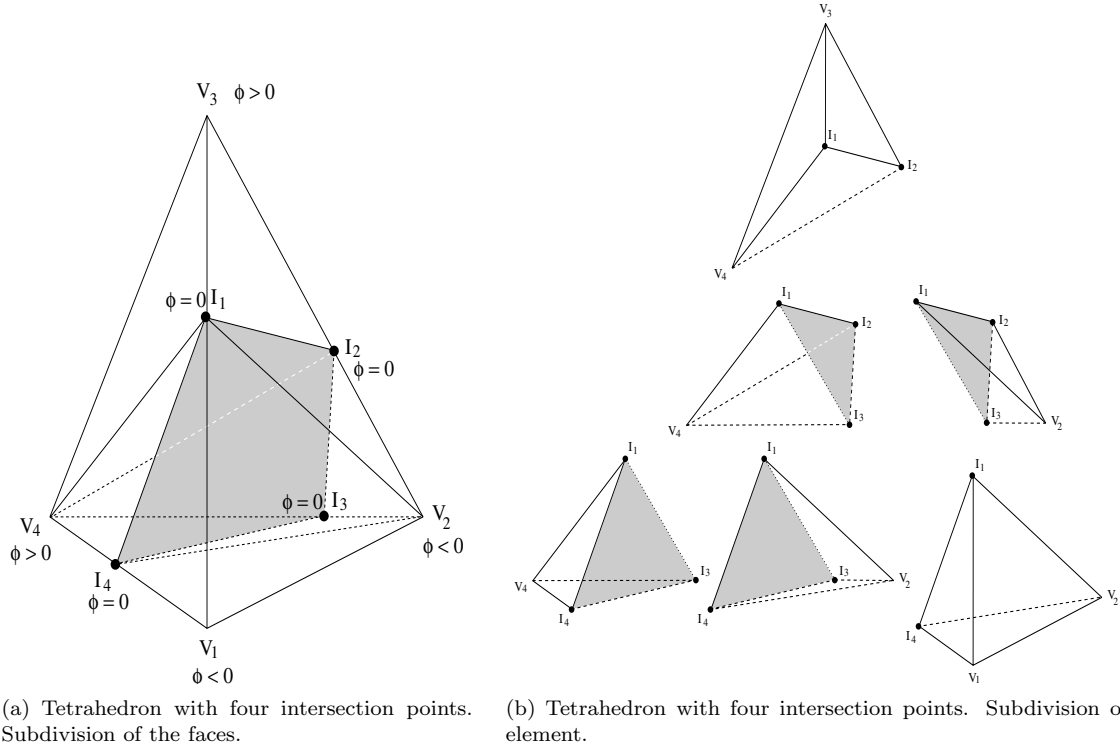


Figure 4.4: Subdivision of a three-dimensional element with four intersection points by the cut-cell method.

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 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 the diffusion problem is solved, the intersection points are removed and the basic mesh is recovered.

## 4.5 Numerical results

The algorithm described above has been implemented in two and three dimensions in the SEPRAN package [66]. First order discretizations are used in the extension of the front velocity and the advection of the level set. The space discretization of the diffusion equation is second order accurate. Furthermore, the time step is determined by  $\Delta t = \min(\Delta t_{CFL}, \Delta t_{min})$ , where  $\Delta t_{CFL}$  is given given by the CFL condition as in Eq. (4.2) and  $\Delta t_{min}$  is used to prevent excessively large time steps. Here  $\Delta t_{min} \propto \Delta x$ , therefore the complete algorithm is first order accurate in time and space. One might also choose  $\Delta t_{min} \propto \frac{\Delta x^2}{D}$  to obtain second order convergence in the diffusion part, but this would not improve the accuracy in the advection of the interface.

The governing equations are such that the mass of the system

$$m(t) = \int_{\Omega} c(\mathbf{x}, t) d\mathbf{x} = c^{part} \int_{\Omega_{part}(t)} d\mathbf{x} + \int_{\Omega_{dp}(t)} c(\mathbf{x}, t) d\mathbf{x}. \quad (4.15)$$

should be conserved during the complete dissolution process. The numerical computation of Eq. (4.15) demands the computation of the percentage of a computational cell occupied by  $\phi > 0$  (resp.  $\phi < 0$ ), which is done as in [67] 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  $m(t)$ .

A number of numerical tests are carried out in order to check the performance of the numerical algorithm presented in this chapter. The computational domain  $\Omega$  will be either the square  $[0, 1]^2$  or the cube  $[0, 1]^3$ . A uniform finite difference mesh, with mesh spacing  $\Delta x = \Delta y = \Delta z = \frac{1}{N}$ , will be used. The geometry of the interface and physical parameters of the test problems are described below.

- Planar interface:  $\Omega_{part}(0) = \{ \mathbf{x} \in \Omega \mid x_1 < s_0 \}$ , where  $s_0 = 0.615$ . The concentrations are  $c^{part} = 0.45$ ,  $c^{sol} = 0.35$ ,  $c^0 = 0.3$ , and the diffusion coefficient is  $D = 1$ .
- Circular interface:  $\Omega_{part}(0) = \{ \mathbf{x} \in \Omega \mid \sqrt{x_1^2 + x_2^2} < s_0 \}$  where  $s_0 = 0.615$ . The concentrations are  $c^{part} = 0.45$ ,  $c^{sol} = 0.33$  and  $c^0 = 0.3$ . The diffusion coefficient is  $D = 1$ .
- Square interface or disk-like particle:  $\Omega_{part} = \{ \mathbf{x} \in \Omega \mid |x_1| < b \text{ and } |x_2| < b \}$  where  $b = 0.55$ . The concentrations are  $c^{part} = 0.45$ ,  $c^{sol} = 0.33$  and  $c^0 = 0.3$ , and the diffusion coefficient is  $D = 1$ .
- Initially perturbed cylindrical particle:  $\Omega_{part}(0) = \{ \mathbf{x} \in \Omega \mid (x_2 - 0.5)^2 + (x_3 - 0.5)^2 \leq r^2 \wedge x_l \leq x_1 \leq x_r \}$ , where  $x_l = 0.15$  and  $x_r = 0.85$ . The perturbation on the surface of the particle is established by

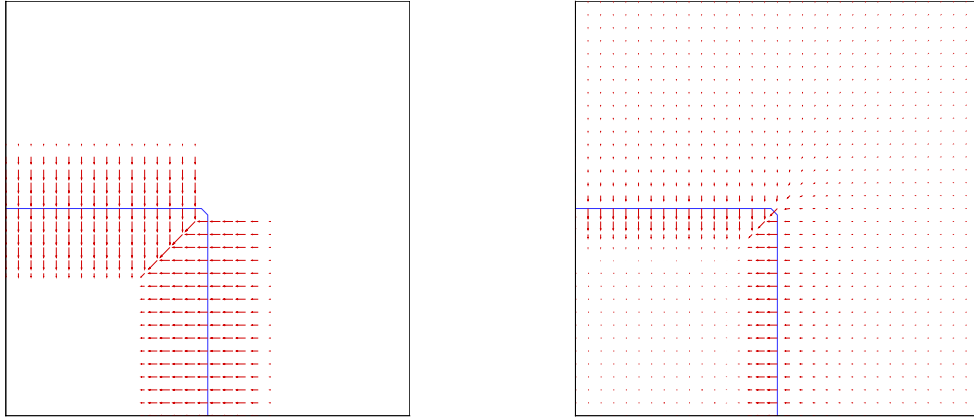
$$r = r(x_1) = \begin{cases} 0.1 - A \sin \left( P(x_1 - (l_i - \frac{\pi}{2P})) \right), & \text{if } |x_1 - l_i| \leq \frac{\pi}{2P}, \text{ for } i = 1, 2, 3, \\ 0.1, & \text{otherwise,} \end{cases}$$

where  $P = \frac{8\pi}{x_r - x_l}$ ,  $A = \frac{\pi}{2P}$  and  $l_i = x_l + i \frac{x_r - x_l}{4}$  for  $i = 1, 2, 3$ . The concentrations are  $c^{part} = 0.45$ ,  $c^{sol} = 0.33$  and  $c^0 = 0.3$ , and the diffusion coefficient is  $D = 1$ .

### 4.5.1 Extension of the front velocity

The Cartesian and normal extensions of the front velocity are plotted in Figure 4.5 together with the interface position at the initial time for the square interface problem. Note that in the Cartesian extension, Figure 4.5(a), the velocity is not extended outside the corner. In this mesh, both the west and south neighbours of the mesh point adjacent to the corner inside the diffusive phase lie in the diffusive phase as well. Hence, in absence of an interface, no velocity is computed. The normal velocity  $v_n$  is computed according to Eq. (4.6) inside the diffusive phase. This provides a continuous velocity field that is extended only in a narrow band inside the particle with Eq. (4.7). Hence, the velocity field obtained in this way, as in Figure 4.5(b), is uniform in the normal direction inside the particle only, and is decreasing in magnitude as we depart from the interface in the normal direction within the diffusive phase due to the concentration distribution.

The average CPU-time per time step used in the extension of the front velocity is shown in Table 4.1. The extension in the normal direction is computed slightly faster, since only one convection equation needs to be solved. The speed up is about 0.8 with respect to the Cartesian extension. The extension in the Cartesian directions does not require the double of CPU-time because the solution of Eqs. (4.5) is cheaper per pseudo-time step.



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

Figure 4.5: The initial position of the interface with the Cartesian extension (left) and the normal extension (right) of the front velocity for the square interface problem.

N	Cartesian Extension	Normal Extension
16	$3.77 \times 10^{-4}$	$3.06 \times 10^{-4}$
32	$1.26 \times 10^{-3}$	$9.15 \times 10^{-4}$
64	$4.90 \times 10^{-3}$	$3.91 \times 10^{-3}$
128	$2.15 \times 10^{-2}$	$1.81 \times 10^{-2}$

Table 4.1: Average CPU-time (seconds) in the extension of the front velocity for the squared interface problem. Processor Intel<sup>®</sup> Pentium<sup>®</sup> 4 (2.80GHz).

## 4.5.2 Convergence to the steady-state solutions and mass conservation

Steady-state solutions are easily determined with a mass balance argument from the initial concentration distribution and interface position. Hence, the final position  $s_\infty$  of the interface in the planar interface test problem is given by

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

Table 4.2 shows the relative errors in mass (given by  $|m(t) - m(0)|/m(0)$ ) and interface position (given by  $|s(t) - s_\infty|/|s_\infty|$ ) at time  $t_{end} = 0.75$ , when the level set method is used. In both cases linear convergence is observed.

In the circular interface test problem, the radius of the particle in the steady state solution  $r_\infty$  is given by

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

under the assumption that the particle remains circular during the dissolution process. However, since the computational domain  $\Omega$  does not have the circular symmetry of the particle, the interface does not remain circular during dissolution. This can be observed in Figure 4.6, where the interface

N	Interface	Mass
16	$6.483 \times 10^{-2}$	$7.248 \times 10^{-3}$
32	$3.318 \times 10^{-2}$	$3.749 \times 10^{-3}$
64	$1.632 \times 10^{-2}$	$1.895 \times 10^{-3}$

Table 4.2: Relative errors in the interface position and mass at time  $t_{end} = 0.75$  with the level set method for the planar interface problem. Initial mass of the system  $m(0) = 0.39225$  and final position of the interface  $s_\infty = 0.4225$ .

position is presented in intervals of 0.1 seconds until the final time  $t_{end} = 1.25$ . Notice that a larger displacement of the interface is observed in the diagonal direction. This is observed more clearly in Table 4.3, 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  $r_\infty$  are presented. The values  $r^{(y=0)}(t)$  and  $r^{(y=x)}(t)$  are defined as follows:

$$r^{(y=0)}(t) = \Gamma(t) \cap \{ (x, y) \in \Omega \mid y = 0 \}, \quad (4.16)$$

$$r^{(y=x)}(t) = \Gamma(t) \cap \{ (x, y) \in \Omega \mid y = x \}. \quad (4.17)$$

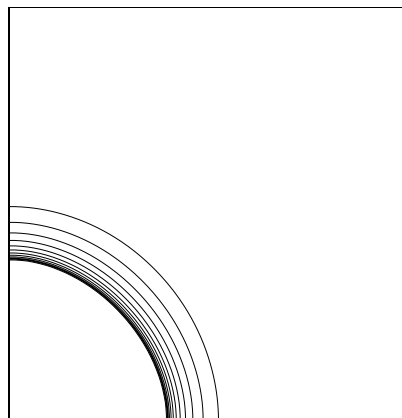


Figure 4.6: Circular interface: snapshots at intervals of 0.1s, using the Cartesian extension of the front velocity.

The error in the diagonal  $r^{(y=x)}$  is always larger than the error in the edge  $r^{(y=0)}$ , independent 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 4.3, 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.

Finally, the relative error in mass at time  $t_{end} = 2.5$  for the square interface test problem is presented in Table 4.4. The normal extension of the front velocity produces slightly better results. Furthermore, it has been observed that the number of iterations required to reinitialize the level set function is slightly reduced when the normal extension of the front velocity is used.

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

Table 4.3: 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  $m(0) = 0.3446$ .

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

Table 4.4: Relative errors in the mass for the squared interface problem. Initial mass:  $m(0) = 0.3458$ .

### 4.5.3 Comparison with a Moving Grid Method

The performance of the level set method is compared in the 2D test problems with the moving grid method [19], in which the ALE convection-diffusion equation

$$\frac{Dc}{Dt} - D\Delta c - \mathbf{u}_{mesh} \cdot \nabla c = 0, \quad (4.18)$$

is solved each time-step in the diffusive phase, where  $\frac{Dc}{Dt}$  denotes the so-called material derivative and  $\mathbf{u}_{mesh}$  stands for the mesh velocity, which is related to the interface velocity. Note that this is an adaptive mesh procedure, in which the computational mesh, only defined inside the diffusive phase and connected to the interface, is updated at each time step. An important feature of the moving grid is that the same nodal points stay on the interface. Due to the movement of the interface, the elements, especially near the interface, may become ill shaped. This is essentially different to what Tryggvason and co-workers [20, 28] present. They combine a stationary mesh technique with a nonstationary curve or surface to represent the interface in two and three spatial dimensions.

Table 4.5 shows the relative errors in mass and interface position for the planar interface test problem, when the moving grid method is used. Similarly to the results for the level set method, Table 4.2, linear convergence is observed. The meshes for both methods have the same number of mesh points. The mesh in the moving grid method, however, is only defined inside 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 Tables 4.2 and 4.5.

Figure 4.7 shows the evolution of the circular interface with the level set (LSM) and the moving grid (MGM) methods. The figure has been enlarged in order to see the differences between both solutions. 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 a smaller mesh width.

Figure 4.8 shows the evolution of the square interface with the level set (Figure 4.8(a)) and

N	Interface	Mass
16	$6.376 \times 10^{-3}$	$8.239 \times 10^{-4}$
32	$3.278 \times 10^{-3}$	$4.867 \times 10^{-4}$
64	$1.867 \times 10^{-3}$	$3.226 \times 10^{-4}$

Table 4.5: Relative errors in the interface position and mass at time  $t_{end} = 0.75$  when the moving grid method is used for the planar interface test problem.

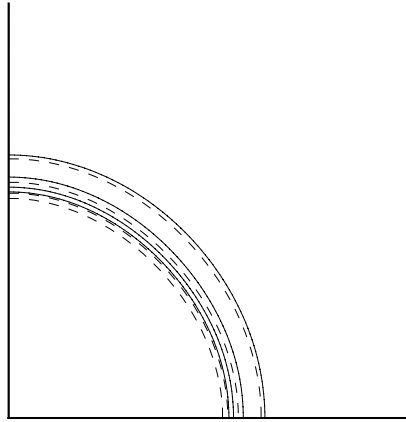
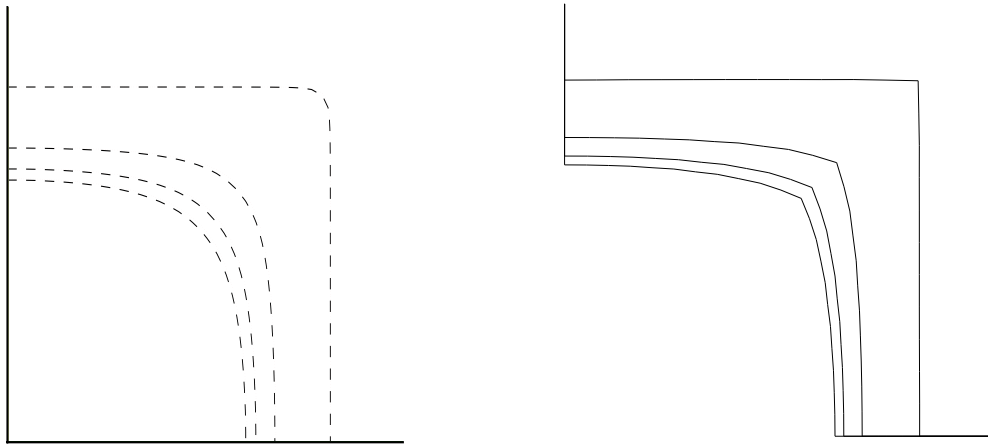


Figure 4.7: Circular interface. Interface position at 0.25, 0.5, 0.75 and 1 (from right to left) for the MGM (solid lines) and the LSM (dashed lines).

the moving grid (Figure 4.8(b)) methods. Furthermore, Figure 4.8(a) shows the evolution of the interface when the Cartesian or the normal extensions of the front velocity are used. Minor differences are observed, although the normal velocity produces a slightly slower displacement of the interface around the corner of the initial interface. The distance of the mesh nodes next to the corner in the normal direction is larger than or equal to the distance in the Cartesian directions, leading to a smaller estimate of the front velocity. The interface evolution with the moving grid method is presented in Figure 4.8(b). In order to represent the initial interface accurately, a mesh point should be placed at the corner. The time evolution of this point is visible in Figure 4.8(b). This comparison shows that, contrary to the level set method, the moving grid method faces difficulties when dealing with singularities on the interface. The mesh in the moving grid method is presented in Figure 4.9 at three different times. This figure shows that remeshing has been necessary at some time between 0.205 and 0.405 to avoid ill shaped elements.

#### 4.5.4 A 3D test problem involving topological changes

In this section we demonstrate the versatility of the method just described by a three-dimensional application. An example is chosen that involves a change in topology. The Cartesian extension of the front velocity is used. The dissolution of an initially perturbed cylindrical particle is presented in Figure 4.10. The computation was carried out with about  $384 \cdot 10^3$  finite elements. Due to the introduced perturbation, the particle breaks into four almost spherical particles, of which the outer particles dissolve faster. The central particles suffer of stronger soft-impingement effects (*i.e.* their diffusion fields overlap making more difficult the flow of atoms from these particles) which delays their dissolution. The change of topology is handled in a natural manner with the level set



(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 4.8: 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) for the square interface problem.

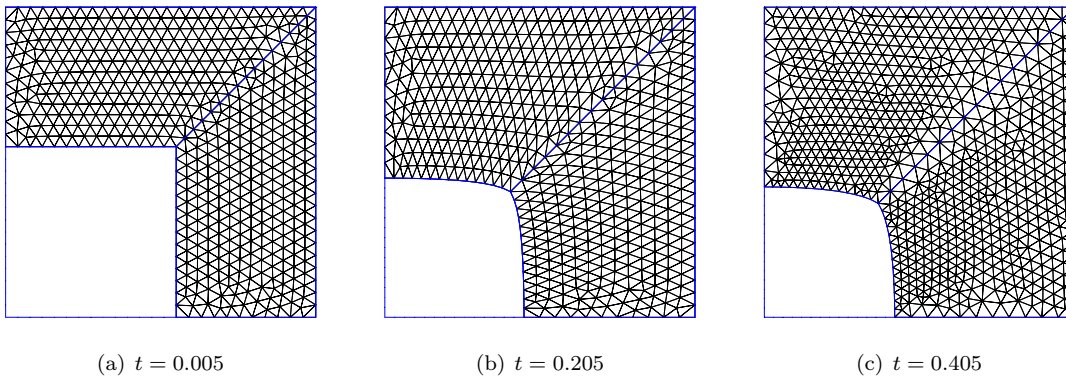


Figure 4.9: Interface position and mesh with the moving grid method at times  $t = 0.005$ ,  $t = 0.205$  and  $t = 0.405$  for the square interface problem.



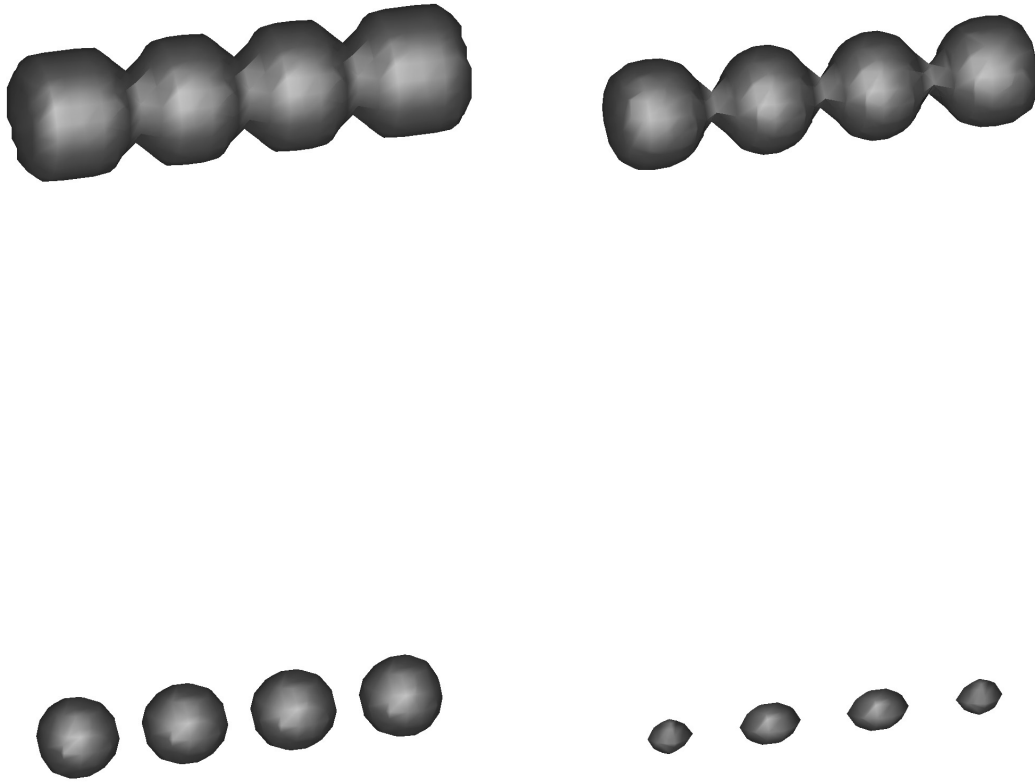


Figure 4.10: Dissolution of an initially perturbed cylindrical particle. Time evolution is from left to right and from top to bottom.

method, due to its implicit representation of the interface. It is our experience with the moving grid method that merging/breaking of interfaces requires an arduous implementation task already in two dimensions, although significant improvements have been obtained in this respect [28].

## 4.6 Conclusions

A numerical technique to simulate dissolution of precipitates in solid state phase transformations has been presented in this chapter. The level set method is used to solve the resulting Stefan problem. A combination of finite difference and finite element methods has been shown to work. Finite difference schemes are used on a uniform Cartesian mesh to approximate hyperbolic equations generated by the level set formulation of the problem. These hyperbolic equations are the level set update Eq. (4.1), extension of the front velocity Eq. (4.5) or (4.7) and reinitialization Eq. (4.8). Finite elements are used to solve the diffusion problem Eq. (2.2) with boundary conditions Eqs. (2.3) and (2.4). The cut-cell approach has been introduced in order to adapt the basic triangulation to the interface position. Interpolation is in general necessary to transfer information between the Cartesian mesh and the finite element triangulation. However, both meshes are based on the same nodal points in this work, and hence interpolation is not needed. The method has been implemented in two and three dimensions.

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 extension in the normal direction is slightly more beneficial: it provides more accurate results, requires fewer pseudo-time-steps when  $\phi$  is reinitialized and uses less CPU-time in its computation. However, when the interface concentration is not constant (for instance if the Gibbs-Thomson relation (2.12) is used or vector-valued Stefan problems are to be solved), extrapolation of the interface velocity in the normal direction is not as feasible as for the problems studied here. This issue will be discussed in the next chapter.

The numerical scheme has been shown to be first order accurate, since the front velocity and the update of the level set function are carried out with first order accurate schemes. Furthermore, singularities and topological changes are handled in a natural way, in contrast with the moving grid method.

---

---

# CHAPTER 5

---

## Higher-dimensional vector-valued Stefan problems: Particle dissolution in multi-component alloys

### 5.1 Introduction

The present chapter is focused on the numerical simulation of particle dissolution in multicomponent alloys, specially in two and three spatial dimensions. The governing equations (2.6)-(2.11) of this problem, described in detail in Chapter 2, are based on the concept of local equilibrium between the adjacent phases. Furthermore, cross-diffusion effects are not considered in this work. Hence, the classical diffusion equation is recovered for each element in the alloy. The interested reader is referred to [15, 16] for further information on this subject.

The level set method developed in Chapter 4 will be generalized to treat dissolution problems in multi-component alloys. The great step to be made in this transition lies in the determination of the interface concentrations, which have to be found as part of the solution and depend both on time and space. Consequently, the procedures presented in Section 4.2.1 to extend the front velocity into the whole computational domain have to be generalized here. On the other hand, the strong nonlinear coupling on the interface concentrations Eq. (2.8) and concentration fluxes through the interface Eq. (2.11) will be solved with a numerical iteration on the interface concentrations. It will be shown in this chapter that a great advantage can be made of the cut-cell approach on both matters.

For the sake of the presentation, the interface position is assumed to be known throughout this chapter. It can be computed using the principles outlined in Chapter 4. First, the continuous extension of the front velocity will be discussed in Section 5.2. Next, the solution to the coupled diffusion problems will be tackled in Section 5.3. After this, a survey of test problems will be used in Section 5.4 to evaluate the performance of the solution method, and the conclusions will be presented in Section 5.5.

## 5.2 Extension of the front velocity for varying $c_1^{sol}$

Equations (2.10) impose that all chemical elements present in the alloy prescribe the same motion to the interface. Thus,  $v_n$  will be computed from  $c_1$  and  $c_1^{sol}$  without loss of generality. It is clear that the extension procedures presented in Chapter 4 need to be adapted to cover the spatial and temporal dependence of the interface concentrations. It was observed from the comparison carried out in the previous chapter that extrapolating the front velocity in the normal direction provided slightly better results. However, its numerical implementation in the case of varying  $c_1^{sol}$  is found to be more laborious because

- (N1) interpolation in the normal direction is needed to determine interface points  $\mathbf{x}_f$  from mesh points  $\mathbf{x}$ , and
- (N2) interpolation from the neighbouring interface points, on which the interface concentration is known, is needed to define  $c_1^{sol}$  at the interface points  $\mathbf{x}_f$  computed in (N1).

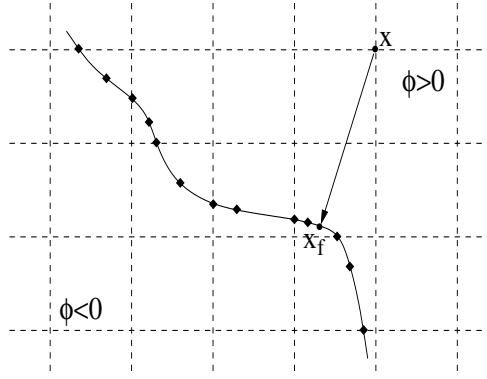


Figure 5.1: The Cartesian mesh with the interface. Interface points denoted by  $\blacklozenge$  are those obtained with the cut-cell method, whereas  $\mathbf{x}_f$  denotes the interface point associated with the mesh point  $\mathbf{x}$ .

This procedure can be visualized as in Figure 5.1. The interface point  $\mathbf{x}_f$  associated with the mesh point  $\mathbf{x}$  at time  $t$  is computed by backwards interpolation of the level set function

$$\mathbf{x}_f(t) := \mathbf{x} - \phi(\mathbf{x}, t)\mathbf{n}(\mathbf{x}, t),$$

for which we assume that the level set function  $\phi$  is a distance function. After solving the diffusion problem, the interfacial concentrations are known at the interface points computed with the cut-cell approach *only*, which are marked with  $\blacklozenge$  in Figure 5.1. Thus, step (N2) is needed to define  $c_1^{sol}$  in  $\mathbf{x}_f$ . The velocity in  $\mathbf{x}$  is consequently defined by

$$v_n(\mathbf{x}, t) = \frac{D_1}{c_1^{part} - c_1^{sol}(\mathbf{x}_f(t), t)} \frac{c_1(\mathbf{x}, t) - c_1^{sol}(\mathbf{x}_f(t), t)}{\phi(\mathbf{x}, t)},$$

and Eq. (4.7) should be applied to advect  $v_n$  inwards the particle. Although (N1) and (N2) involve simple operations, they should be carried out for all the mesh points within the diffusive phase, that for dissolution problems represent an increasing fraction of the number of mesh points in the computational domain.

On the other hand, the Cartesian extension of the front velocity and the reconstruction of the interface performed by the cut-cell method match perfectly. When extending the Cartesian components of the front velocity

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

onto the whole computational domain  $\Omega$ , the following steps are to be taken:

- (C1) Find mesh points adjacent to the interface in each Cartesian direction.
- (C2) Compute the interface points in each direction by linear interpolation of the level set function.
- (C3) Compute the Cartesian components of the front velocity and assign these values to the mesh points adjacent to the interface.
- (C4) Advect the Cartesian components of the front velocity onto  $\Omega$  with Eqs. (4.5).

The interface points computed in (C2) are those provided by the cut-cell method. Therefore, the interface concentration  $c_1^{sol}$  is known at those points. Hence, the computations needed in steps (C3) and (C4) are straightforward tasks similar to the binary case (*i.e.*  $c^{sol}$  constant). Therefore, the extension of the Cartesian components of the interface velocity is preferred in the case of multi-component alloys.

### 5.3 The coupled diffusion problems

At each time step,  $p$  diffusion equations have to be solved simultaneously. A Galerkin Finite Element method is used for that purpose. The underlying triangulation needs to be conformed to the interface location, for which a cut-cell method is implemented (see Chapter 4 for further details). As a result, the background triangulation is enriched with a number of interface points. The hyperbolic relation Eq. (2.8) on the interface concentrations and the Stefan conditions (2.10) on the interface velocity are reformulated as the zero of the function  $\mathbf{f} : \mathbb{R}_+^p \rightarrow \mathbb{R}^p$  given by

$$\begin{cases} f_1(\mathbf{c}^{sol}) = \prod_{i=1}^p (c_i^{sol})^{\tilde{n}_i} - \mathcal{K}(T) \\ f_i(\mathbf{c}^{sol}) = \frac{D_i}{c_i^{part} - c_i^{sol}} \frac{\partial c_i}{\partial \mathbf{n}} - \frac{D_{i-1}}{c_{i-1}^{part} - c_{i-1}^{sol}} \frac{\partial c_{i-1}}{\partial \mathbf{n}}, \quad i = 2, \dots, p \end{cases} \quad (5.1)$$

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

$$\text{(P)} \quad \begin{cases} \frac{\partial c_i}{\partial t}(\mathbf{x}, t) = D_i \Delta c_i(\mathbf{x}, t), & \mathbf{x} \in \Omega_{dp}(t), \quad i = 1, \dots, p, \\ D_i \frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x}, t) = 0, & \mathbf{x} \in \partial \Omega_{dp}(t) \setminus \Gamma(t), \quad i = 1, \dots, p, \\ c_i(\mathbf{x}, t) = c_i^{sol}(\mathbf{x}, t), & \mathbf{x} \in \Gamma(t), \quad i = 1, \dots, p, \\ \mathbf{f}(\mathbf{c}^{sol}(\mathbf{x}, t)) = \mathbf{0}, & \mathbf{x} \in \Gamma(t). \end{cases} \quad (5.2)$$

The strong nonlinear coupling imposed by  $\mathbf{f}$  is solved with an iterative method on the interface concentrations  $c_i^{sol}$ . Each iteration will require the solution of  $p$  diffusion problems with prescribed Dirichlet and Neumann boundary conditions, the computation of the normal fluxes  $\frac{\partial c_i}{\partial \mathbf{n}}$  along the interface and the update of  $\mathbf{f}$ . Therefore, an efficient, robust iterative method is needed of which the extension to higher dimensional problems is affordable.

### 5.3.1 Determination of the interface concentrations

The interface concentrations  $c_i^{sol}$  are found with an iterative method. The performance of Newton, Broyden and Picard methods was investigated in [68] for a one-dimensional problem. The highlights of this comparison are summarized below.

- No analytical expressions for the partial derivatives of  $\mathbf{f}$  are available. Thus, the Jacobian matrix needs to be approximated numerically in the Newton method. Using central differences

$$\frac{\partial f_i}{\partial c_j^{sol}} = \frac{f_i(c_1^{sol}, \dots, c_j^{sol} + \varepsilon, \dots, c_p^{sol}) - f_i(c_1^{sol}, \dots, c_j^{sol} - \varepsilon, \dots, c_p^{sol})}{2\varepsilon}, \quad (5.3)$$

with a sufficiently small increment  $\varepsilon$ , demands  $2p^2$  additional evaluations of  $\mathbf{f}$  per interface point and per iteration. Therefore,  $2p^2$  problems (5.2) with modified Dirichlet boundary conditions are to be solved in order to approximate the Jacobian of  $\mathbf{f}$  in a single interface point and iteration.

- Broyden's method builds a sequential approximation of the Jacobian that avoids the additional evaluations of  $\mathbf{f}$ . However, it still solves a  $p \times p$  system of equations per interface point and iteration.
- The convergence of both Broyden and Newton methods rely on sufficiently accurate initial approximations to the solution. However, those approximations are only possible for idealized geometries. For general problems, only crude approximations can be taken.
- The cost of one Picard iteration is one evaluation of  $\mathbf{f}$  per interface point.

The number of interface points grows linearly in two-dimensional problems and quadratically in three-dimensional problems with the level of refinement of the computational mesh. Therefore, a Picard iteration is preferred to solve two- and three-dimensional problems in spite of its slow convergence rate due to its low computational cost per iteration.

Thus, given  $\mathbf{c}_0^{sol}$ , the zero of  $\mathbf{f}$  is approximated with a fixed-point iteration

$$\mathbf{c}_k^{sol} = \mathbf{g}(\mathbf{c}_{k-1}^{sol}), \quad (5.4)$$

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

$$\begin{cases} g_1(\mathbf{c}^{sol}) = \frac{\mathcal{K}(T)}{(c_1^{sol})^{\tilde{n}_1-1} \prod_{j=2}^p (c_j^{sol})^{\tilde{n}_j}} \\ g_i(\mathbf{c}^{sol}) = c_i^{sol} + \delta_i \left( \frac{D_i}{c_i^{part} - c_i^{sol}} \frac{\partial c_i}{\partial \mathbf{n}} - \frac{D_{i-1}}{c_{i-1}^{part} - c_{i-1}^{sol}} \frac{\partial c_{i-1}}{\partial \mathbf{n}} \right), \quad i = 2, \dots, p, \end{cases} \quad (5.5)$$

where the relaxation parameters  $\delta_i$  are introduced to make  $\mathbf{g}$  a contraction and hence to achieve convergence. It has been seen [68] that  $\delta_i$  depend on the mesh size as well as on the diffusivities. In fact, the choice of  $\mathbf{g}$  should be made according the diffusivities as well. Table 5.1 shows the asymptotic behaviour of the interface concentrations in the similarity solution [14] of a planar interface with  $p = 2$  and  $D_2 = 1$ . In this case, the interface position is given by  $s(t) = s_0 + \alpha\sqrt{t}$ , where  $\alpha$  and the interface concentrations  $c_1^{sol}$  and  $c_2^{sol}$  are the solutions to the system

$$\frac{\alpha}{2} = \frac{c_i^0 - c_i^{sol}}{c_i^{part} - c_i^{sol}} \sqrt{\frac{D_i}{\pi}} \frac{\exp(-\frac{\alpha^2}{4D_i})}{\operatorname{erfc}(\frac{\alpha}{2\sqrt{D_i}})}, \quad i = 1, 2, \quad (5.6)$$

$$(c_1^{sol})^{\tilde{n}_1} (c_2^{sol})^{\tilde{n}_2} = \mathcal{K}(T),$$

	$c_1^0 = c_2^0 = 0$	$0 = c_1^0 \neq c_2^0 > 0$	$0 < c_1^0 \neq c_2^0 > 0$
$D_1 \rightarrow 0$	$c_1^{sol} \rightarrow \bar{n}_1 \sqrt{\frac{\mathcal{K}(T)}{(c_2^{part})^{\bar{n}_2}}}$ $c_2^{sol} \rightarrow c_2^{part}$	$c_1^{sol} \rightarrow \bar{n}_1 \sqrt{\frac{\mathcal{K}(T)}{(c_2^{part})^{\bar{n}_2}}}$ $c_2^{sol} \rightarrow c_2^{part}$	$c_1^{sol} \rightarrow c_1^0$ $c_2^{sol} \rightarrow \bar{n}_2 \sqrt{\frac{\mathcal{K}(T)}{(c_1^0)^{\bar{n}_1}}}$
$D_1 \rightarrow \infty$	$c_1^{sol} \rightarrow c_1^{part}$ $c_2^{sol} \rightarrow \bar{n}_2 \sqrt{\frac{\mathcal{K}(T)}{(c_1^{part})^{\bar{n}_1}}}$	$c_1^{sol} \rightarrow \bar{n}_1 \sqrt{\frac{\mathcal{K}(T)}{(c_2^0)^{\bar{n}_2}}}$ $c_2^{sol} \rightarrow c_2^0$	$c_1^{sol} \rightarrow \bar{n}_1 \sqrt{\frac{\mathcal{K}(T)}{(c_2^0)^{\bar{n}_2}}}$ $c_2^{sol} \rightarrow c_2^0$
$\alpha$	$< 0$	$< 0$	$\leq 0$

Table 5.1: Asymptotic behaviour of the interface concentrations for different values of  $D_1$ .

For instance, in the case of  $c_1^0 = c_2^0 = 0$ , if  $D_1$  is large, then  $c_2^{sol}$  is close to  $c_2^{part}$ , whereas if  $D_1$  is small, then  $c_1^{sol}$  is close to  $c_1^{part}$  in order to achieve the balance of atoms through the interface for both species Eq. (2.11). Similar results were observed numerically in [68] when determining the interface concentrations of a one-dimensional planar interface at time  $\Delta t$ . In cases like described above,  $\mathbf{g}$  should be defined as

$$g_i(\mathbf{c}^{sol}) = c_i^{sol} + \delta_i \left( (c_{i-1}^{part} - c_{i-1}^{sol}) D_i \frac{\partial c_i}{\partial \mathbf{n}} - (c_i^{part} - c_i^{sol}) D_{i-1} \frac{\partial c_{i-1}}{\partial \mathbf{n}} \right), \quad i = 2, \dots, p.$$

The convergence of the fixed-point iteration (5.4) depends moreover on the numerical approximation of the normal fluxes of the concentrations across the interface. For a two-dimensional problem, the following approximation is used:

$$D_i \frac{\partial c_i}{\partial \mathbf{n}}(\mathbf{x}_{f,k}, t) = \frac{1}{2} \left( \frac{1}{\|\mathbf{x}_{f,k} - \mathbf{x}_{f,k-1}\|} \int_{\Gamma_{k-1,k}} D_i \nabla c_i \cdot \mathbf{n} \, d\Gamma + \frac{1}{\|\mathbf{x}_{f,k+1} - \mathbf{x}_{f,k}\|} \int_{\Gamma_{k,k+1}} D_i \nabla c_i \cdot \mathbf{n} \, d\Gamma \right), \quad (5.7)$$

where  $\mathbf{x}_{f,k}$  denotes an interface point,  $\mathbf{x}_{f,k-1}$  and  $\mathbf{x}_{f,k+1}$  its two nearest neighbours and  $\Gamma_{k,k\pm 1}$  the portion of interface between them. The generalization of this approximation to three dimensions is straightforward and hence omitted here. The accuracy of this approximation depends on the shape of the interface elements computed with the cut-cell method, which in turn is controlled by the tolerance  $tol_{cc}$  that determines whether an interface point should be shifted to a mesh point of the background finite element mesh. For binary problems,  $tol_{cc} = 0.01$  is used as a default value. When the background mesh is intersected with the interface, this gives very narrow elements, but this is not a problem because  $D \frac{\partial c}{\partial \mathbf{n}}$  has not to be approximated. However, for multi-component

problems, we need to use  $tol_{cc} = 0.3$  as a default value to avoid these ill-shaped elements. And still, the convergence of (5.4) might be delayed by the local errors in the narrow elements.

The fixed-point iteration  $\mathbf{c}_k^{sol} = \mathbf{g}(\mathbf{c}_{k-1}^{sol})$  is stopped when the  $L_\infty$ -norm of  $\mathbf{f}(c_k^{sol})$  is smaller than a prescribed tolerance  $tol$ .

### Determination of $\mathbf{c}_0^{sol}$

Due to the limited a priori knowledge of the interfacial concentrations  $c_i^{sol}$ , which has been summarized in Table 5.1, providing an initial approximation  $c_{i,0}^{sol}(\mathbf{x}, 0)$  for  $i = 1, \dots, p$  each time step is a difficult task.

The initialization of the interface concentrations (*i.e.*  $t = 0$ ) is somewhat complicated. For idealized geometries, self-similarity solutions [69] might be used to compute the interface concentrations and even the concentrations in the diffusive phase at some initialization time  $t_0$ . For general problems however, these approximations do not capture the spatial variations of  $c_i^{sol}$  along the interface and hence cannot be used. In those cases, the diffusion problem (**P**) is solved with a regularization time  $\Delta t_0 \propto \Delta x^2$  to find suitable concentrations in the diffusive phase and at the interface. In this case, the initial guesses  $c_{i,0}^{sol}(\mathbf{x}, 0)$  for the fixed-point iteration (5.4) are arbitrary constant values that satisfy the hyperbolic relation (2.8). Thus, for instance,  $c_{1,0}^{sol} = c_{2,0}^{sol} = 1$  are used for the test problem studied in Section 5.4.3. In other cases, the values provided by the similarity solutions for planar interfaces can be used to define  $c_{i,0}^{sol}$  for  $i = 1, \dots, p$ .

However, the situation is relatively easier when an approximation to the interface concentrations at time  $t^{n+1}$  has to be given. In this case, one can try to exploit the information from the interface concentrations at the previous time step. One possibility is to interpolate the solution from the previous time step. The interface points at time  $t^n$  could be extrapolated in the normal direction for a distance  $v_n \Delta t^n$ , as sketched in Figure 5.2, to define a curve  $\Gamma'(t^{n+1})$  which is, in general, a good approximation to the interface at time  $t^{n+1}$ . The interface concentrations at  $\Gamma'(t^{n+1})$  could be defined likewise. Subsequently, an initial approximation of the interface concentrations on  $\Gamma(t^{n+1})$  could be obtained by interpolation of the extrapolated interfacial concentrations on  $\Gamma'(t^{n+1})$  to the interface points in  $\Gamma(t^{n+1})$ . Recall that the cut-cell method shifts interface points to background mesh points when their distances are smaller than  $tol_{cc}$ . These shifts spoil the local accuracy of  $\Gamma'(t^{n+1})$ . Moreover, the application of this technique becomes more difficult when the interface changes its topology. In those cases, the approximated curve  $\Gamma'(t^{n+1})$  is not only inaccurate but also singular (*i.e.* with cusps), see Figure 5.2. Thus, in order to achieve an accurate approximation of  $\Gamma(t^{n+1})$ , the interface points that have disappeared since the previous time step should not be taken into account. However, deciding whether a interface point disappears or not is a toilsome task, and therefore this technique has been disregarded.

In this study we use a simpler approach. The average value of the interface concentrations

$$\bar{c}_i^{sol}(t^n) = \frac{1}{n_f} \sum_{k=1}^{n_f} c_i^{sol}(\mathbf{x}_{f,k}, t^n), \quad i = 1, \dots, p, \quad (5.8)$$

where  $n_f$  denotes the number of interface points  $\mathbf{x}_f$  at time  $t^n$ , is used as initial approximation of the interface concentrations at time  $t^{n+1}$ . Note that with this approximation, the spatial variations of  $c_i^{sol}(\cdot, t^n)$  are neglected. Another possibility is to define  $c_{i,0}^{sol}(\mathbf{x}, t^{n+1})$  from the concentrations  $c_i$  at the previous time. For instance, one could define

$$c_{i,0}^{sol}(\mathbf{x}_f, t^{n+1}) = c_i(\bar{\mathbf{x}}_f, t^n),$$

where  $\bar{\mathbf{x}}_f$  denotes the closest mesh point inside the diffusive phase at  $t^n$  to the interface point  $\mathbf{x}_f$ . Note that the diffusive phase at the previous time step must be used to avoid using nodes that



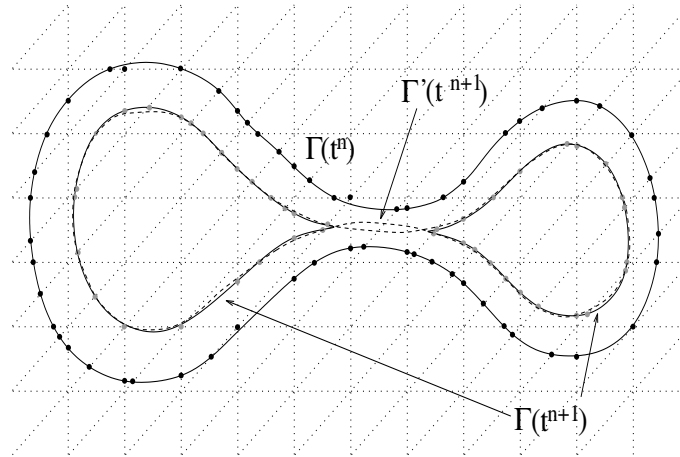


Figure 5.2: The background FE mesh with the interface  $\Gamma$  at two consecutive times, the interface points computed with the cut-cell method and a reconstruction  $\Gamma'(t^{n+1})$  (dashed curve) of  $\Gamma(t^{n+1})$  by advection in the normal direction the points on  $\Gamma(t^n)$ .

have changed phase, *i.e.* that were inside the particle at  $t^n$  and are inside the diffusive phase  $t^{n+1}$ , where the concentrations are the particle concentrations. This approximation would preserve some of the geometrical information of the solution at the previous time.

#### Local adaptations when the interface crosses a mesh point

It was discussed for binary problems that the discretization of the diffusion equation should be locally adapted for mesh points that have changed phase within the computational time step, see Section 4.3. For those mesh points  $\mathbf{x}$ , the time discretization was replaced with

$$\frac{c(\mathbf{x}, t^{n+1}) - c^{sol}}{t^{n+1} - t^*(\mathbf{x})},$$

where  $t^*(\mathbf{x})$  denotes the time (between  $t^n$  and  $t^{n+1}$ ) at which the interface is exactly on  $\mathbf{x}$ . In that case, the adaptation was simple because the interface concentration  $c^{sol}$  was known. The same argument applies to multi-component problems. However, a similar adaptation is not possible because the interface concentrations are not known but have to be found as part of the solution. Instead, the CFL condition on the time-stepping Eq. (4.2) is chosen according to the tolerance  $tol_{cc}$ . Normally,  $tol_{cc} = 0.3$  for vector Stefan problems. Then CFL is chosen smaller than  $tol_{cc}$ , namely CFL=0.25. This choice ensures that before crossing a mesh point, the interface will *lie* unavoidably on this mesh point and hence a continuous concentration profile will exist.

## 5.4 Numerical results

A number of numerical tests are carried out in order to evaluate the performance of the numerical method presented in this chapter. The effects of adding new elements to the alloy are investigated in Section 5.4.1. The dissolution of a two-dimensional dumbbell-shaped particle is used in Section 5.4.2 to analyze the values of the interface concentrations and their time evolution. The dissolution of a three-dimensional dumbbell-shaped particle is presented in Section 5.4.3. Finally, the computational cost of the numerical method for a three-dimensional test problem is discussed in Section 5.4.4.

### 5.4.1 Comparison with the similarity and steady-state solutions

The influence of the numerical method on accuracy and dissolution kinetics of secondary chemical elements is investigated by means of an academic test problem. Consider the dissolution of a planar interface in an alloy with one, two, three and four diffusive chemical species respectively. Note that the scalar Stefan problem studied in Chapter 4 is recovered in the case of only one diffusive species present in the alloy. The computational domain is  $\Omega = [0, 1]^2$ , whereas the interface is initially located at  $\Gamma(0) = \{ \mathbf{x} \in \Omega \mid x_1 = s_0 \}$  with  $s_0 = 0.615$ . Furthermore, the particle is located at the left side of the interface and the diffusive phase at its right. The parameters of the problems, listed in Table 5.2, have been chosen in such a way that the position of the interface  $s_\infty$  at the steady-state solution is the same for all cases under study. For general geometries, the equilibrium interface concentrations  $c_{i,\infty}^{sol}$  and equilibrium particle area (*i.e.* volume)  $V_\infty$  are the solution to system (E) based on a conservation of mass argument and the hyperbolic equation (2.8)

$$\begin{aligned} c_i^{part} V_0 + c_i^0 (V_\Omega - V_0) &= c_i^{part} V_\infty + c_{i,\infty}^{sol} (V_\Omega - V_\infty), \quad i = 1, \dots, p, \\ \prod_{i=1}^p (c_{i,\infty}^{sol})^{\tilde{n}_i} &= \mathcal{K}(T), \end{aligned} \tag{E}$$

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

One species	$c_i^{part} = 5$	$c_i^0 = 5$	$c_i^{sol} = 1$	$D = 1$
Two species	$c_i^{part} = 5$	$c_i^0 = 5$	$\tilde{n}_i = 1$ $\mathcal{K}(T) = 1$	$D_1 = 1, D_2 = 5$
Three species	$c_i^{part} = 5$	$c_i^0 = 5$	$\tilde{n}_i = 1$ $\mathcal{K}(T) = 1$	$D_1 = 1, D_2 = 5$ $D_3 = 0.5$
Four species	$c_i^{part} = 5$	$c_i^0 = 5$	$\tilde{n}_i = 1$ $\mathcal{K}(T) = 1$	$D_1 = 1, D_2 = 5$ $D_3 = 0.5, D_4 = 20$

Table 5.2: Parameters for the test problem of Section 5.4.1.

Figure 5.3 shows the interface position as a function of time for the four cases presented above. The numerical solutions (dashed curves) agree with the similarity solutions (solid curves) at the early stages of the dissolution, and, as time evolves, they converge towards an equilibrium value due to the boundedness of  $\Omega$  and the parameters of the problems. Furthermore, the particle dissolves more rapidly in the ternary alloy than in the binary one because a faster secondary element has been added to the binary alloy, *i.e.*  $D_2 > D$ . The same occurs when quaternary alloy is enriched with the fourth element. On the other hand, when a slow element is added to the ternary alloy, the dissolution process is retarded. Furthermore, it is worth to note that the dissolution process evolves at a similar rate when a binary and a quaternary alloy are considered. This is even more pronounced when the alloy consists of two and four elements. Table 5.3 presents the effective particle concentrations  $c_{eff}^{part}$ , the effective solid solubility  $c_{eff}^{sol}$  and effective diffusion coefficient  $D_{eff}$  as defined in [14] for the dissolution of the particle in a quasi-binary alloy. The coefficients  $\alpha$  and

$$\alpha_{eff} := -2 \frac{c_{eff}^{sol}}{c_{eff}^{part}} \sqrt{\frac{D_{eff}}{\pi}}$$

give a measure of the dissolution rate for the dissolution of the particle in the multi-component alloy and in the quasi-binary alloy respectively. The results show the multi-component problems

with two and four alloying elements can be approximated by almost the same quasi-binary problem.

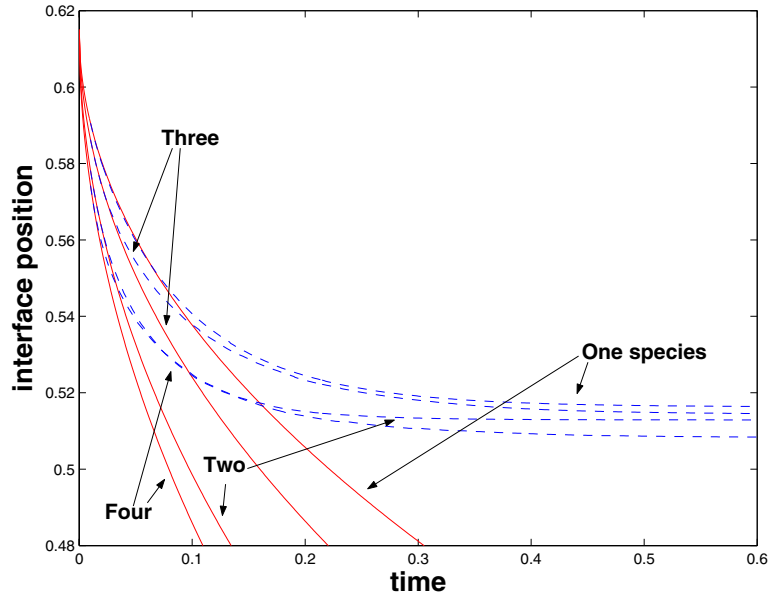


Figure 5.3: Interface position vs time for the dissolution of a planar interface in an alloy with one, two, three and four chemical species respectively. Numerical solutions presented with solid curves and similarity solutions presented with dashed curves.

# species	$c_{eff}^{part}$	$c_{eff}^{sol}$	$D_{eff}$	$\alpha_{eff}$	$\alpha$
One	5	1	1	-0.2444	-0.2444
Three	5	1	0.7937	-0.2011	-0.2876
Two	5	1	2.2361	-0.3375	-0.3682
Four	5	1	2.1147	-0.3282	-0.4081

Table 5.3: Effective particle concentrations, solid solubility, diffusion coefficient and dissolution rate for the quasi-binary alloy, and the dissolution rate coefficient for the multi-component alloy.

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, the impact of the cut-cell method in the accuracy of the results is presented in Table 5.4 for the dissolution in the binary alloy. In this case, it is not necessary to compute  $\frac{\partial c}{\partial \mathbf{n}}$  along the interface, which allows to use a small value for  $tol_{cc}$ . However,  $tol_{cc}$  should be chosen large enough for dissolution in multi-component alloys to prevent ill-shaped elements on the interface which would make the fixed-point iteration given by (5.5) diverge due to poor approximations of the normal fluxes of the concentrations along the interface. Thus,  $tol_{cc}=0.3$  is used as a default tolerance for the dissolution in multi-component alloys. Surprisingly, the largest difference in 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  $\Delta x$ , 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)

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

The relative errors in the interface position at  $t = 0.75$  for the multi-component alloys are presented in Table 5.5. The numerical results show that the accuracy depends on the dissolution rate and does not depend on the number of diffusive elements being present in the alloy. Thus, the numerical errors are smaller for slower dissolution rates, and larger for faster dissolution rates.

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

Table 5.4: Relative error in the interface position at  $t = 0.75$  for the problem with only one diffusive species. Effect of  $tol_{cc}$  on the accuracy:  $tol_{cc}=0.01$  is the default value for dissolution in binary alloys and  $tol_{cc}=0.3$  is the default value for dissolution in multi-component alloys.

Mesh size	Two species	Three species	Four species
$6.25 \cdot 10^{-2}$	$4.41 \cdot 10^{-2}$	$3.79 \cdot 10^{-2}$	$5.15 \cdot 10^{-2}$
$3.13 \cdot 10^{-2}$	$2.84 \cdot 10^{-2}$	$2.05 \cdot 10^{-2}$	$3.62 \cdot 10^{-2}$
$1.56 \cdot 10^{-2}$	$1.71 \cdot 10^{-2}$	$1.36 \cdot 10^{-2}$	$2.57 \cdot 10^{-2}$
$7.81 \cdot 10^{-3}$	$1.13 \cdot 10^{-2}$	$8.39 \cdot 10^{-3}$	$2.02 \cdot 10^{-2}$

Table 5.5: Relative error in the interface position at  $t = 0.75$  for the problems with two, three and four diffusive species.

## 5.4.2 Evolution of the interface concentrations for a two-dimensional test problem

In this section, the time and space variations of the interface concentrations are analyzed in detail for the dissolution of a two-dimensional dumbbell-shaped particle. The computational domain is  $\Omega = [-0.8, 0.8] \times [-0.5, 0.5]$ , and the particle is located at the center. The geometry of the particle is defined as the zero level set of  $\phi^0(\mathbf{x}) = \min(\phi_{circ}(\mathbf{x}), \phi_{bar}(\mathbf{x}))$ , where  $\phi_{circ}(\mathbf{x}) = \min(\sqrt{(x_1 + 0.25)^2 + x_2^2} - 0.15, \sqrt{(x_1 - 0.25)^2 + x_2^2} - 0.2)$  corresponds to the exterior circles and  $\phi_{bar}(\mathbf{x}) = \max(|x_1| - 0.25, |x_2| - 0.035)$  corresponds to the connecting bar. The alloy consists of three diffusive species, *i.e.*  $p = 3$ . The physical parameters of the problems are the following:  $\tilde{n}_i = 1$ ,  $c_i^0 = 0$  for  $i = 1, 2, 3$ ,  $c_1^{part} = 10$ ,  $c_2^{part} = 10$ ,  $c_3^{part} = 5$ ,  $D_1 = 1$ ,  $D_2 = 20$ ,  $D_3 = 80$  and  $\mathcal{K}(T) = 1$ . The diffusivities have been chosen this way to mimic the diffusivities ratios in realistic alloys. Furthermore, the particle concentrations and the particle dimensions have been tuned to obtain a morphological change (*i.e.* break up of the particle) and not complete dissolution, so the steady state solution is reached.

The evolution of the interface and the interface concentrations are plotted in Figures 5.4, 5.5, 5.6 and 5.7. A front view of the interface concentrations is presented to make easier the visualization of the results. The left column of Figures 5.4 and 5.6 presents the approximation of the interface that is made by cut-cell method. The right column of these figures present the interface concentration of the first species. The interface concentrations of the second and third species are

plotted in the left and right columns of 5.5 and 5.7 respectively.

The numerical results show a number of interesting features, some of them are difficult to analyze because of the nonlinear nature of the problem. Below, we try to explain some of these implications:

- In order to satisfy the balance of atoms crossing the interface for each species, the slowest alloying element has the highest interface concentration, the middle alloying element present the middle interface concentration and the fastest alloying element has the smallest interface concentration.
- As time evolves, the interface concentrations for each alloying element tend to reach a flat profile, *i.e.* a constant value along the interface, which finally converges to the equilibrium values, which are the solution to system **(E)**. In this case  $c_{1,\infty}^{sol} = c_{2,\infty}^{sol} = 1.26$  and  $c_{3,\infty}^{sol} = 0.63$ .
- Moreover, the fastest species converge first to this flat profile, the middle species arrives to its flat profile later and the latest to reach its flat profile is the slowest species.
- The interface concentration of the first species (the slowest of the alloy) monotonically decreases in time to the equilibrium  $c_{1,\infty}^{sol}$ , whereas the interface concentration of the third species (the fastest of the alloy) monotonically increases to the  $c_{3,\infty}^{sol}$ . However, none of those possibilities hold for the second chemical species in the alloy. This can be observed following the interface concentration of the points in the connecting bar at the early stages of the dissolution. However, one should keep in mind that the interface concentration of the second species should be such that  $c_1^{sol}$ ,  $c_2^{sol}$  and  $c_3^{sol}$  satisfy the hyperbolic relation (2.8).
- The topology of the interface plays an important role in the determination of the interface concentrations. At the early stages of the dissolution, and for the first alloying element (the slowest), the circular parts of the interface have a higher interface concentration than the planar region at the center. Moreover, the curvature of the circular region on the left is higher, which implies that the interface concentration is higher there. Due to the balance of atoms crossing the interface, the interface concentration of the third alloying element (the fastest) follows the mirrored image of the interface concentration of the first element.
- As time evolves, and the central part becomes thinner, the concentration  $c_1^{sol}$  decrease faster in the circular regions of the interface than in the planar region, and after some time, the concentration in the planar region is higher than in the circular regions. This is due to, in our opinion, the impingement effects between the two circular particles, which influence the diffusion field of the central part. The dissolution in the central part is delayed because of the impingement, whereas in the rest of the domain it proceeds normally. Hence, the time evolution of the interface concentrations on the central part are consequently delayed as well, resulting in the flip-flop behaviour explained before.

### 5.4.3 Dissolution of a three-dimensional dumbbell shaped particle

In this section, the dissolution of a three-dimensional dumbbell-like particle in a ternary alloy (*i.e.*  $p = 2$ ) is simulated. The initial geometry of the dumbbell is given by two spheres, whose centers and radius are  $\mathbf{c}_l = (-0.275, 0, 0)^t$ ,  $\mathbf{c}_r = (0.21, 0, 0)^t$ ,  $r_l = 0.15$  and  $r_r = 0.2$ , connected by a cylindrical bar of radius  $r_{bar} = 0.05$ . The initial level set function for this interface can be constructed as presented in Section 5.4.2 for the two dimensional case. The computational domain is  $\Omega = [-0.5, 0.5]^3$ . The parameters of the problem are:  $\tilde{n}_i = 1$ ,  $c_i^{part} = 5$ ,  $c_i^0 = 0$  for  $i = 1, 2$ ,  $D_1 = 1$ ,  $D_2 = 2$  and  $\mathcal{K}(T) = 1$ . For this set of parameters, the particle will break into two sub-particles which will dissolve individually. The evolution of the interface is presented in Figure 5.8. Figure 5.8(c) shows clearly the cusps in the interface just after the break up of the particle, which are smoothed out rapidly as dissolution evolves. The solution of the nonlinearly coupled problem

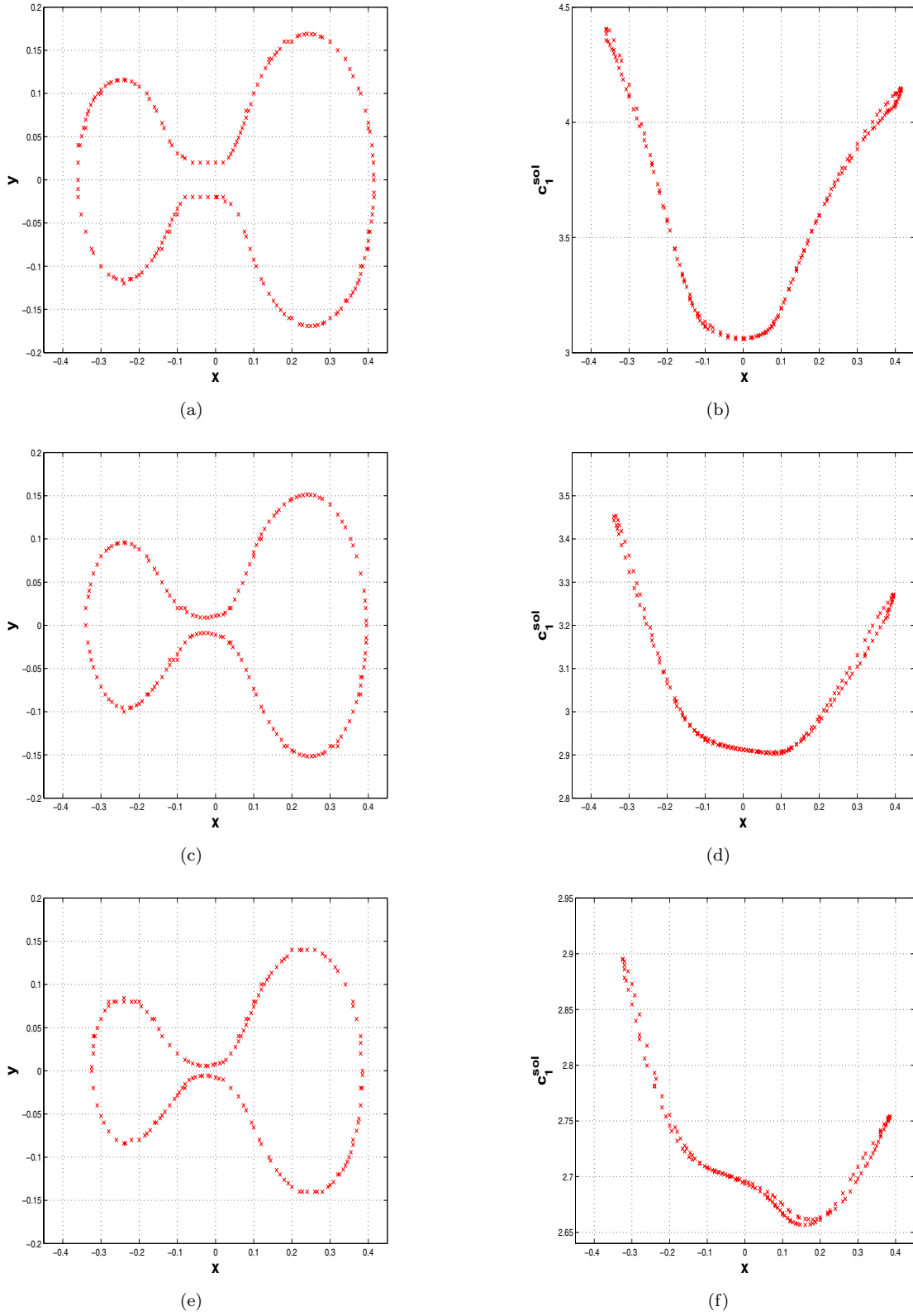


Figure 5.4: Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom):  $t = 3.3 \times 10^{-3}$ ,  $9 \times 10^{-3}$ ,  $1.4 \times 10^{-2}$ . Interface position (left) and  $c_1^{sol}$  (right).

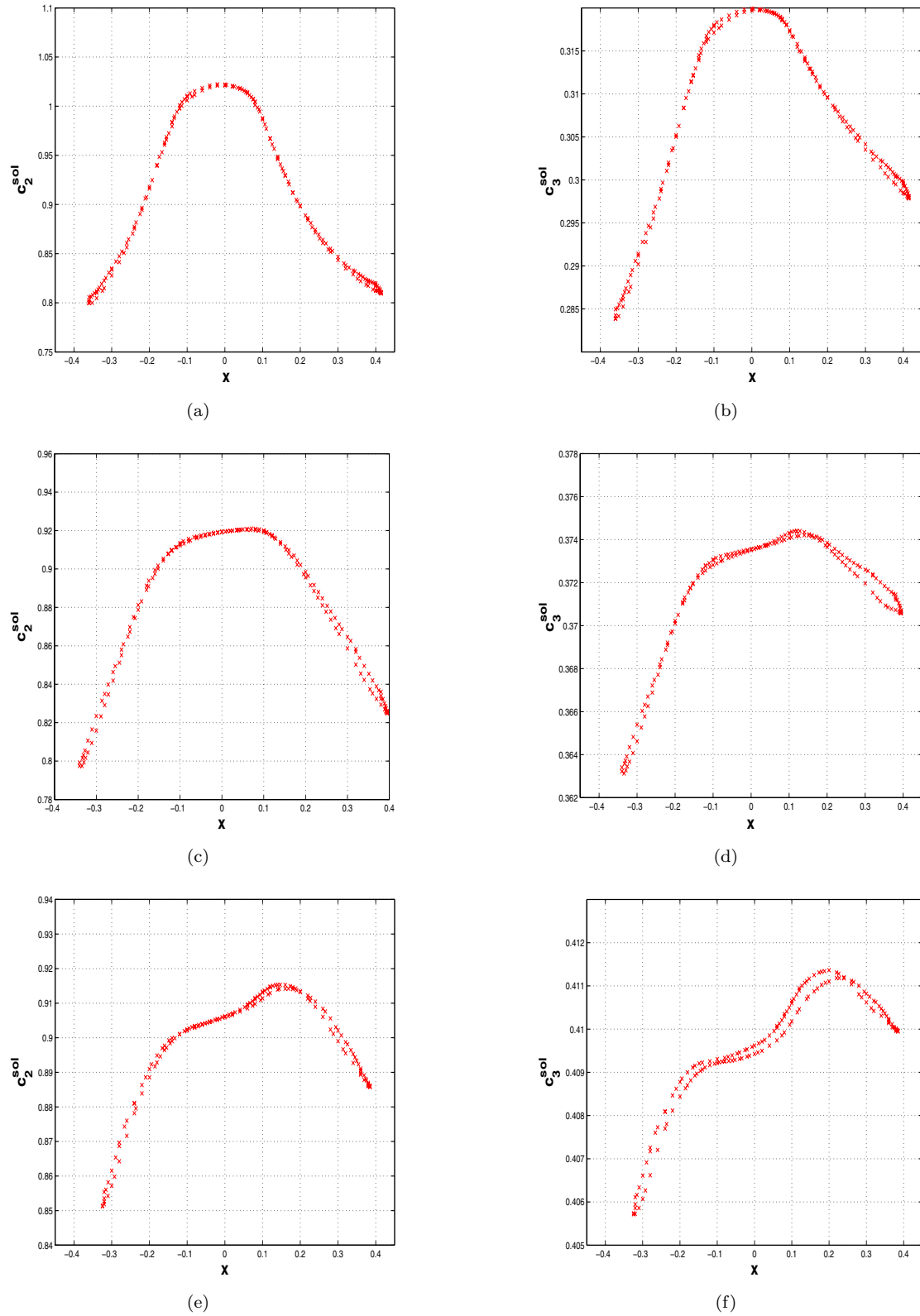


Figure 5.5: Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom):  $t = 3.3 \times 10^{-3}$ ,  $9 \times 10^{-3}$ ,  $1.4 \times 10^{-2}$ .  $c_2^{sol}$  (left) and  $c_3^{sol}$  (right).

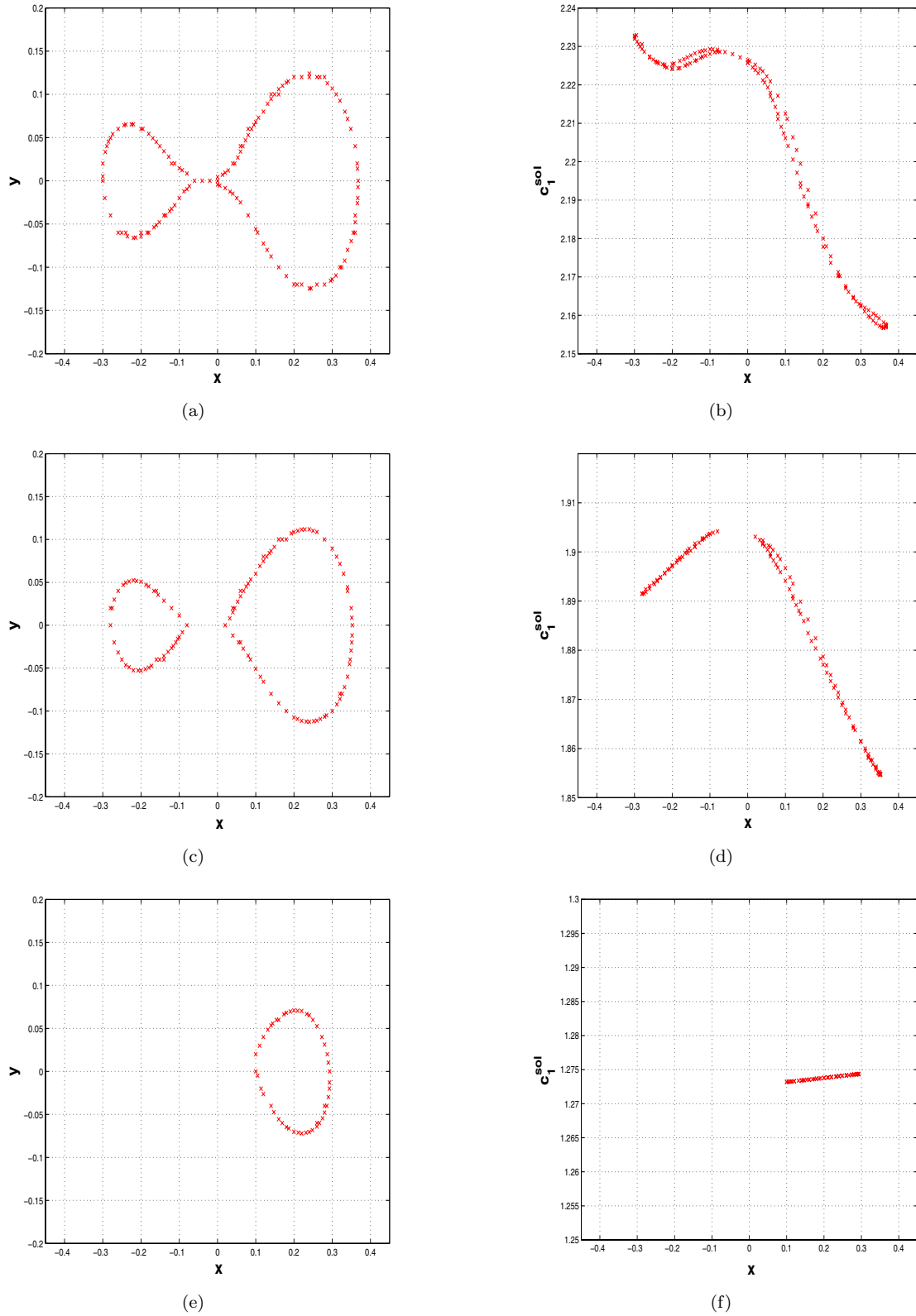


Figure 5.6: Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom):  $t = 3 \times 10^{-2}$ ,  $5 \times 10^{-2}$ ,  $5 \times 10^{-1}$ . Interface position (left) and  $c_1^{sol}$  (right).



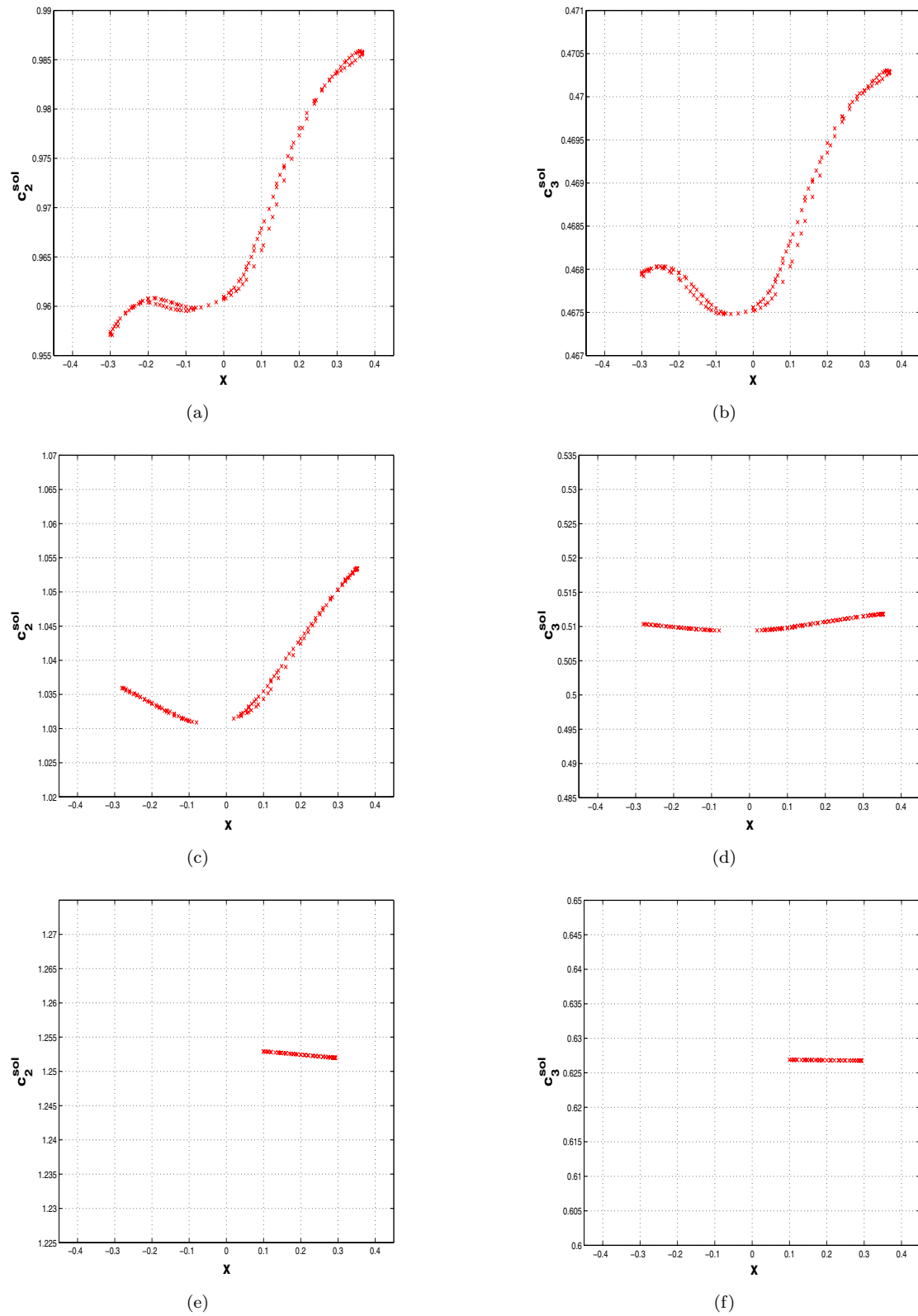


Figure 5.7: Dissolution of a two-dimensional dumbbell-like particle in a quaternary alloy. Times (from top to bottom):  $t = 3 \times 10^{-2}$ ,  $5 \times 10^{-2}$ ,  $5 \times 10^{-1}$ .  $c_2^{sol}$  (left) and  $c_3^{sol}$  (right).

(5.2) is extremely costly for that configuration because the cusps give raise to narrow elements after the background mesh is intersected with the interface with the cut-cell method, which delay the convergence of our fixed point iteration (5.5).

#### 5.4.4 Evaluation of the computational cost of the fixed point iteration

This section is devoted to analyze the computational cost of the numerical method presented in this chapter on a relatively simple three-dimensional test problem. In this case, we consider the dissolution of a cylindrical particle in a ternary alloy (*i.e.*  $p = 2$ ). The computational domain is the unitary cube  $\Omega = [0, 1]^3$ , and the particle is initially given by  $\Omega_{part}(0) = \{ \mathbf{x} \in \Omega \mid x_l < x_l < x_r \wedge \sqrt{(x_2 - c_2)^2 + (x_3 - c_3)^2} - r^2 \}$  with  $x_l = 0.15$ ,  $x_r = 0.85$ ,  $c_2 = 0.5$ ,  $c_3 = 0.5$  and  $r = 0.2$ . The parameters of the problem are:  $\tilde{n}_i = 1$ ,  $c_i^{part} = 5$ ,  $c_i^0 = 0$  for  $i = 1, 2$ ,  $D_1 = 1$ ,  $D_2 = 2$  and  $\mathcal{K}(T) = 1$ .

The computational cost, number of fixed-point iterations and number of interface points as functions of the dissolution time are presented in Figure 5.9. The computational mesh consists of  $41^3 \approx 69 \times 10^3$  nodes and approximately  $384 \times 10^3$  elements. One time step of our method requires the extension of the front velocity, the advection of the interface, the reinitialization of the level set function and the solution of the coupled diffusion problems (5.2). Of those steps, the most expensive both in memory and computational cost is the solution of the coupled diffusion problems. Furthermore, one of the most difficult parts of this problem is the initialization of interface concentrations. In this test problem, similarity solutions cannot be used to initialize the matrix concentrations, and their approximation to the interface concentrations is relatively poor since it does not incorporate the topology of the interface. Furthermore, the initial matrix concentration has a jump discontinuity on the interface, which also complicates the convergence of the fixed-point iteration. Hence, the relaxation parameter  $\delta$  has to be tuned in the initialization step. In this case, we have used  $\delta = 7.5 \times 10^{-3} \Delta x = 1.875 \times 10^{-4}$ , which was sufficient to initialize the interface concentrations. Consequently, the initialization step takes about four hours and 1600 fixed-point iterations. After the interface concentrations have been initialized, there is a short period of time where each time step takes about 110-120 minutes and 600 fixed-point iterations. This is because of the sharp gradients of the concentration, due to the initial discontinuity in the matrix concentrations. After this smoothing time, the cpu-time and number of fixed-point iterations gradually decay because the interface concentrations converge to a flat profile which makes the convergence easier. Hence, the bulk of the computation takes from 10 to 20 minutes and about 70 fixed-point iterations per time step. Finally, a little rise on the cpu-time and number of fixed-points iterations is observed when the particle is next to be completely dissolved.

The convergence rate of the fixed-point iteration depends on many factors, such as the dissolution time, the quality of the mesh after the background mesh is intersected with the interface and the relaxation parameters  $\delta_i$ . In this case, we have used  $\delta_i$  constant through the complete simulation. Generally, the values of  $\delta_i$  which are valid for the initialization step are valid during the whole computer simulation, although this is not always the case. For instance, for the dissolution of the three-dimensional particle presented in Section 5.4.3, the parameter  $\delta$  had to be reduced when the particle breaks up. On the other hand, excessively small values of  $\delta_i$  delay the convergence rate substantially.

## 5.5 Conclusions

In this chapter, the numerical method presented in Chapter 4 for particle dissolution in binary alloys has been extended to multi-component alloys. As far as we know, it is the first time that this problem has been solved in three dimensions. A great effort has been put in the description of the numerical method and the numerical difficulties that arise when the solution of these complex nonlinear problems are to be solved in two and three spatial dimensions. The a priori knowledge of

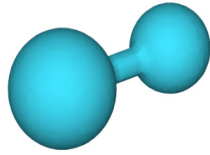
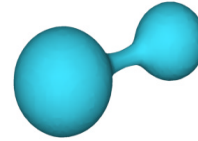
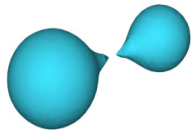
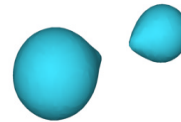
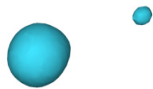
(a)  $t=0$ (b)  $t = 2.9 \times 10^{-3}$ (c)  $t = 7.4 \times 10^{-3}$ (d)  $t = 1.05 \times 10^{-2}$ (e)  $t = 2.85 \times 10^{-2}$ (f)  $t = 4.82 \times 10^{-2}$ 

Figure 5.8: Dissolution of a three-dimensional dumbbell-like particle in a ternary alloy.

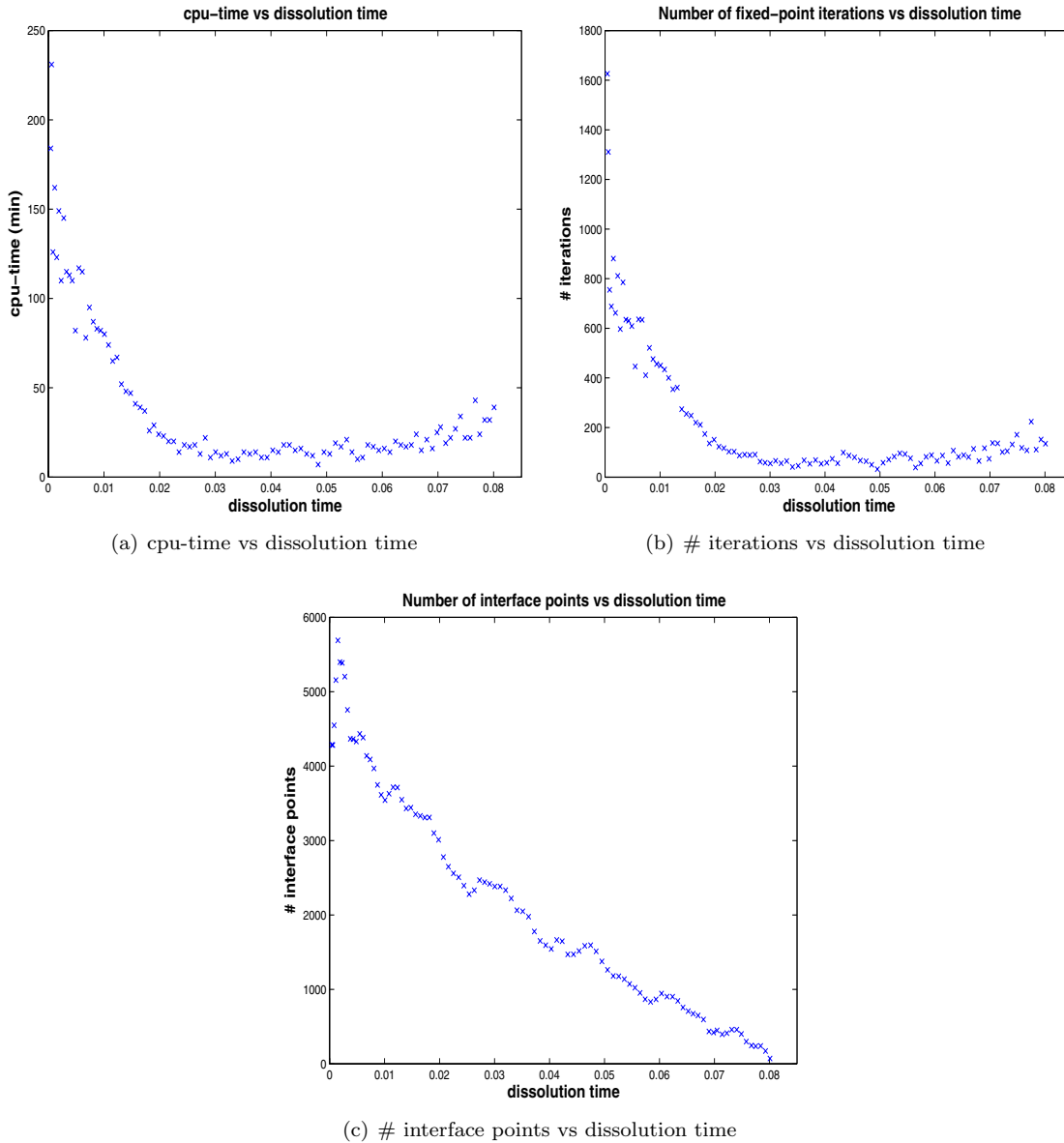


Figure 5.9: The computational cost of the dissolution of a cylindrical particle. Processor Intel<sup>®</sup> Pentium<sup>®</sup> 4 (3.00GHz)

the problem must be exploited to define the most suitable fixed-point iteration. In this work, the relaxation parameters have been tuned for each test problem. In general, it is not difficult to come across values for  $\delta_i$  that ensure convergence of the fixed-point iteration, since  $\delta_i$  sufficiently small would work in any case. However, a too small choice of  $\delta_i$  would yield a very slow convergence rate, and hence of any practical value. Therefore, a method to optimize the choice of  $\delta_i$  is needed to improve the efficiency of this numerical method.

The test problems studied in this chapter have shown that the performance of our method is not affected by the number of chemical species in the alloy. Moreover, convergence to the steady state solutions has been observed and agreement with similarity solution defined in unbounded

domains has been obtained at the early stages of the dissolution. The interface concentrations always follow the same pattern: after a transition time, the interface concentrations reach a constant value from which they converge to the equilibrium values, if they exist. Soft-impingement effects, however, delay the evolution of the interface concentration, which results in a change of the local distribution of the interface concentrations.

The analysis of the computational cost of the method shows that there is room for improvement. In any case, the initialization step is always going to be difficult, and this is the most expensive step. However, in our opinion, a better choice of the initial guess for the fixed-point iteration would help the convergence of our method considerably. Moreover, the convergence rate could be also improved by increasing  $\delta_i$  with time, and by using some post-processing technique to smooth out the errors in  $D_i \frac{\partial c_i}{\partial \mathbf{n}}$  that arise at the narrowest interfacial elements.



---

---

# CHAPTER 6

---

## Metallurgical applications

### 6.1 Introduction

In this chapter the level set method presented in previous chapters is applied to three industrial problems. The coupled dissolution of a spherical cementite particle and growth of an austenite shell in a ferrite matrix is analyzed in Section 6.2.1. Next, the spheroidization process of a cementite plate in a ferrite phase is simulated, and the effect of surface micro-cracks in the dissolution kinetics is investigated in Section 6.2.2. Finally, the effect of the particle morphology on the dissolution kinetics is studied in Section 6.2.3.

### 6.2 Applications

#### 6.2.1 Dissolution of spherical cementite

Consider a spherical cementite particle of stoichiometry  $\text{Fe}_3\text{C}$  that is initially located in a ferrite matrix. At  $t = 0$  the temperature is increased up to  $800^\circ\text{C}$ , hence above the  $A_1$ -temperature. Then, two phenomena take place simultaneously: ferrite transforms into austenite and the cementite particle starts to dissolve. The nucleation of austenite takes place at the interface separating the cementite particle and ferrite phase. This is modelled by the presence of an expanding austenite shell between the dissolving cementite particle and ferrite phase. The diffusion of carbon in austenite and ferrite is assumed to be rate-determining. Further, the presence of other alloying elements is disregarded, so we treat the steel as a binary alloy.

Mathematically speaking the problem is stated as follows:

$$\frac{\partial c}{\partial t} = D_p \frac{1}{r^2} \frac{\partial}{\partial r} \left\{ r^2 \frac{\partial c}{\partial r} \right\}, \text{ for } r \in \Omega_p(t), t > 0, p \in \{\gamma, \alpha\}. \quad (6.1)$$

in spherical coordinates. Here  $\Omega_\gamma$  and  $\Omega_\alpha$  respectively denote the austenite and ferrite phase and are represented by  $\Omega_\gamma(t) := \{r \in \mathbb{R} : S_1(t) < r < S_2(t)\}$  and  $\Omega_\alpha := \{r \in \mathbb{R} : S_2(t) < r < R\}$ . Here  $R$  denotes the radius of the cell, which is set at  $5 \mu\text{m}$ , and  $S_1(t)$  and  $S_2(t)$  respectively denote the positions of the cementite/austenite and the austenite/ferrite interface. The initial thickness of the austenite phase is set at  $0 \mu\text{m}$ . At the interfaces the concentrations are prescribed by the solubilities according to the assumption of local equilibrium. The movement of the interfaces is

given by:

$$\begin{aligned} (c^\theta - c^{\gamma/\theta}) \frac{dS_1}{dt}(t) &= D_\gamma \frac{\partial c}{\partial r}(S_1^+(t), t), \\ (c^{\gamma/\alpha} - c^{\alpha/\gamma}) \frac{dS_2}{dt}(t) &= D_\alpha \frac{\partial c}{\partial r}(S_2^+(t), t) - D_\gamma \frac{\partial c}{\partial r}(S_2^-(t), t), \end{aligned} \quad (6.2)$$

In Figure 6.1 we present some plots at various times of the concentration profile where the three adjacent phases are present. In the calculations we use a diffusivity of carbon in austenite

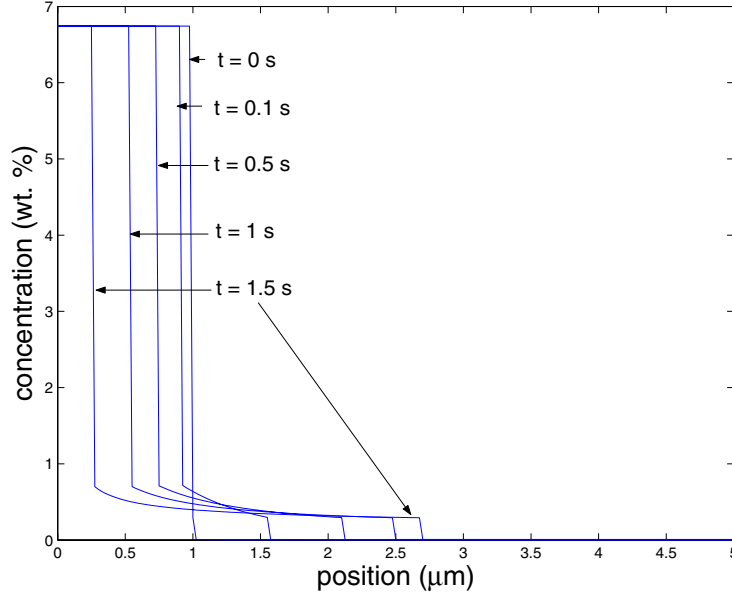


Figure 6.1: Carbon concentration profiles at consecutive times.

$D_\gamma = 2.98 \mu\text{m}^2/\text{s}$  and in ferrite  $D_\alpha = 78.7 \mu\text{m}^2/\text{s}$  [70]. The cementite particle is assumed to be stoichiometric at all stages of the dissolution process. The cementite particle concentration is  $c^\theta = 6.743 \text{ wt.}\%$ . The interface concentrations are  $c^{\gamma/\theta} = 0.71719 \text{ wt.}\%$ ,  $c^{\gamma/\alpha} = 0.29224 \text{ wt.}\%$  and  $c^{\alpha/\gamma} = 0.741 \cdot 10^{-2} \text{ wt.}\%$ . We note that Atkinson *et al.* [71] constructed a self-similar solution for a planar phase. Our computations have been done for the dissolution of a spherical cementite particle. If the initial thickness of the austenite phase is nonzero, we use for the initial concentration within the austenite phase,  $c^0$ , the steady-state solution of the spherical diffusion equation with the above mentioned interface concentration as boundary conditions. Hence,

$$c^0(r) = c^{\gamma/\theta} + \frac{c^{\gamma/\theta} - c^{\gamma/\alpha}}{\left(\frac{1}{S_1(0)} - \frac{1}{S_2(0)}\right)} \left(\frac{1}{r} - \frac{1}{S_1(0)}\right). \quad (6.3)$$

In Figure 6.2 the interface positions are plotted as a function of time. In many simulations the ferrite to austenite transformation is assumed to be much faster than the cementite dissolution. Hence the ferrite phase is often disregarded. Finally, several plots of the cementite-austenite interface as a function of time are presented in Figure 6.3 for several values of the initial position of the austenite-ferrite interface. Figure 6.3 shows the dependence of the cementite particle radius on the initial thickness of the austenite phase, especially in the regime of a small initial thickness of the austenite phase. Hence the assumption that the ferrite phase can be disregarded in the calculations is contradicted for a binary iron-carbon system. Note that the dissolution times are very small because only the diffusion of carbon, which diffuses very fast in iron, is taken into account. For the case that chromium is also present, the low diffusion coefficient of chromium



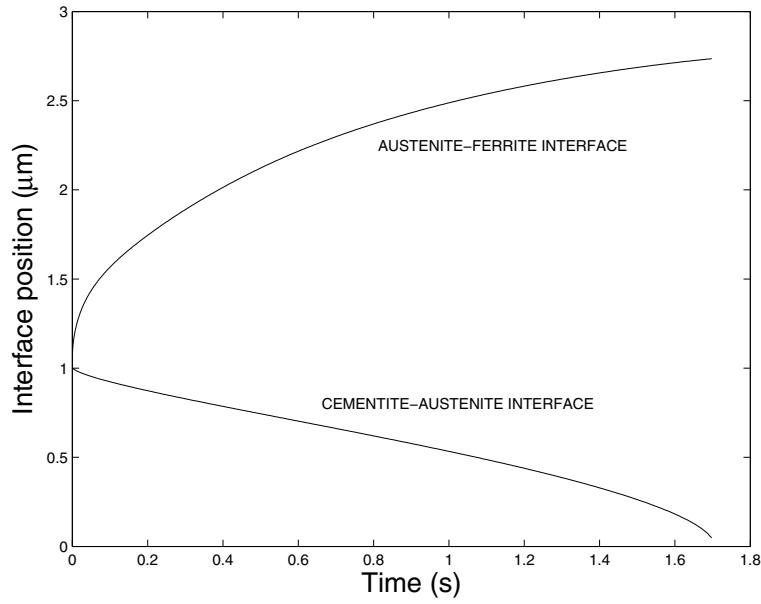


Figure 6.2: The interfacial positions of the cementite-austenite and austenite-ferrite interfaces as a function of time.

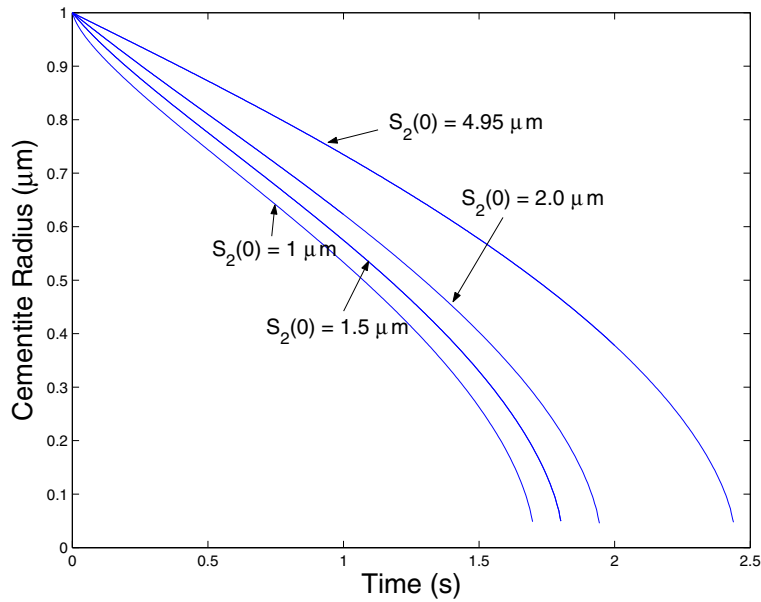


Figure 6.3: The interfacial positions of the cementite-austenite interface as a function of time for several values of the initial position of the austenite-ferrite interface.

will be the rate determining factor. An estimate of the order of magnitude for the dissolution of cementite with carbon can be made using the effective diffusion coefficient as defined in [15], which is approximated by

$$D_{\text{eff}} \approx \sqrt{D_C D_{\text{Cr}}}, \quad (6.4)$$

with  $D_C = 2.98 \mu\text{m}^2/\text{s}$  and  $D_{\text{Cr}} = 6.73 \cdot 10^{-6} \mu\text{m}^2/\text{s}$  as the diffusion coefficients of carbon and

chromium respectively in the austenite phase, we have  $D_{\text{eff}} \approx 4.4 \cdot 10^{-3} \mu\text{m}^2/\text{s}$ . Note that the relation (6.4) only holds for an unbounded domain and hence provides us with a zeroth order approximation. Since the dissolution process is scalable, the time of the dissolution process can be estimated by  $t_{\text{dis}} \approx 1.6 \times \frac{D_{\text{eff}}}{D_C} \approx 1.6 \times 677 = 1117$  seconds. The larger dissolution time (for  $S_2(0) = 4.95 \mu\text{m}$ ) will be around 1693 seconds. Note that in this estimate the effective interface concentration has been set equal to the interface concentration of carbon on the interface between cementite and austenite. The relation for the effective interface concentration that was derived in [15] only holds for the case that the interface concentrations satisfy an idealized solubility product for a perfectly stoichiometric cementite phase. Since cementite is not perfectly stoichiometric, the effective interface concentration cannot be determined using the equation by Vermolen *et al.* [15]. The present work only aims at being qualitative and not quantitative, although after scaling the dissolution times resemble the dissolution times of a cementite particle in an AISI 52100 steel.

Note that for this case with two moving interfaces, two level set functions are used. For each interface there is a level set function, which is put at zero at the corresponding interface and whose sign changes there. The diffusion coefficients that are used in the adjacent phases are obtained by checking the signs of the level set functions to identify the various phases.

## 6.2.2 Spheroidization process in lamellar structures

The metallurgical application studied in this section is inspired by an AISI 52100 steel, which is widely used to make hardened components such as bearings, gears, transmission shifts, etc. due to its excellent properties. The spheroidization process of dissolving lamellar pearlite structures, which is the major part of the soft annealing process in order to make the steel machinable, will be simulated here. AISI 52100 steels contain mainly 1.01 wt.% of carbon and 1.36 wt.% of chromium. However, the diffusion of chromium is neglected here. Thus, the alloy is treated as a binary alloy considering only the diffusion of carbon. This physically justifiable binary assumption is made due to the large computational times involved in the solution of 3D vector Stefan problems which involve diffusivities varying several orders of magnitude (the reader is referred to Chapter 5 for further details on this concern). Therefore, the calculations presented in this section correspond with a hypothetical alloy, and the results serve to qualitatively illustrate the spheroidization process. In a recent work due to Zhao *et al.* [72], the diffusion of carbon and chromium is treated for a dissolving spherical cementite particle. In this article, Zhao and co-workers use a 1D formulation of the problem due to the spherical symmetry, which reduces the complexity of the problem tremendously, but also limits the ability of the model for dealing with more general geometries.

Consider a planar cementite phase surrounded by a planar ferrite matrix. The lamellae structure of pearlite is simplified into one cementite plate for reasons of symmetry. In order to induce the spheroidization process, a perturbation on the particle surface is introduced. Two perturbations for the planar particle are assumed: one is a knucklebone shape [73] and the second is a sinusoidal perturbation. Furthermore, it is assumed that the voids in the particle shape are filled by the ferrite phase entirely. The initial configurations for those perturbations are given in Figure 6.4(a) and Figure 6.5(a) respectively. The temperature has been chosen at  $T = 800^\circ\text{C} > 789^\circ\text{C}$ , which is the finish temperature of ferrite dissolution [72], and the plate dimensions of  $0.1 \times 1 \times 5 \mu\text{m}^3$  have been used. The plate dissolves in a computational cell with dimensions  $0.5 \times 2.4 \times 5 \mu\text{m}^3$ . Hence, the initial volume fraction of the dissolving plate is 8.33%. The concentration of carbon within the cementite particle is given by  $c^\theta = 6.743$  wt.%. Further, the initial carbon composition in the ferrite matrix, which is 'inherited' by the austenite upon its first formation, is set at 0 wt.%. We note that the overall carbon concentration in the alloy is less than 1.0 wt.%, which holds for the AISI 52100 steel. We chose this in order to be able to visualize the complete dissolution of the plates including breaking up into sub-platelets as the dissolution process proceeds. The concentration at the interface between the cementite and ferrite phases is set equal to the value that follows from local equilibrium at  $T = 800^\circ\text{C}$ , given by  $c^{\gamma/\theta} = 0.71719$  wt.%. For the diffusion coefficient the value  $D = 2.98 \mu\text{m}^2/\text{s}$  [70] (page 99) is used, corresponding to the

temperature of 800°C.

A few snapshots of the dissolution process of the cementite plate with a knucklebone shape (*i.e.* linear perturbation) and with a sinusoidal perturbation are given in Figures 6.4 and 6.5 respectively. In both cases, the particle gradually splits up into adjacent sub-particles and, as time proceeds, the sub-particles dissolve entirely. The importance of breaking up of dissolving phases is illustrated in the measurements presented in [74]. Furthermore, although the depth of the crack is the same in both cases, the geometry of the sub-particles and their appearance time are quite different. This is because of the linear crack describing the knucklebone plate produces a sharper and more uniform perturbation, whereas the sinusoidal perturbation results in a smoother surface with a varying crack distribution. Thus, the sub-particles induced by the knucklebone shape become more and more spheroidized as dissolution evolves, whereas those induced by the sinusoidal perturbation present more elongated shapes. The displacement of the interface with respect to its original position varies both locally (for each sub unit in the plate) and macroscopically (edge units behave differently to center units). Furthermore, the initially sharper edges are rounded off during the dissolution process.

Next, the normalized volume ( $V(t)/V_0$ , where  $V_0$  is the initial volume of the unperturbed plate) of the cementite particle as a function of the normalized dissolution time ( $t/t_{diss}$ , where  $t_{diss}$  denotes the time of complete dissolution of the unperturbed plate) for different initial geometries of the perturbed particle is presented in Figure 6.6. The first curve (from the right) corresponds to the dissolution of the unperturbed particle. The second curve is based on the dissolution of a collection of 20 platelets whose width and thickness have been increased to have the same initial volume fraction as the unperturbed plate, *i.e.* 8.33%. The third and fourth curves correspond, respectively, to the dissolution of a thicker plate (*i.e.* its size has been enlarged in the  $z$ -direction) and a wider plate (*i.e.* its size has been enlarged in the  $x$ -direction) with a linear crack. The dimensions of the perturbed particles have been modified in order to have an initial volume fraction of 8.33%, as the unperturbed plate. From the simulations it is clear that the cracks give an acceleration of the dissolution of the cementite plate, which is also commonly known in the metal processing community. For the last case, where the thickness has not been enlarged, the dissolution speed is almost three times as large as the dissolution speed of the unperturbed plate. Note that the thickness of the particles in case 2 is larger than the thickness of the unperturbed plate, which explains why the reduction of the dissolution time is relatively small.

Subsequently, we present the dissolution time of several individual particles of a set of 20 neighbouring particles. The used configuration and numbering of the particles is sketched in Figure 6.7. The dissolution kinetics for an edge particle (I), its first neighbour (II) and one of the central particles (V) are plotted in Figure 6.8. The curves labeled with T1 correspond to a simulation in which the 20 adjacent particles are present, whereas the curves labeled with T2 correspond to a simulation in which particles III, VIII, XIII and XVIII were removed. Since the edge particle is attached to the non-flux boundary and hence atoms can not flow out of the particle in that direction, its dissolution speed is reduced compared to the dissolution speed of the inner particles. Furthermore, in the case where the 20 particles are present (T1), the inner particles dissolve at the same rate. The differences observed in the curves for particles II and V are due to mesh effects and numerical errors rather than different dissolution kinetics. On the other hand, removing particles III, VIII, XIII and XVIII has a strong impact on the dissolution kinetics on the remaining particles. Thus, the dissolution time of particle II is reduced in approximately a 50%, whereas the dissolution time of particles I and V is reduced in approximately a 30%. Cross-sectional profiles parallel to the  $xy$ -plane at consecutive times have been plotted in Figures 6.9 and 6.10 for T1 and T2 respectively. In the case where the 20 platelets are present, the strong soft-impingement effects between neighbouring particles causes that the dissolution follows towards the center. However, when particles III, VIII, XIII and XVIII are removed, the particles next to them dissolve also in the  $y$  direction, *i.e.* their atoms can scape in the  $y$  direction as well, accelerating the dissolution process.

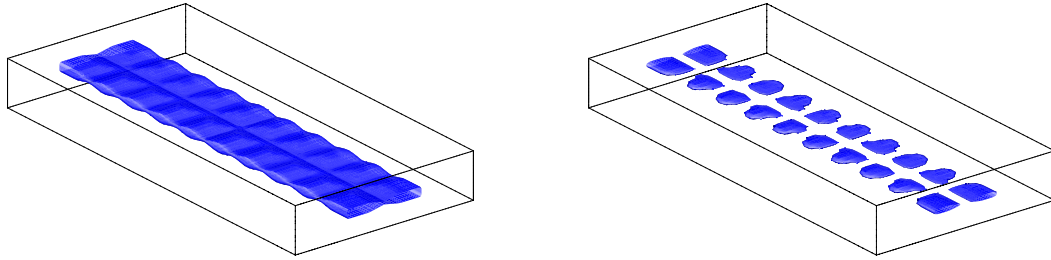
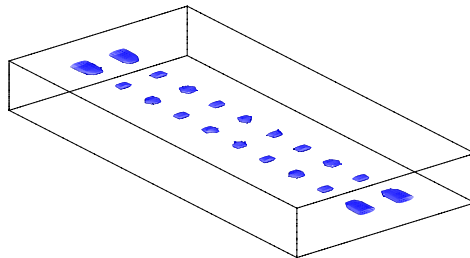
(a)  $t = 0$ .(b)  $t = 8.87 \times 10^{-3}$ .(c)  $t = 1.64 \times 10^{-2}$ .

Figure 6.4: Dissolution of a cementite plate with a knucklebone shape.

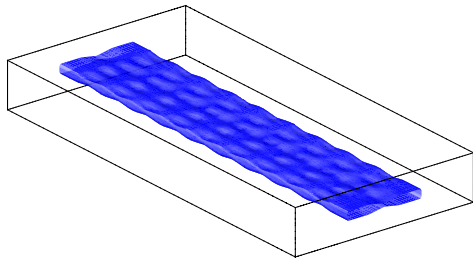
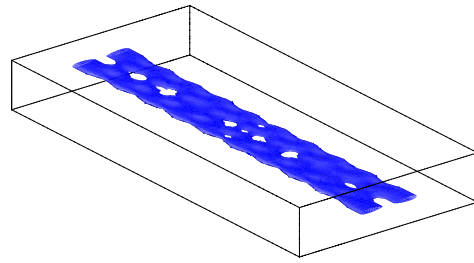
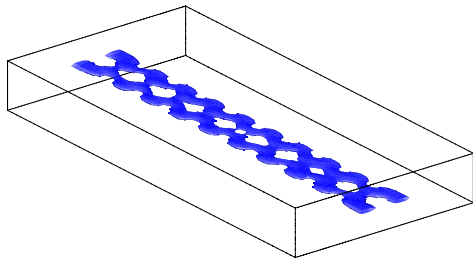
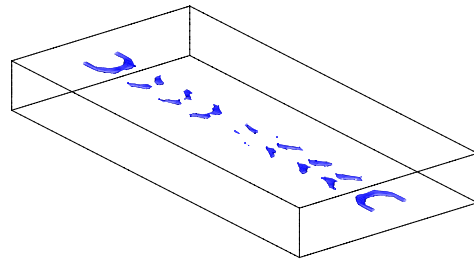
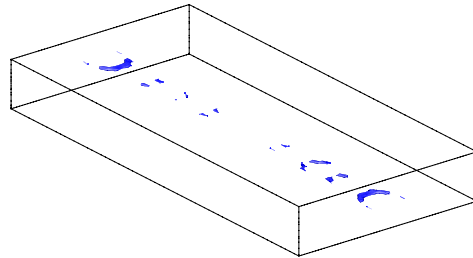
(a)  $t = 0$ .(b)  $t = 1.83 \times 10^{-2}$ .(c)  $t = 2.44 \times 10^{-2}$ .(d)  $t = 3.60 \times 10^{-2}$ .(e)  $t = 3.88 \times 10^{-2}$ .

Figure 6.5: Dissolution of a cementite plate with a sinusoidal perturbation.

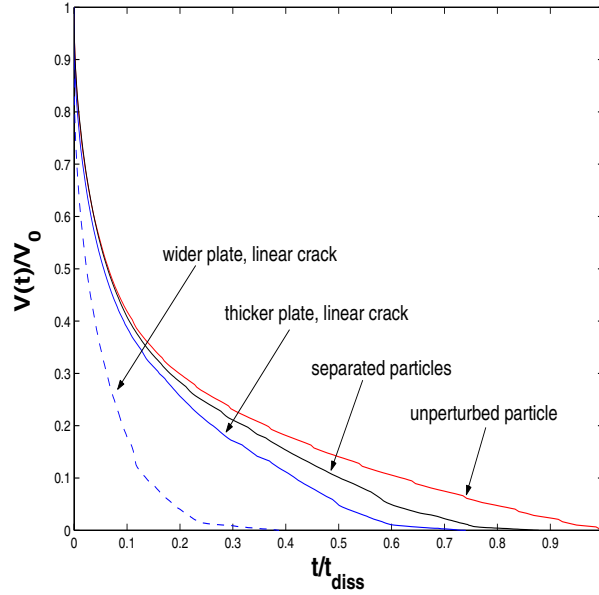


Figure 6.6: The volume of the cementite particle as a function of time for several perturbations on its surface.

### 6.2.3 The effect of the particle morphology on the dissolution kinetics

In this section, we study the influence of the morphology of the particle on the dissolution kinetics. To this end, we consider a family of  $\text{Mg}_2\text{Si}$  particles whose shapes vary from a circle to a very elongated ellipse dissolving in an aluminium alloy. The physical parameters of the problems are shown in Table 6.1. The diffusion coefficients  $D_{\text{Mg}}$  and  $D_{\text{Si}}$  and the solubility product  $\mathcal{K}(T)$  are given by the following Arrhenius relationships:

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

where  $R = 8.3144 \text{ J/Kmol}$  denotes the universal gas constant and  $T = 560^\circ\text{C}$  stands for the temperature at which the phase transformation takes place.

$c_{\text{Mg}}^{\text{part}} = 65\text{wt}\%$	$D_{\text{Mg}}^0 = 4.9 \cdot 10^7 \mu\text{m}^2/\text{s}$	$n_{\text{Mg}} = 2/3$
$c_{\text{Si}}^{\text{part}} = 35\text{wt}\%$	$D_{\text{Si}}^0 = 2.02 \cdot 10^8 \mu\text{m}^2/\text{s}$	$n_{\text{Si}} = 1/3$
$c_{\text{Mg}}^0 = 0\text{wt}\%$	$Q_{\text{Mg}}^d = 1.24 \cdot 10^5 \text{ J/mol}$	$k_0 = 89$
$c_{\text{Si}}^0 = 0.038\text{wt}\%$	$Q_{\text{Si}}^d = 1.36 \cdot 10^5 \text{ J/mol}$	$Q_s = 3.197 \cdot 10^4 \text{ J/mol}$

Table 6.1: Physical parameters for the dissolution of the  $\text{Mg}_2\text{Si}$  particles.

In order to compare the dissolution kinetics of the different particles, the volume fraction of the initial particle is fixed. To achieve this, all simulations are carried out over a square cell of area  $225 \mu\text{m}^2$  and the area of the initial particle is set equal to  $\pi \mu\text{m}^2$ . This corresponds with a circular particle of radius  $1 \mu\text{m}$  (curve I in the figures) or an ellipse for which its semimajor and semiminor

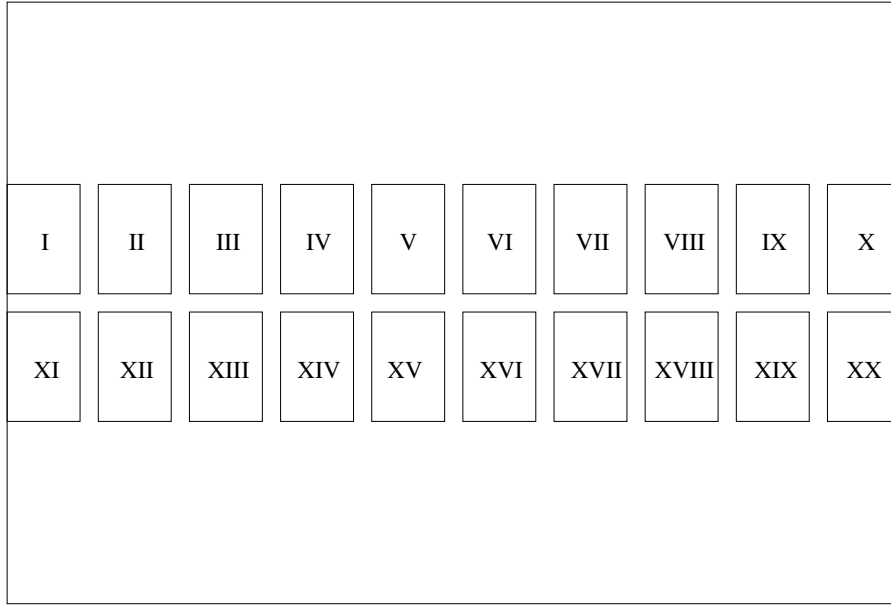


Figure 6.7: The configuration and numbering of the dissolving isolated particles at a cross-section at the plane  $z = 0.25\mu\text{m}$ . The bounding box denotes the boundary of the computational domain.

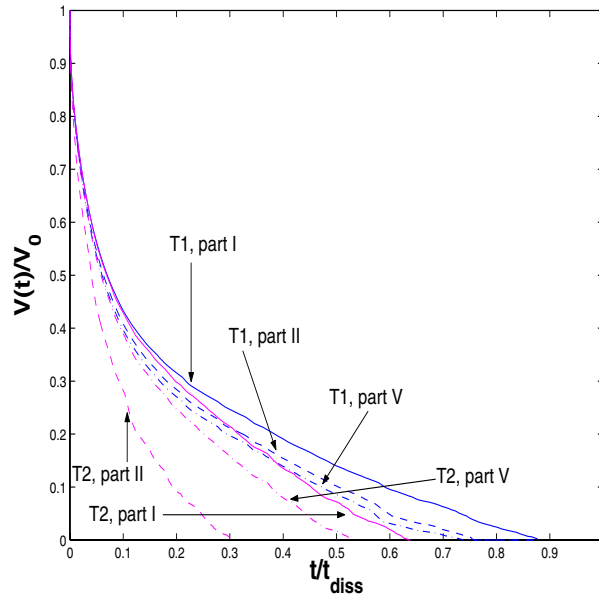


Figure 6.8: The volume of several individual cementite particles as a function of time for the case of 20 particles (T1) and for 16 particles in which particles III, VIII, XIII and XVIII have been removed (T2).

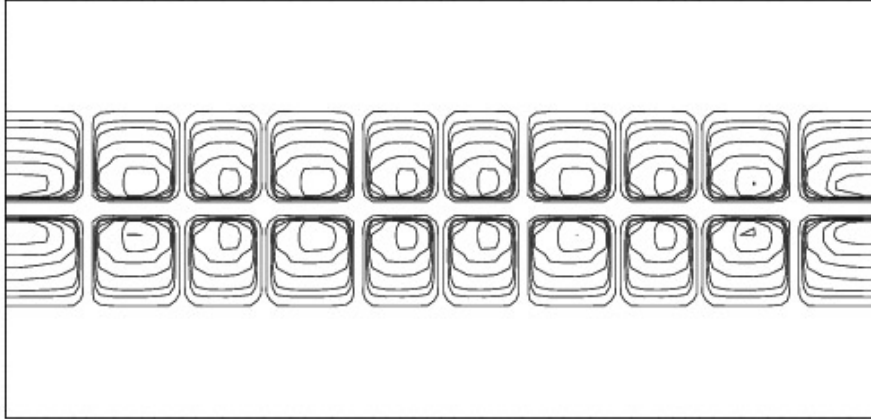


Figure 6.9: Interface profile at the cross section parallel to the  $xy$ -plane at  $z = 0.25$  for the 20 platelets (T1). Time evolution goes from the exterior towards the interior.

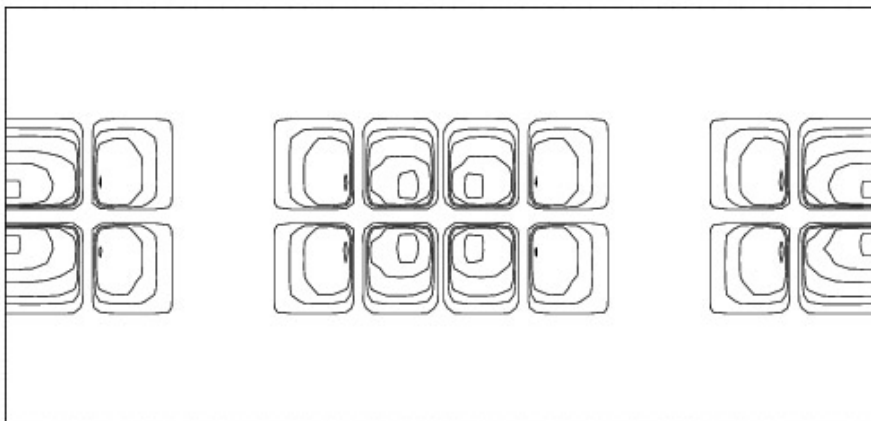


Figure 6.10: Interface profile at the cross section parallel to the  $xy$ -plane at  $z = 0.25$  for the 16 platelets (T2). Time evolution goes from the exterior towards the interior.



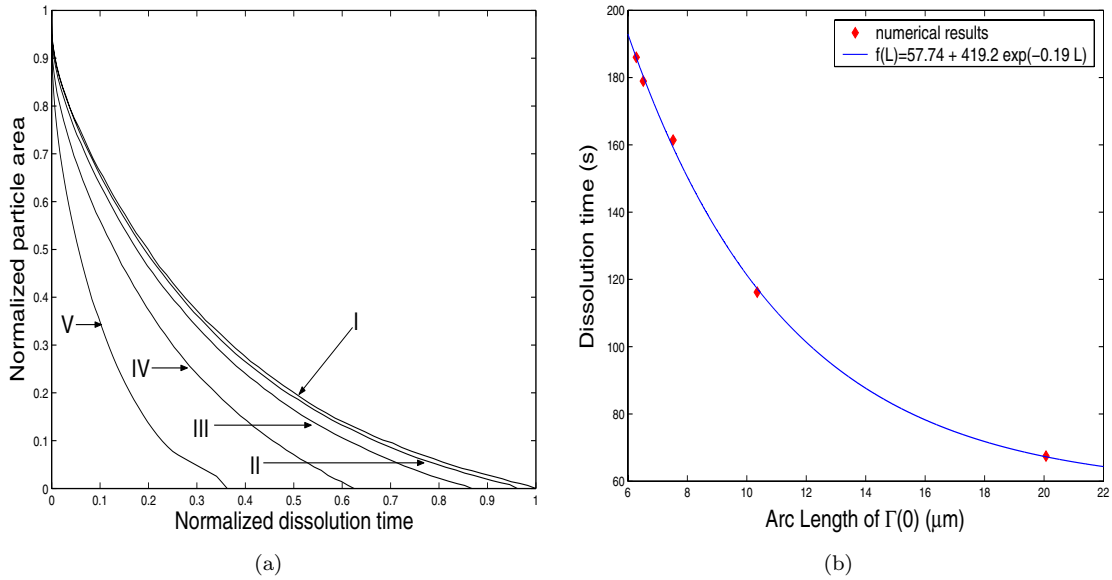


Figure 6.11: Dissolution of the  $\text{Mg}_2\text{Si}$  particles: (a) Normalized particle area vs normalized time, (b) dissolution time vs arc length of the initial interface.

axes  $a$  and  $b$  satisfy  $ab = 1\mu\text{m}^2$ . In our numerical experiments, we have used  $b = 0.8, 0.6, 0.4, 0.2\mu\text{m}$  (curves II, III, IV and V respectively in the figures).

The normalized area  $A(t)/A(0)$  of the  $\text{Mg}_2\text{Si}$  particles is presented in Figure 6.11(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  $t_{diss} = 186.04\text{s}$ . 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 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 Figure 6.11(b). The function  $f(L)$  approximating the dissolution time has been fitted to the numerical results with a least squares method.

In Figure 6.12 we present the time-history of 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 largely 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 an area of about 6 or 7 control volumes of our computational 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.

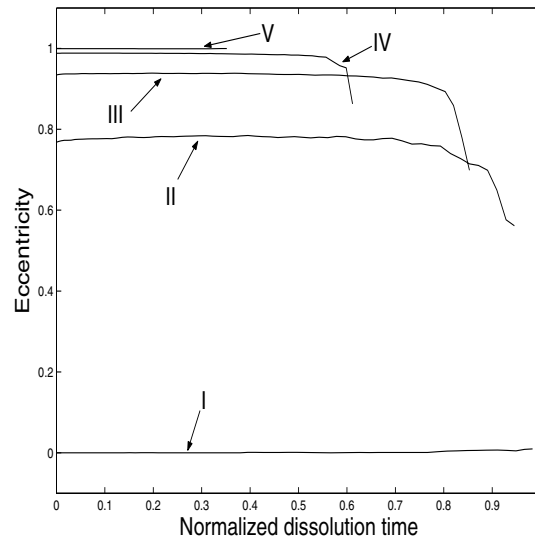


Figure 6.12: Dissolution of the  $\text{Mg}_2\text{Si}$  particles: eccentricity of the particle vs normalized dissolution time

### 6.3 Conclusions

A three-dimensional sharp interface model based on the level set method is shown to be able to deal with the complex local and macroscopic topological changes during solid state particle dissolution. The level set method based model offers a more realistic description of particle dissolution than the dissolution models enforcing shape conservation. Using physical parameters for cementite plates during austenization of a binary Fe-C alloy, the model illustrated the significant acceleration of the dissolution due to multiple fractures in the cementite plates in accordance with experimental and industrial experience. Furthermore, the model has been used to demonstrate experimentally that the morphology of the particle exerts a strong influence on the dissolution kinetics. The numerical results for the dissolution of  $\text{Mg}_2\text{Si}$  particles indicate an exponential acceleration of the dissolution process with the circumference of the initial particle.

---

---

# CHAPTER 7

---

## Models for nucleation and growth

### 7.1 Introduction

As pointed out in Chapter 1, precipitation of secondary phases may occur spontaneously or be metallurgically stimulated by the thermal treatment of the alloy (with an age-hardening treatment for instance). Size, shape and arrangement of the precipitates strongly influence the mechanical properties. In classical nucleation theory, the formation of new precipitates or particles is controlled by three different processes: nucleation, due to the clustering of atoms, growth, due to the diffusional transport of atoms, and coarsening, due to the reduction of the overall interfacial energy by the growth of largest particles at the expense of the smallest ones, *i.e.* Oswald Ripening. These three processes generally take place simultaneously [75, 76]. Robson [77] predicts the cases in which nucleation and coarsening proceed simultaneously or sequentially as a function of the interface energy and solute supersaturation. Although growth and coarsening may be modelled as a Stefan problem similar to those presented in Chapter 2, the early-stage phenomenon of nucleation is totally different and cannot be embedded in our model. Thus, two different models for compact simulation of nucleation, early growth and coarsening will be treated in Section 7.2. Subsequently, the growth of precipitates at later stages, modelled as a Stefan problem, will be tackled in Section 7.3.

### 7.2 Models of precipitation

Two different models of nucleation and early growth in supersaturated binary alloys are presented in this section: a particle size distribution model [75, 78] and a Monte Carlo method based on the exchange Ising model [79]. Both models deal with nucleation, growth and coarsening in a compact fashion. Particle distribution models predict the evolution of a system with spherical precipitates, for which the classical nucleation theory provides the nucleation and growth laws. On the other hand, Monte Carlo methods provide direct modelling at the atomic scale of the precipitation process. The diffusion mechanism plays an important role in the Monte Carlo simulations. A Monte Carlo method based on the exchange diffusion mechanism [79, 80, 81], also referred to as Kawasaki dynamics, exchange neighbouring atoms, whereas Monte Carlo methods based on the vacancy diffusion mechanism [82, 83, 84] only allow exchanges between atoms and neighbouring vacancies. The shape of the precipitates seems to be independent of the diffusion mechanism. However, Fratzl and Pernrose [85] showed that the asymptotic growth of the precipitates is approximated faster (*i.e.* using fewer Monte Carlo iterations) with the vacancy models. Liu *et al.* [86] present a Monte Carlo method which combines both diffusion mechanisms, although the atom-atom exchanges are admitted with lower probabilities.

### 7.2.1 Particle size distribution models

The Kampmann and Wagner [78] model is used to describe the precipitation of spherical particles in a supersaturated matrix. This model keeps track of the number of particles of different radius *classes* that are present in the system by means of a transport equation that couples the nucleation with the growth/dissolution of present particles in a simple fashion.

#### Nucleation of new particles

The number of stable heterogeneous nuclei that are formed in the system is given by [87]

$$j = j_0 \exp \left( - \left( \frac{A_0}{RT} \right)^3 \left( \ln \left( \bar{c}/c_e \right) \right)^{-2} \right) \exp \left( - \frac{Q_d}{RT} \right), \quad (7.1)$$

where  $j_0$  is a pre-exponential factor,  $A_0$  denotes the activation energy for nucleation,  $R$  the universal gas constant ( $R = 8.314 \frac{J}{K \text{ mol}}$ ),  $T$  the temperature,  $\bar{c}$  the mean concentration in the matrix,  $Q_d$  the activation energy for diffusion and  $c_e$  the equilibrium concentration at the particle/matrix interface obtained from the phase diagram.

#### Dissolution/growth of particles

The rate at which a spherical particle of radius  $r$  and concentration  $c_{part}$  grows or dissolves is approximated by

$$v = \frac{dr}{dt} = \frac{\bar{c} - c_i}{c_{part} - c_i} \frac{D}{r}, \quad (7.2)$$

where  $D$  is the diffusivity, and the interface concentration  $c_i$  is given by the Gibbs-Thomson relation

$$c_i = c_e \exp \left( \frac{2\sigma V_m}{rRT} \right), \quad (7.3)$$

where  $\sigma$  is the particle-matrix interface energy and  $V_m$  the molar volume of the particle. Note that plugging the curvature  $\kappa(\mathbf{x}, t) = \frac{1}{r}$  in Eq. (2.12) results into Eq. (7.3). The critical radius  $r^*$  at which the particle neither will grow nor dissolve (*i.e.*  $v = 0$ ) is given by

$$r^* = \frac{2\sigma V_m}{RT} \frac{1}{\ln \left( \bar{c}/c_e \right)}. \quad (7.4)$$

The particles with radius  $r < r^*$  will dissolve and are referred to as unstable. The particles with radius  $r > r^*$  will grow and hence are called stable. Furthermore, Eq. (7.4) shows that as solute is removed from the matrix into the precipitates (*i.e.*  $\bar{c}$  decreases towards  $c_e$ ), the critical radius  $r^*$  increases. Hence, larger particles will be unstable and dissolve as the average matrix concentration decreases.

#### Particle size distribution

A particle size distribution function  $\varphi = \varphi(r)$  is introduced to describe the density of particles as function of the radius. A number of radius "classes" of length  $\Delta r$  are defined in the computational cell. Thus, particles of radius within  $r - \frac{\Delta r}{2}$  and  $r + \frac{\Delta r}{2}$  belong to the class of particles of radius  $r$ . The number of particles per  $m^3$  and per radius class is given by  $N = \varphi \Delta r$ . The particle volume fraction  $f$  is computed by

$$f = \sum_i \frac{4}{3} \pi r_i^3 N_i, \quad (7.5)$$

and the mean concentration in the matrix is given by  $\bar{c} = \frac{c_0 - c_{part}f}{1-f}$ , where  $c_0$  denotes the concentration in the matrix.

The evolution of the system is given by the mass balance

$$\frac{\partial N}{\partial t} = -\frac{\partial(Nv)}{\partial r} + S, \quad (7.6)$$

where the source term  $S$  describes the formation of new particles. Because only particles with radius larger than the critical radius  $r^*$  are stable, the source term  $S$  is defined as follows

$$S(r) := \begin{cases} j, & \text{if } 1.05r^* - \frac{\Delta r}{2} < r \leq 1.05r^* + \frac{\Delta r}{2}, \\ 0, & \text{otherwise,} \end{cases} \quad (7.7)$$

A growth radius 5% larger than the critical radius is imposed on the nucleated particles to initiate the growth reaction [87]. We assume that below this threshold, the nucleated particles will dissolve immediately. Note that the source  $S$  defined in (7.7) is time-dependent because  $r^*$  is. The continuity equation (7.6) is discretized with a Backward Euler method in time and a first order upwind scheme in space.

### Numerical results and discussion

The precipitation of  $\text{Mg}_2\text{Si}$  particles in an industrial Al-Mg-Si alloy is simulated here. The ternary alloy is simplified into a *binary* Al-Mg<sub>2</sub>Si alloy, where Mg is considered to be the solute in the Al-rich matrix. The interface equilibrium concentration  $c_e$  is derived from the quasi-binary Al-Mg<sub>2</sub>Si section of the ternary Al-Mg-Si phase diagram, which yields the Arrhenius relation

$$c_e = c_s \exp\left(-\frac{Q_s}{RT}\right).$$

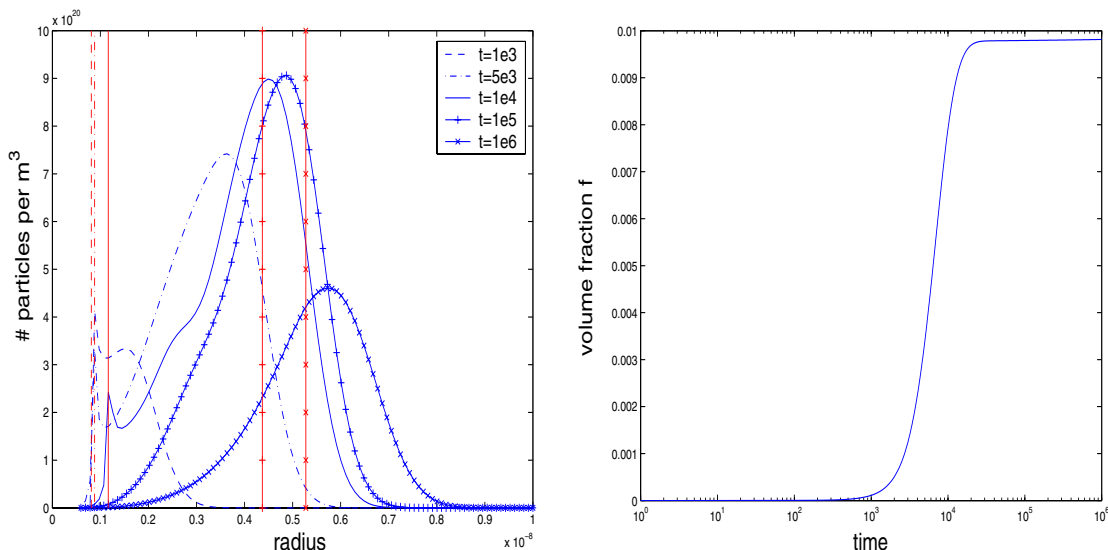
The parameters of the problem, taken from [87], are listed in Table 7.1.

$j_0 = 9,66 \times 10^{34} \frac{\#particles}{m^3s}$	$A_0 = 16220 \frac{J}{mol}$	$Q_d = 1.5 \times 10^{-5} \frac{J}{mol}$	$T = 453 K$
$c_{part} = 63.4 \text{ wt}\%$	$c_0 = 0.63 \text{ wt}\%$	$\sigma = 0.2 \frac{J}{m^2}$	$V_m = 3.95 \times 10^{-5} \frac{m^3}{mol}$
	$c_s = 970 \text{ wt}\%$	$Q_s = 47175 \frac{J}{mol}$	

Table 7.1: Parameters for the particle distribution model.

In the computational approach, only classes of radius above  $r_0 = 5\text{\AA}$  are considered. Radii below this threshold give rise to undefined interface concentrations Eq. (7.3) and interface velocities Eq. (7.2). The maximum radius class is  $R_{max} = 100\text{\AA}$ , and the computational grid is divided in 100 radius classes, *i.e.*  $\Delta r = 0.95\text{\AA}$ . A fixed time-stepping  $\Delta t = 10^2$  is used in the computation of the numerical solution. Furthermore, the initial particle size distribution  $\varphi$  follows a normal density distribution whose mean is the initial critical radius  $r^* \approx 8.09\text{\AA}$  and whose variance is  $2\text{\AA}$ .

Figure 7.1(a) shows the number density of particles  $N$  at various times. The horizontal lines represent the critical nucleation radius  $r^*$  at the corresponding times. Note that from Eq. (7.4),  $r^*$  increases as the mean concentration  $\bar{c}$  decreases, *i.e.* larger particles become unstable as solute is removed from the matrix. Figure 7.1(b) shows the volume fraction  $f$  as a function of time. The number density of particles is increasing until approximately  $t = 10^5$ . This corresponds to the period of nucleation. The steep gradient in the volume fraction between  $t = 3 \cdot 10^2$  and  $t = 3 \cdot 10^4$  indicates that nucleation and growth are happening simultaneously. For  $t \leq 10^4$ , the number of particles that dissolve, *i.e.* those whose radius is less than the critical radius, is a small fraction of the total number of particles. Thus, coarsening is negligible at this stage of precipitation



(a) The number density of particles  $N$  (blue curves) and the critical radius (horizontal red lines) at various times.

(b) The volume fraction  $f$  as a function of time.

Figure 7.1: Results for the particle size distribution model

process. However, between  $t = 10^4$  and  $t = 10^5$ , the number of particles is still increasing, but a large number of them are below the critical radius and will dissolve. Therefore, nucleation and coarsening are taking place simultaneously. From  $t = 10^5$  until the final time  $t = 10^6$  the number of particles is decreasing and the volume fraction is stabilized. Therefore, coarsening is the only process taking place in the evolution of the system.

## 7.2.2 Monte Carlo models

A two-dimensional Monte Carlo method, based on the Ising model, is used to simulate precipitation of particles in a quasi-binary Al-Mg<sub>2</sub>Si. We implement a direct exchange Ising model, which is more in agreement with interstitial diffusion mechanisms than with substitutional diffusion mechanisms. Since the atomic sizes of Al, Mg and Si are comparable, this assumption may seem unrealistic. However, since the Al-Mg<sub>2</sub>Si alloy is treated as a binary alloy, we consider that one molecule of Mg<sub>2</sub>Si is one 'atom' of solute in the Al-rich phase, and hence we can fairly assume that the transport of atoms occur due to interstitial mechanisms. Ferrite and austenite are examples of binary alloys where the transport of atoms is due to interstitial diffusion. Substitutional diffusion mechanisms can be dealt with by the so-called vacancy mechanism models [82, 83, 84].

Consider a two-dimensional square lattice with  $N = L \times L$  sites and periodic boundary conditions. In this lattice, a binary alloy AB is defined. In this case A=Al and B=Mg<sub>2</sub>Si. Each site on the lattice is occupied either by an A-atom or by a B-atom. The dynamics of the nucleation model are described with a simple conserved-order-parameter Ising model [88], in which the order-parameter is the alloy composition. One can postulate the Hamiltonian of the system as

$$H = -J \sum_{\langle ij \rangle} s_i s_j, \quad (7.8)$$

where  $\langle ij \rangle$  indicates that over nearest-neighbours is summed once,  $J$  denotes the interaction energy between nearest-neighbours and the variable  $s$  is defined as follows:  $s_i = 1$  if the site  $i$  is occupied by an A-atom, and  $s_i = -1$  if it is occupied by a B-atom. Note that alike pairs of neighbours contribute  $-J$  to the total energy, whereas unlike pairs contribute  $J$ . Thus, the stable

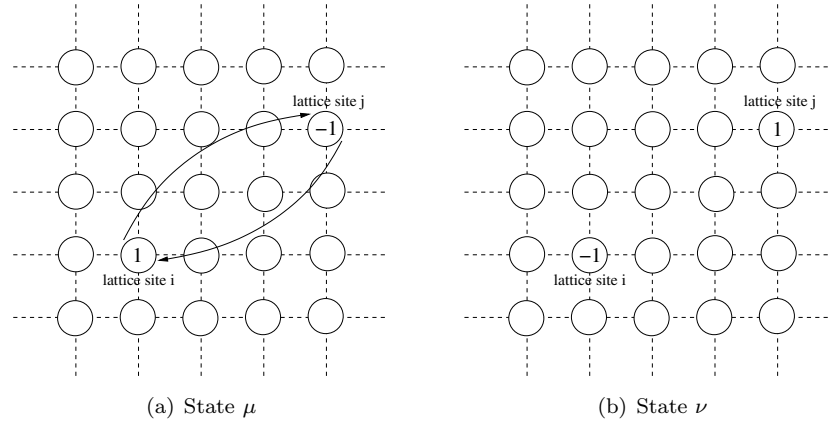


Figure 7.2: Arrangement of the lattice for two consecutive states  $\mu$  and  $\nu$ .

state of the system will be that in which all the alike atoms are clustered together in such a way that the length of interface separating the clusters is minimized.

It is easy to prove, after simple mathematical manipulations, that the Hamiltonian can be written as

$$H = -J \sum_{i=1}^N n_i + \text{constant}, \quad (7.9)$$

where  $n_i$  denotes the number of nearest neighbours of  $i$  that contain atoms of the same species (*i.e.* the spin coordination number).

The transition from an arrangement of the atoms  $\mu$  to a successive arrangement  $\nu$  is given the Metropolis acceptance probability

$$A(\mu \rightarrow \nu) = \begin{cases} \exp\left(-\frac{H_\nu - H_\mu}{k_B T}\right), & \text{if } H_\nu - H_\mu > 0, \\ 1, & \text{otherwise,} \end{cases}$$

where  $k_B$  denotes the Boltzmann's constant and  $T$  the temperature. At low temperatures, the mobility of the atoms is lower and hence reaching equilibrium states (stables or metastables) takes longer times. In the Monte Carlo model this means that most of the exchanges are rejected, and hence many iterations are wasted. Since only equilibrium states are to be simulated here, a continuous time algorithm [88] which allows non-local exchanges is implemented in order to speed up the performance of the Monte Carlo method. A similar algorithm is used in [84]. Two consecutive states of the system  $\mu$  and  $\nu$  will only differ in two lattice sites  $i$  and  $j$  where the exchange of atoms is carried out, as sketched in Figure 7.2. The lattice sites  $i$  and  $j$  are chosen randomly among the list of A-atoms and B-atoms with probabilities proportional to  $\exp\left(-\frac{2n_i}{k_B T}\right)$  and  $\exp\left(-\frac{2n_j}{k_B T}\right)$  respectively. Subsequently, the exchange of atoms in  $i$  and  $j$  is accepted, and the elapsed time  $\Delta t$  in this transformation is measured by

$$\Delta t^{-1} = \sum_{q=0}^4 m_{A,q} \exp\left(-\frac{2q}{k_B T}\right) \cdot \sum_{r=0}^4 m_{B,r} \exp\left(-\frac{2r}{k_B T}\right), \quad (7.10)$$

where  $m_{A,q}$  denotes the number of A-atoms having  $q$  nearest-neighbours A-atoms, and  $m_{B,r}$  is defined similarly.

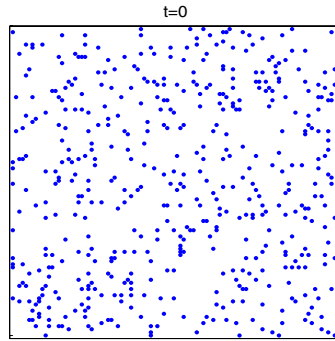
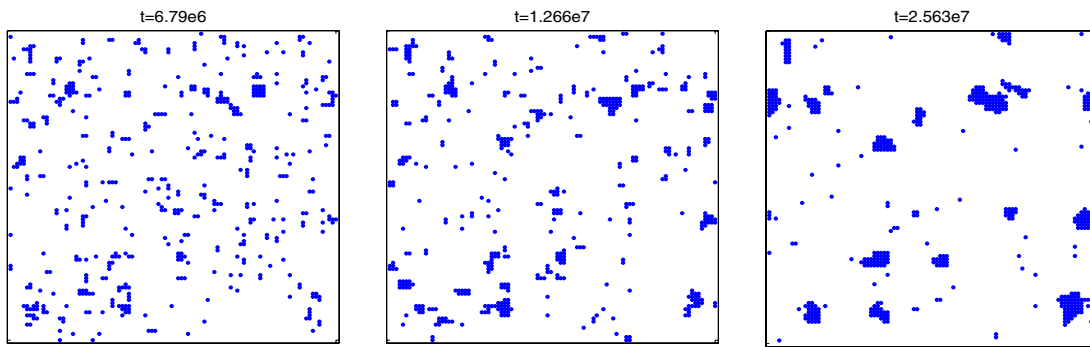


Figure 7.3: Initial arrangement of the atoms.



(a) Arrangement of the atoms after 550 MCS.

(b) Arrangement of the atoms after 1550 MCS.

(c) Arrangement of the atoms after 6700 MCS.

Figure 7.4: Snapshots of the precipitation process: nucleation, growth and coarsening respectively.

Figure 7.3 shows the initial arrangement of atoms in the Al-Mg<sub>2</sub>Si alloy under study, in which the concentration of Al is  $c = 0.95\%$  and the bounding energy  $J = 1.2k_B T$ . Figure 7.4 shows the snapshots of the systems at the three stages of the precipitation process: nucleation of a large number of small particles in Figure 7.4(a), growth of particles in Figure 7.4(b) and finally coarsening in Figure 7.4(c), since most of the smaller particles in Figure 7.4(b) have dissolved and the larger ones have grown. The time of these snapshots is given both in the precipitation time computed with Eq. (7.10) and in number of Monte Carlo steps (1 MCS = 1 exchange of atoms).

The evolution of the system can be observed in more detail in Figure 7.5. The number of particles present in the system is plotted as a function of the normalized time in Figure 7.5(a). A particle has been arbitrarily defined as a cluster with four or more B-atoms. The number of particles on the system is increasing, with the typical fluctuations due to the probabilistic nature of the model, until approximately  $t = 0.3t_{end}$ . One could argue that this must be the time at which nucleation stops. Figure 7.4(a) corresponds to  $t = 0.28t_{end}$ . Subsequently, the number of particles remains oscillating around 40 for the period of time between  $0.3t_{end}$  and approximately  $0.5t_{end}$ . However, the (normalized) volume of these particles, presented in Figure 7.5(b), is increasing. Therefore, this corresponds to pure growth of the precipitates, with negligible coarsening. Figure 7.4(b) corresponds to  $t = 0.49t_{end}$ . From  $0.5t_{end}$  until the end of the simulation, the number of particles decreases whereas their volume increases. Thus, pure coarsening is taking place. Figure 7.4(c) corresponds to  $t = 0.99t_{end}$ .



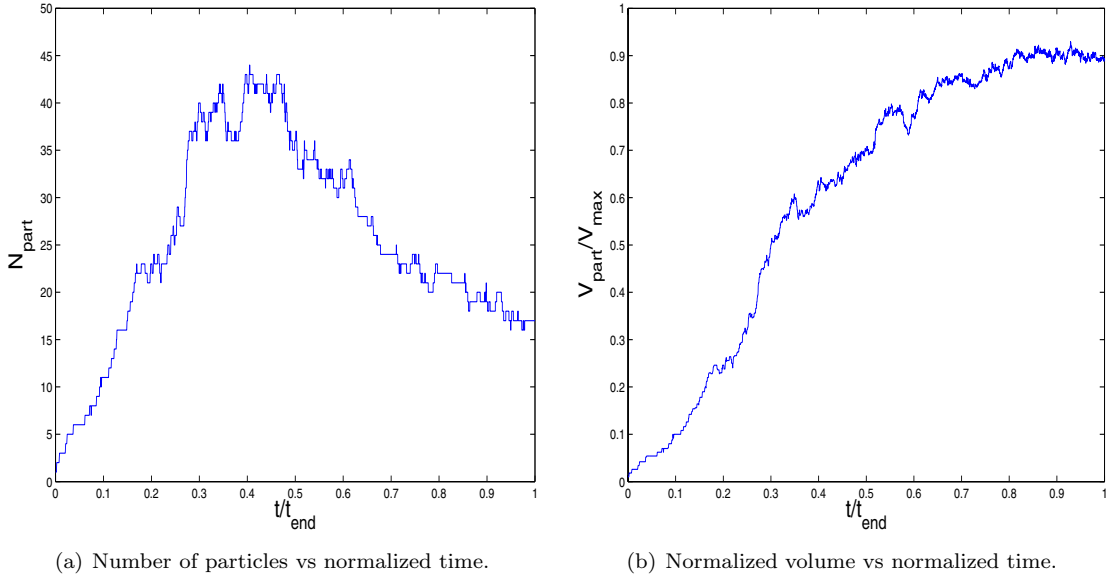


Figure 7.5: Number of particles and volume with the Monte Carlo method.

### 7.3 Growth of precipitates

Growth of particles is governed by the same principles as dissolution. Thus, the governing equations presented in Chapter 2 and the Level Set method developed in Chapter 4 can be directly used. For the sake of the presentation, only a two-dimensional binary alloy is considered, and the Cartesian extension of the front velocity Eq. (4.5) is used. The only difference with the algorithm presented in Chapter 2 lies on the discretization of the space derivatives in Eq. (2.2), which is done with Finite Difference schemes instead of with Finite Element methods. The reason for this is that combination of the Gibbs-Thomson effects Eq. (2.12) with the cut-cell method has not been implemented in SEPRAN at the current time. However, it represents a fairly straightforward extension of the code. Therefore, for mesh points away from the interface, central differences are used. For the mesh points adjacent to the interface, the discretizations need to be locally adapted to the interface. Let  $\mathbf{x}_{ij}$  denote a mesh point adjacent to the interface that has its west and south neighbours inside the particle. In this case, the following approximation is used [55]

$$\Delta c_{ij} = P_2''[\mathbf{x}_{fj}, \mathbf{x}_{ij}, \mathbf{x}_{i+1j}](\mathbf{x}_{ij}) + P_2''[\mathbf{x}_{if}, \mathbf{x}_{ij}, \mathbf{x}_{ij+1}](\mathbf{x}_{ij}) \quad (7.11)$$

where  $P_2$  denotes the quadratic Lagrangian interpolation polynomial in one-dimension built from the concentration at the points given in brackets, where  $\mathbf{x}_{fj}$  (resp.  $\mathbf{x}_{if}$ ) denotes the interface point between  $\mathbf{x}_{i-1j}$  and  $\mathbf{x}_{ij}$  (resp.  $\mathbf{x}_{ij-1}$  and  $\mathbf{x}_{ij}$ ) obtained from linear interpolation of the level set function. Similar expressions are obtained for all the possible combinations of north/south/east/west neighbours inside the particle.

An initial diamond-like particle is located in the center of the computational domain  $\Omega = [-0.5, 0.5]^2$ . The parameters of the problem are

- the particle concentration  $c^{part}=0.45$  wt%,
- the initial concentration  $c^0=0.3$  wt% in the diffusive phase,
- the equilibrium concentration  $c^{sol}=0.1$  wt%,
- the diffusivity  $D = 1$  m/s<sup>2</sup>,

- and the distance from the center of the particle to the corners,  $d = 0.1 m$ .

Figure 7.6(a) shows the interface at various times, when the curvature effect Eq. (2.12) is not taken into account. Finger-like patterns appear in the corners of the precipitate, which evolve according to the morphological instability described by Mullins-Sekerka [89, 90]. This instability is inherent to the problem, however it is triggered by the numerical errors made in the interface approximation and by the cusped initial shape of the interface. When the interfacial concentration is given by the Gibbs-Thomson relation Eq. (2.12) a more stable growth of the particle is observed, see Figures 7.6(b) and 7.6(c), especially for larger values of the surface tension. The local curvature  $\kappa$  of the interface is computed from the level set function Eqs. (4.3) and (4.4), that for two-dimensional problems results into

$$\kappa = \nabla \cdot \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) = \frac{\phi_y^2 \phi_{xx} - 2\phi_x \phi_y \phi_{xy} + \phi_x^2 \phi_{yy}}{(\phi_x^2 + \phi_y^2)^{3/2}}. \quad (7.12)$$

Central differences are used to compute  $\kappa$  in the mesh points adjacent to the interface. The curvature at the interface points is subsequently obtained by linear interpolation in the Cartesian directions. The curvature of the initial interface is zero everywhere except at the corners, where  $\kappa > 0$ . In fact,  $\kappa \rightarrow \infty$  in the corners. Thus, the initial interface concentration will be

$$c_{\max}^{sol}(\mathbf{x}, 0) = \begin{cases} c^{sol}, & \text{if } \mathbf{x} \text{ is not on a corner,} \\ c^{sol} \exp(2\vartheta \kappa(\mathbf{x}, 0)) > c^{sol}, & \text{if } \mathbf{x} \text{ is one of the corners.} \end{cases}$$

Thus, depending on the constant  $\vartheta$ , the interface concentration  $c^{sol}(\mathbf{x}, t)$  will stay below  $c^0$  or will exceed it. In the first case, the interface will move outwards, with a slower velocity at the corners. In the second case, corners will move inwards and the rest of interface will move outwards, see Figure 7.6(c). Both cases eventually result into a smooth circular-like particle, with approximately constant curvature and therefore constant velocity in the normal direction. It is observed that the outer boundaries have a large influence in the results. For the three cases, a square-like structure is obtained as the interface approaches the outer boundaries.

## 7.4 Conclusions

This chapter has been dedicated to the study of two other processes occurring during homogenization of industrial aluminium alloys: the precipitation and growth of particles. Two models of precipitation have been described. Particle size distribution models [87] resolve nucleation, growth and coarsening of particles in an elegant and compact fashion. However, their applicability to more complex systems is so far limited, since all the particles in the system are assumed to be spherical in order to make use of the nucleation law predicted from theory. On the other hand, Monte Carlo methods [79] overcome some of these geometrical restrictions by dealing with the precipitation process at the atomic scale. Needle-shaped particles can be induced using uniaxial interactions or elastic misfits. Moreover, their extension to multi-component alloys can be implemented in a rather straightforward manner. When equilibrium solutions are to be simulated, efficient algorithms that allow non-local exchanges, as the continuous time algorithm [88] or the residence-time algorithm [84], are to be implemented

The growth of individual particles can be dealt with the Level Set method developed in Chapters 4 and 5. The Gibbs-Thomson effect must be incorporated to stabilize the numerical results. The curvature in the mesh points adjacent to the interface is computed from the level set function, and subsequently interpolated in the Cartesian directions to the interface.

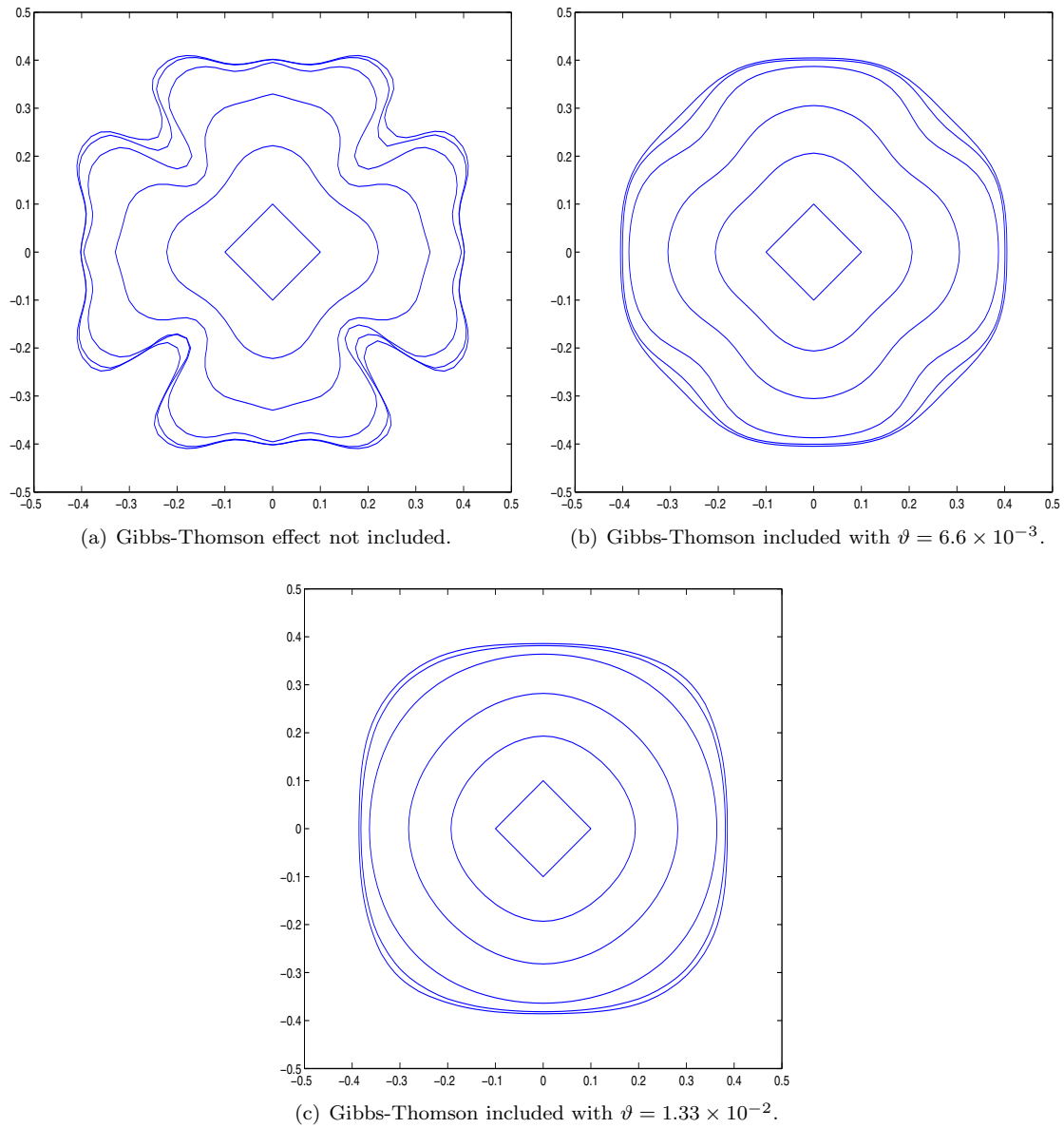


Figure 7.6: Growth of a diamond-like particle. Snapshots of the interface position at times 0, 0.01, 0.025, 0.05, 0.075 and 0.1.



---

---

# CHAPTER 8

---

## Conclusions

### 8.1 Conclusions

The purpose of this work is to develop a numerical method to solve vector Stefan problems in higher dimensions, with application to dissolution of particles in multi-component alloys. The level set method is preferred to a moving grid method because it handles topological changes in a natural fashion. The phase field method is not used because of the adaptive mesh techniques required and the prohibitive time step sizes which are necessary in order to solve the interface region with a sufficiently high resolution and, furthermore, because of lack of knowledge concerning the large amount of parameters involved in the phase field formulation. However, the choice of the level set method still involves a number of mathematical and numerical challenges which have to be overcome in order to provide reliable computer simulations.

First, a continuous extension of the front velocity needs to be generated over the computational domain at each time step in order to carry out the interface advection. Two alternative extension procedures have been studied in this work. The extrapolation of the front velocity in the normal direction is simple and of low computational cost for dissolution of particles in binary alloys if the interface concentration is constant. However, for varying interface concentrations, occurring when the particle dissolves in a multi-component alloy or when curvature effects are added, its implementation is neither simple nor computationally cheap. Contrarily, the extension of the Cartesian components of the front velocity independently in the proper upwind direction demands the same complexity and amount of work for constant and varying interface concentrations, thanks to the reconstruction of the interface made by the cut-cell method.

Second, the level set function is kept '*close*' to a signed distance function within the regions near the moving interface, in order to prevent flat or steep gradients near the interface. Distance functions are characterized by the norm of their gradient being equal to one. The level set function is only reinitialized whenever the norm of its gradient differs from one more than 5% in a band around of the interface. Thus, how often the level set function is reinitialized depends on the width of this critical band around the interface, but also on the movement of the interface. Furthermore, it is well known that reinitialization causes a shift of the interface. In order to avoid that mesh points adjacent to the interface change phase because of reinitialization of the level set function, a modified reinitialization equation [58] is solved with high order discretizations.

Third, the background finite element mesh needs to be adapted to the interface position each time step, in order to implement the interface conditions of the diffusion problem(s). The cut-cell

method gives an elegant linear reconstruction of the front, enriching the background basis with interface points and dividing the intersected elements according to a predefined rule. The 'enriched' mesh is only used to solve the diffusion problem(s) and subsequently to extend the front velocity. After that, the interface points are disregarded and the background triangulation is recovered.

Fourth, the nonlinearly coupled problem for particle dissolution in multi-component alloys needs to be solved with an iterative method. Due to the workload per iteration, a fixed-point method with relaxation is chosen. The definition of the fixed-point iteration strongly depends on the physical problem to be solved, and the relaxation parameters  $\delta_i$  depend, moreover, on the geometry of the particle. In this work, the parameters  $\delta_i$  were chosen to be constant (in time and space) throughout the calculations, and the initial guess of the interface concentrations is the average of the interface concentrations at the previous time step. These choices give convergence, but result in a poor convergence rate, specially for three-dimensional problems.

Fifth, the discontinuity jump in the concentrations across the interface demands severe measures to solve the diffusion problem(s) accurately whenever mesh points that were initially located inside the particle appear for the first time inside the diffusive phase. In those cases, the discretization of the diffusion equation(s) has to be adapted locally, considering that the time integration starts at the time at which the interface passes through the mentioned mesh points. Only the time discretization needs to be adjusted, because the backward Euler method is used. For binary alloys, this adaptation is easily achieved, because the interface concentration is a known constant. However, for multi-component alloys this adaptation is not straightforward, because the interface concentration(s) at the time that the interface hits the mesh points are unknown and furthermore, its determination would present a very laborious task closely related to Lagrangian methods for the advection of interfaces. Instead, the time-stepping is chosen such that the cut-cell method will locate interface points in the mesh points added to the diffusive phase in the new time step. In this way we explicitly avoid dealing with the time derivative of the concentration(s) on the conflicting mesh points, using instead the Dirichlet condition for the interface points.

The model is applied successfully to the dissolution of a pearlitic structure of cementite in three dimensions. Using physical parameters for cementite plates during austenization of a binary Fe-C alloy, our simulations indicate a significant acceleration of dissolution due to multiple fractures in the cementite plate, which is in accordance with experimental and industrial experience.

## 8.2 Recommendations for future research

In order to solve vector Stefan problems efficiently, the fixed-point algorithm proposed in this thesis should be optimized. Improvements can be made in several points. First, the initial guess of the interface concentrations should be defined such that local variations of the interface concentrations due to the geometry of the particle are captured. For instance, for a given interface point, one might prescribe the concentrations at the previous time step of the closest mesh point, or an average of the concentrations at the neighbouring mesh points. These would bring the initial guess closer to the actual solution, reducing considerably the number of iterations needed to converge. Second, the parameters  $\delta_i$  should be increased with time. Our experience is that the initialization step is the most critical, *i.e.* requires  $\delta_i$  to be small, because of the initial discontinuous concentration profiles. After some stabilization time,  $\delta_i$  might be increased to speed up the convergence. Third, a pre-processing algorithm might be developed to smooth out the normal fluxes  $\frac{\partial c_i}{\partial \mathbf{n}}$  across the interface and therefore to reduce the influence on the convergence of the fixed-point iteration of the most narrow interface elements.

In order to be able to simulate stable growth of particles, the interface curvature needs to be computed to determine the interface concentration(s) with the Gibbs-Thomson effect. Direct

computation of the curvature after linearization of the interface made by the cut-cell method might be toilsome, especially for three-dimensional problems. Instead, the level set function can be used to define the curvature  $\kappa(\mathbf{x}, t) = \nabla \cdot \frac{\nabla \phi}{\|\nabla \phi\|}$  at the mesh points adjacent to the interface, and from those points the curvature can be interpolated to the interface points.

The code can be extended easily to include several interfaces (one interface for each class of particles) being present at the same time. One level set function should be defined for each interface. Therefore, the extension of the front velocity, the advection of the interface and the reinitialization should be carried out independently for each level set function. The diffusive phase would be given as the region where all the level set functions are positive, and the cut-cell method should be applied to define the intersection of the background mesh with all the interfaces.

A relevant metallurgical application that could be simulated, if both curvature effects and multiple interfaces are implemented, is the growth of an  $\alpha$ -particle lying on the rim of a  $\beta$ -plate [1]. This application presents a mathematical challenge, because of the triple points that appear in the contact area.

Another interesting feature that could be added to the code is an adaptive grid refinement around the moving interface. This would lower the computational requirements to capture the movement of the interface accurately. However, due to the combination with the finite difference schemes used for the level set equations, this grid refinement should be done such that the background mesh points coincide with the Cartesian grid. Otherwise, interpolation between the two meshes should be used.





---



---

# APPENDIX A

---

## Convergence analysis of the fixed-point iteration in the vector Stefan problem using the similarity solution of the planar interface

### The similarity solution for the planar interface

The similarity solution for the dissolution of a planar interface in a multi-component alloy will be used here to analyze the convergence of the fixed-point iteration (5.4). We will prove that the definition of the function  $g$  should be based on the physical parameters of the problem, and we will give an upper bound for the convergence rate. Let us assume that we deal with a ternary alloy (*i.e.*  $p = 2$ ) and  $\tilde{n}_1 = \tilde{n}_2 = 1$ . According to [14], the analytical solution of the dissolution of a planar interface in an unbounded domain is given by

$$c_i(x, t) = \begin{cases} c_i^{part}, & \text{if } x < s(t), \\ c_i^{sol}, & \text{if } x = s(t), \\ c_i^0 + \frac{c_i^{sol} - c_i^0}{\operatorname{erfc}\left(\frac{\alpha}{2\sqrt{D_i}}\right)} \operatorname{erfc}\left(\frac{x - s_0}{2\sqrt{D_i}t}\right), & \text{if } x > s(t), \end{cases} \quad (\text{A.1})$$

for  $i = 1, 2$ , where the interface position is given by  $s(t) = s_0 + \alpha\sqrt{t}$ , and  $\alpha$ , and  $c_1^{sol}$  and  $c_2^{sol}$  are the solution of

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

$$c_1^{sol} c_2^{sol} = \mathcal{K}(T).$$

### The scalar fixed-point iterations

The hyperbolic relation (2.8) is used here to eliminate  $c_1^{sol}$  from the system, *i.e.*

$$c_1^{sol} = c_1^{sol}(c_2^{sol}) = \frac{\mathcal{K}(T)}{c_2^{sol}}. \quad (\text{A.3})$$

The Stefan conditions (2.11) on the interface are used to define  $g$  as one of the following

$$g_1(c_2^{sol}) = c_2^{sol} + \delta \left( \frac{D_2}{c_2^{part} - c_2^{sol}} \frac{\partial c_2}{\partial x}(s(t), t) - \frac{D_1}{c_1^{part} - c_1^{sol}(c_2^{sol})} \frac{\partial c_1(c_2^{sol})}{\partial x}(s(t), t) \right), \quad (\text{A.4})$$

$$g_2(c_2^{sol}) = c_2^{sol} + \delta \left( D_2 (c_1^{part} - c_1^{sol}(c_2^{sol})) \frac{\partial c_2}{\partial x}(s(t), t) - D_1 (c_2^{part} - c_2^{sol}) \frac{\partial c_1(c_2^{sol})}{\partial x}(s(t), t) \right), \quad (\text{A.5})$$

where  $\delta$  denotes a relaxation parameter. Assuming that  $c_i^{part} \gg c_i^{sol}$  [14], it is easy to prove, after simple mathematical manipulations, that

$$\begin{aligned} g_1(c_2^{sol}) &= c_2^{sol} + \frac{\delta}{\sqrt{\pi t}} H_1(c_2^{sol}), \\ H_1(c_2^{sol}) &= -\sqrt{D_2} \frac{c_2^{sol} - c_2^0}{c_2^{part} - c_2^{sol}} + \sqrt{D_1} \frac{\frac{\mathcal{K}(T)}{c_2^{sol}} - c_1^0}{c_1^{part} - \frac{\mathcal{K}(T)}{c_2^{sol}}}, \\ H_1'(c_2^{sol}) &= -\sqrt{D_2} \frac{c_2^{part} - c_2^0}{(c_2^{part} - c_2^{sol})^2} - \sqrt{D_1} \frac{\mathcal{K}(T)(c_1^{part} - c_1^0)}{(c_1^{part} c_2^{sol} - \mathcal{K}(T))^2}, \end{aligned} \quad (\text{A.6})$$

and

$$\begin{aligned} g_2(c_2^{sol}) &= c_2^{sol} + \frac{\delta}{\sqrt{\pi t}} H_2(c_2^{sol}), \\ H_2(c_2^{sol}) &= -\sqrt{D_2} \left( c_1^{part} - \frac{\mathcal{K}(T)}{c_2^{sol}} \right) (c_2^{sol} - c_2^0) + \sqrt{D_1} (c_2^{part} - c_2^{sol}) \left( \frac{\mathcal{K}(T)}{c_2^{sol}} - c_1^0 \right), \\ H_2'(c_2^{sol}) &= -\sqrt{D_2} \left( c_1^{part} - \mathcal{K}(T) \frac{c_2^0}{(c_2^{sol})^2} \right) - \sqrt{D_1} \left( \mathcal{K}(T) \frac{c_2^{part}}{(c_2^{sol})^2} - c_1^0 \right). \end{aligned} \quad (\text{A.7})$$

The following theorem gives sufficient conditions for the existence and uniqueness of a fixed-point for  $g$ , and an upper bound of the error.

**Theorem:**

If  $g : [a, b] \rightarrow [a, b]$  is a continuous, differentiable function, and

$$|g'(x)| \leq \beta < 1$$

for  $x \in (a, b)$ , then the fixed-point iteration  $p_n = g(p_{n-1})$  converges to the unique fixed point  $p$  of  $g$  in  $[a, b]$  for any  $p_0 \in [a, b]$ .

Moreover,  $|p_n - p| \leq \beta |p_{n-1} - p| \leq \beta^n |p_0 - p|$ .

*Proof:* See [91].

Hence, the parameter  $\beta$  is an indicator of the convergence rate. For  $\beta \ll 1$  the convergence is fast, whereas for  $\beta \approx 1$  the convergence is slow.

Next, the convergence of the fixed-point iteration defined by  $g_1$  is analyzed. Note that  $|g'(x)| \leq \beta < 1$  is equivalent to

$$-\beta - 1 < \frac{\delta}{\sqrt{\pi t}} H_1'(c_2^{sol}) < \beta - 1. \quad (\text{A.8})$$

The function  $H_1'$  is defined in the interval  $(\frac{\mathcal{K}(T)}{c_1^{part}}, c_2^{part})$ . Note that  $c_2^{sol} \rightarrow c_2^{part}$  when  $\frac{D_2}{D_1} \rightarrow 0$  (i.e.  $D_1 \gg D_2 \rightarrow 0$  or  $D_2 \ll D_1 \rightarrow \infty$ ), whereas  $c_2^{sol} \rightarrow \frac{\mathcal{K}(T)}{c_1^{part}}$  (i.e.  $c_1^{sol} \rightarrow c_1^{part}$ ) when  $\frac{D_2}{D_1} \rightarrow \infty$ .

Moreover,  $H'_1$  is continuous, differentiable and upper and lower bounded on any compact interval  $[a, b]$  contained in  $(\frac{\mathcal{K}(T)}{c_1^{part}}, c_2^{part})$ . Let us denote by  $m$  and  $M$  the minimum and the maximum of  $H'_1$  in  $[a, b]$ . Then, equation (A.8) is equivalent to

$$-\beta - 1 < \frac{\delta}{\sqrt{\pi t}} m < \frac{\delta}{\sqrt{\pi t}} M < \beta - 1. \quad (\text{A.9})$$

Note that  $H'_1$  is negative, hence  $m \leq M < 0$ . Rearranging terms in the previous equation yields

$$\frac{(1 + \beta)\sqrt{\pi t}}{-m} > \delta > \frac{(1 - \beta)\sqrt{\pi t}}{-M}, \quad (\text{A.10})$$

and from this equation it is easy to prove that

$$\beta > \left| \frac{M - m}{M + m} \right|. \quad (\text{A.11})$$

Relation (A.10) gives an upper and lower bound for the relaxation parameter  $\delta$  in the case that the convergence rate  $\beta$  can be reached. Relation (A.11) establishes a lower bound for  $\beta$  which depends on the maximum and minimum of  $H'_1$  on the interval  $[a, b]$  where the solution is sought.

Note that  $\left| \frac{M - m}{M + m} \right| < 1$ .

To conclude, the two fixed point methods proposed above are compared. The idea is to study the lower bound (A.11) for both methods, in order to elucidate which one is more efficient. The parameters  $D_2 = 1$ ,  $c_{1,2}^{part} = 15$ ,  $c_{1,2}^0 = 0$  and  $\mathcal{K}(T) = 1$  are fixed, whereas  $D_1$  is varied. Hence, the fixed point  $c_2^{sol}$  of  $g_1$  and  $g_2$  will vary as well. The interval  $[a, b]$  where the lower bound of the convergence rate  $\beta$  Eq. (A.11) is computed is  $[c_2^{sol} - 0.1, c_1^{sol} + 0.1]$  in all the cases. The results are presented in Table A.1. The conclusion is that the fixed-point iteration defined by  $g_2$  is more efficient than that defined by  $g_1$  when the fixed point  $c_2^{sol}$  is closer to  $\mathcal{K}(T)/c_1^{part}$  and  $c_2^{part}$ , because both  $\mathcal{K}(T)/c_1^{part}$  and  $c_2^{part}$  are poles of  $H'_1$ . Note that  $H'_2$  has a pole at 0 as well, but  $\mathcal{K}(T)/c_1^{part} > 0$ . Moreover, the convergence rate of both methods is decreased considerably when  $D_1 \rightarrow 0$  since  $\left| \frac{M - m}{M + m} \right| \rightarrow 1$ . On the other hand,  $g_1$  is very efficient when the solution  $c_2^{sol}$  is far from  $\mathcal{K}(T)/c_1^{part}$  and  $c_2^{part}$ .

$D_1$	$c_2^{sol}$	$g_1$	$g_2$
$10^{-3}$	0.191	$9.39 \times 10^{-1}$	$5.55 \times 10^{-1}$
$10^{-1}$	0.571	$2.09 \times 10^{-1}$	$1.75 \times 10^{-1}$
10	1.751	$4.73 \times 10^{-2}$	$5.81 \times 10^{-2}$
$10^3$	5.228	$6.89 \times 10^{-4}$	$2.05 \times 10^{-2}$
$10^5$	13.054	$9.89 \times 10^{-2}$	$9.95 \times 10^{-3}$

Table A.1: The lower bound  $\left| \frac{M - m}{M + m} \right|$  of the fixed-point iterations defined by  $g_1$  and  $g_2$  for different locations of the solution  $c_2^{sol}$ .



---

# BIBLIOGRAPHY

- [1] N. Kuijpers. *Kinetics of the  $\beta$ -AlFeSi to  $\alpha$ -Al(FeMn)Si transformation in Al-Mg-Si alloys*. PhD Thesis, Delft University of Technology, 2004.
- [2] M.J. Whelan. On the kinetics of particle dissolution. *Metal Sci. J.*, 3:95–97, 1969.
- [3] U.L. Baty, R.A. Tanzilli, and R.W. Heckel. Dissolution kinetics of CuAl<sub>2</sub> in Al-4Cu alloy. *Metall. Trans.*, 1:1651–1656, 1970.
- [4] U.H. Tundal and N. Ryum. Dissolution of particles in binary alloys: Part I, Computer simulations. *Metall. Trans.*, 23.
- [5] F.V. Nolfi jr., P.G. Shewmon, and J.S. Foster. The dissolution and growth kinetics of spherical particles. *Trans. Metall. Soc. AIME*, 245:1427–1433, 1969.
- [6] H.B. Aaron and G.R. Kotler. Second phase dissolution. *Metall. Trans.*, 2(2):393–407, 1971.
- [7] D.A. Porter and K.E. Easterling. *Phase transformations in metals and alloys*. Chapman & Hall, London, 1992.
- [8] J. Svododa, F.D. Fischer, P. Fratzl, E. Gamjäger, and N.K. Simha. Kinetics of interfaces during diffusional transformations. *Acta Mater.*, 49:1249–1259, 2001.
- [9] O. Reiso, N. Ryum, and J. Strid. Melting and dissolution of secondary phase particles in AlMgSi-alloys. *Metall. Trans. A*, 24A:2629–2641, 1993.
- [10] J. Ågren. Diffusion in phases with several components and sublattices. *J. Phys. Chem. Solids*, 43:421–430, 1981.
- [11] J.M. Vitek, S.A. Vitek, and S.A. David. Modelling of diffusion controlled phase transformation in ternary systems and application to the ferrite-austenite transformation in the Fe-Cr-Ni system. *Metall. Trans. A*, 26A:2007–2025, 1995.
- [12] F.J. Vermolen, C. Vuik, and S. van der Zwaag. The dissolution of a stoichiometric second phase in ternary alloys: a numerical analysis. *Mater. Sci. Engn. A*, 246:93–103, 1998.
- [13] F.J. Vermolen and C. Vuik. A numerical method to compute the dissolution of second phases in ternary alloys. *J. Comp. Appl. Math.*, 93:123–143, 1998.
- [14] F.J. Vermolen and C. Vuik. A mathematical model for the dissolution of particles in multi-component alloys. *J. Comp. Appl. Math.*, 126:233–254, 2000.

- [15] F.J. Vermolen, C. Vuik, and S. van der Zwaag. Particle dissolution and cross-diffusion in multi-component alloys. *Mater. Sci. Eng. A*, 347:265–279, 2003.
- [16] F.J. Vermolen and C. Vuik. Solution of vector Stefan problems with cross-diffusion. *J. Comp. Appl. Math.*, 176:179–201, 2005.
- [17] J. Crank. *Free and moving boundary problems*. Clarendon Press, Oxford, 1984.
- [18] W.D. Murray and F. Landis. Numerical and machine solutions of transient heat-conduction problems involving melting or freezing. *Trans. ASME (C)*, 81:106–112, 1959.
- [19] G. Segal, C. Vuik, and F. Vermolen. A conserving discretization for the free boundary in a two-dimensional Stefan problem. *J. Comp. Phys.*, 141:1–21, 1998.
- [20] D. Juric and G. Tryggvason. A front-tracking method for dendritic solidification. *J. Comp. Phys.*, 123:127–148, 1996.
- [21] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comp. Phys.*, 79:12, 1988.
- [22] S. Chen, B. Merriman, S. Osher, and P. Smereka. A simple level set method for solving Stefan problems. *J. Comp. Phys.*, 135:8–29, 1997.
- [23] F. Gibou, R. Fedkiw, R. Caflish, and S. Osher. A level set approach for the numerical simulation of dendritic growth. *J. Sci. Comp.*, 19(1):183–199, 2003.
- [24] Y.-T. Kim, N. Goldenfeld, and J. Dantzig. Computation of dendritic microstructures using a level set method. *Phys. Rev. E*, 62:2471–2474, 2000.
- [25] A.A. Wheeler, W.J. Boettinger, and G.B. McFadden. Phase-field model for isothermal phase transitions in binary alloys. *Phys. Rev. A*, 45:7424–7439, 1992.
- [26] I. Kovačević and B. Šarler. Solution of a phase-field model for dissolution of primary particles in binary aluminum alloys by an r-adaptive mesh-free method. *Mater. Sci. Engn. A*, pages 423–428, 2005.
- [27] U. Grafe, B. Böttger, Tiaden J., and S.G. Fries. Coupling of multicomponent thermodynamics to a phase field model: application to solidification and solid-state transformations of superalloys. *Scripta Mater.*, 42:1179–1186, 2000.
- [28] 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.*, 169:708–759, 2001.
- [29] J.A. Mackenzie and M.L. Robertson. A Moving Mesh Method for the solution of the One-Dimensional Phase-Field Equations. *J. Comp. Phys.*, 181:526–544, 2002.
- [30] A. Schmidt. Computational of three dimensional dendrites with finite elements. *J. Comp. Phys.*, 125:293, 1996.
- [31] E. Burman, M. Picasso, and J. Rappaz. Analysis and computation of dendritic growth in binary alloys using a phase-field model. In M. Feistauer, V. Dolejší, P. Knobolch, and K Najzar, editors, *Numerical Mathematics and Advanced Applications. Proceedings of ENUMATH 2003*, pages 204–220, Berlin, 2004. Springer.
- [32] G. Caginalp. Stefan and Hele-Shaw type models as asymptotics of the phase-field equations. *Phys. Rev. A*, 39:5887–5896, 1989.
- [33] C. Vuik, G. Segal, and F.J. Vermolen. A conserving discretization for a Stefan problem with an interface reaction at the free boundary. *Comput. Visual Sci.*, 3:109–114, 2000.

- [34] M. Fabbri and V.R. Voller. The Phase-Field Method in the Sharp-Interface Limit: a comparison between model potentials. *J. Comp. Phys.*, 130:256–265, 1997.
- [35] C. Vuik and C. Cuvelier. Numerical solution of an etching problem. *J. Comp. Phys.*, 59:247–263, 1985.
- [36] J. Goodman and D.N. Ostrov. On the early exercise boundary of the American put option. *SIAM J. Appl. Math.*, 62:1823–1835, 2002.
- [37] G.W. Evans. A note on the existence of a solution to a problem of Stefan. *Q. Appl. Math.*, 9:185–193, 1951.
- [38] J. Douglas. A uniqueness theorem for the solution of a Stefan problem. *Proc. Amer. Math. Soc.*, 8:402–408, 1957.
- [39] J.M. Hill. *One-dimensional Stefan problems: an introduction*. Longman Scientific & Technical, Harlow, 1987.
- [40] W.J. Minkowycz and E.M. Sparrow. *Advances in Numerical Heat Transfer*, volume 1. Taylor & Francis, Washington, 1997.
- [41] C.M. Elliott and J.R. Ockendon. *Weak and variational methods for moving boundary problems*. Pitman Press, Boston, 1982.
- [42] V.R. Voller and M. Cross. Accurate Solutions of Moving boundary Problems using the Enthalpy Method. *Int. J. Heat Mass Transfer*, 24:545–556, 1981.
- [43] V.R. Voller. An implicit enthalpy solution for phase change problems: with application to a binary alloy solidification. *Appl. Math. Modelling*, 11:110–116, 1987.
- [44] B. Nedjar. An enthalpy-based finite element method for nonlinear heat problems involving phase change. *Comput. Struct.*, 80:9–21, 2002.
- [45] Y.C. Lam, J.C. Chai, P. Rath, H. Zheng, and V.M. Murukeshan. A fixed-grid method for chemical etching. *Int. Comm. Heat Mass Transfer*, 31:1123–1131, 2004.
- [46] J.A. Sethian. *Level set methods and fast marching methods*. Cambridge University Press, New York, 1999.
- [47] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag, New York, 2003.
- [48] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comp. Phys.*, 114:146–159, 1994.
- [49] D. Adalsteinsson and J.A. Sethian. The fast construction of extension velocities in level set methods. *J. Comp. Phys.*, 148:2–22, 1999.
- [50] R. Kobayashi. Melting and numerical simulations of dendritic growth. *Physica D*, 63:410–423, 1993.
- [51] G. Caginalp and E.A. Socolovsky. Computation of Sharp Phase Boundaries by Streaking: The Planar and Spherically Symmetric Cases. *J. Comp. Phys.*, 95:85–100, 1991.
- [52] S.-L. Wang, R.F. Sekerka, A.A. Wheeler, B.T. Murray, S.R. Coriell, R.J. Braun, and G.B. McFadden. Thermodynamically-consistent phase-field models for solidification. *Physica D*, 69:189–200, 1993.
- [53] G. Caginalp and J.T. Lin. A numerical analysis of an Anisotropic Phase Field Model. *IMA J. Appl. Math.*, 39:51–66, 1987.

- [54] W. Huang, Y. Ren, and D.R. Russell. Moving Mesh Partial Differential Equations (MMPDES) based on the equidistribution principle. *SIAM J. Numer. Anal.*, 31:709–730, 1994.
- [55] E. Javierre, C. Vuik, F.J. Vermolen, and S. van der Zwaag. A comparison of numerical models for one-dimensional Stefan problems. *J. Comp. Appl. Math.*, 192:445–459, 2006.
- [56] D. Adalsteinsson and J.A. Sethian. A fast Level Set Method for propating interfaces. *J. Comp. Phys.*, 118:269–277, 1995.
- [57] G. Russo and P. Smereka. A remark in computing distance functions. *J. Comp. Phys.*, 163:51–67, 2000.
- [58] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A PDE-Based Fast Local Level Set Method. *J. Comp. Phys.*, 155:410–438, 1999.
- [59] H.K. Zhao. A fast sweeping method for Eikonal equations. *Math. Comp.*, 74:603–627, 2004.
- [60] H.K. Zhao, S. Osher, B. Merriman, and M. Kang. Implicit and non-parametric shape reconstruction from unorganized data using a variational level set method. *Computer Vision and Image Understanding*, 80:295–319, 2000.
- [61] Y.R. Tsai. Rapid and accurate computation of the distance function using grids. *J. Comp. Phys.*, 178:175–195, 2002.
- [62] J. Chessa, P. Smolinski, and T. Belytschko. The extended finite element method (XFEM) for solidification problems. *Int. J. Numer. Meth. Engng.*, 53:1959–1977, 2002.
- [63] 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.
- [64] S. Osher and R.P. Fedkiw. Level set methods: an overview and some recent results. *J. Comp. Phys.*, 169:463–502, 2001.
- [65] G.S. Jiang and D. Peng. Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM J. Sci. Comp.*, 21:2126–2143, 2000.
- [66] G. Segal. *SEPRAN introduction*. Ingenieursbureau Sepr, Leidschendam, The Netherlands, 1995.
- [67] 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*, 47:339–361, 2005.
- [68] E. Javierre, C. Vuik, F.J. Vermolen, A. Segal, and S. van der Zwaag. The Level Set method for solid-solid phase transformations. In A. Bermúdez de Castro, D. Gómez, P. Quintela, and P. Salgado, editors, *Numerical Mathematics and Advanced Applications. Proceedings of ENUMATH 2005*, pages 712–719, Berlin, 2006. Springer.
- [69] F.J. Vermolen, C. Vuik, E. Javierre, and S. van der Zwaag. Review on some Stefan problems for particle dissolution in solid metallic alloys. *Nonlinear Analysis: Modelling and Control*, 10(3):257–292, 2005.
- [70] W.D. Callister. *Materials science and engineering, an introduction, 4<sup>th</sup> edition*. John Wiley and Sons, New York, 1997.
- [71] C. Atkinson, T. Akbay, and R.C. Reed. Theory for reaustenization from ferrite-cementite mixtures in Fe-C-X steels. *Acta Metall. Mater.*, 43 (5):2013–2031, 1995.
- [72] L. Zhao, F.J. Vermolen, A. Wauthier, and J. Sietsma. Cementite Dissolution at 860 °C in an Fe-Cr-C Steel. *Metall. Mater. Trans. A*, 37A:1841–, 2006.



- [73] K.E. Thelning. *Steel and its heat treatment*. Butterworths, London, 1975.
- [74] D. Hernández-Silva, R.D. Morales, and J.G. Cabañas-Moreno. The spheroidization of cementite in a medium carbon steel by means of subcritical and intercritical annealing. *ISIJ International*, 32:1297–1305, 1992.
- [75] J.S. Langer and A.J. Schwartz. Kinetics of nucleation in near-critical fluids. *Phys. Rev. A*, 21(3):948–958, 1980.
- [76] H.I. Aaronson and F.K. LeGoues. An Assessment of Studies on Homogeneous Diffusional Nucleation Kinetics in Binary Metallic Alloys. *Metall. Trans. A*, 23(7):1915–1945, 1992.
- [77] J.D. Robson. Modelling the overlap of nucleation, growth and coarsening during precipitation. *Acta Mater.*, 52:4669–4676, 2004.
- [78] R. Wagner and R. Kampmann. *Materials Science and Technology, A Comprehensive Treatment*, volume 5. VCH, Weinheim, 1991.
- [79] J.-M. Liu. Dynamics of directional coarsening in binary alloys: Monte Carlo simulation. *J. Mater. Sci.*, 32:1765–1773, 1997.
- [80] J.-M. Liu. Precipitation in binary alloys with anisotropic interaction: a Monte-Carlo approach. *J. Mater. Sci. Lett.*, 14:1734–1737, 1995.
- [81] P. Fratzl and O. Penrose. Ising model for phase separation in alloys with anisotropic elastic interactions - II. A computer experiment. *Acta Mater.*, 44(8):3227–3239, 1996.
- [82] K. Yaldram and K. Binder. Spinodal decomposition of a two-dimensional model alloy with mobile vacancies. *Acta Metall. Mater.*, 39(4):707–717, 1991.
- [83] P. Fratzl and O. Penrose. Kinetics of spinodal decomposition in the Ising model with vacancy diffusion. *Phys. Rev. B*, 50(5):3477–3480, 1994.
- [84] F. Soisson, A. Barbu, and G. Martin. Monte Carlo simulations of Copper precipitation in dilute Iron-Copper alloys during thermal ageing and under electron irradiation. *Acta Mater.*, 44(9):3789–3800, 1996.
- [85] P. Fratzl and O. Penrose. Competing mechanisms for particle coarsening in phase separation with vacancy dynamics. *Phys. Rev. B*, 55(10):R6101–R6104, 1997.
- [86] L.M. Liu, Z.G. Liu, Z.C. Wu, X.L. Guo, and X.K. Meng. A Monte Carlo approach of phase separation in binary alloys with mobile vacancies. *Mater. Lett.*, 22:23–27, 1995.
- [87] O.R. Myhr and Ø. Grong. Modelling of non-isothermal transformations in alloys containing a particle distribution. *Acta Mater.*, 48:1605–1615, 2000.
- [88] M.E.J. Newman and G.T. Barkema. *Monte Carlo methods in statistical physics*. Clarendon Press, Oxford, 1999.
- [89] W.W. Mullins and R.F. Sekerka. Morphological stability of a particle growing by diffusion or heat flow. *J. Appl. Phys.*, 34(2):323–329, 1963.
- [90] W.W. Mullins and R.F. Sekerka. Stability of a Planar Interface During Solidification of a Dilute Binary Alloy. *J. Appl. Phys.*, 35(2):444–451, 1963.
- [91] R.L. Burden and J.D. Faires. *Numerical analysis*. Brooks/Cole, Australia, 2001.



---

# List of Publications

## Refereed Journals

- E. Javierre, C. Vuik, F.J. Vermolen and A. Segal, A Level Set Method for Three Dimensional Vector Stefan Problems: Simulations of Particle Dissolution in Multi-component Alloys, *Journal of Computational Physics*, submitted.
- F.J. Vermolen, E. Javierre, C. Vuik, L. Zhao, S. van der Zwaag, A Three Dimensional Model for Particle Dissolution in Binary Alloys, *Computational Materials Science*, to appear.
- E. Javierre, C. Vuik, F.J. Vermolen, S. van der Zwaag, A comparison of numerical models for one-dimensional Stefan problems, *Journal of Computational and Applied Mathematics*, vol. 192, pages 445-459, 2006.
- F.J. Vermolen, C. Vuik, E. Javierre, S. van der Zwaag, Review on some Stefan Problems for Particle Dissolution in Solid Metallic Alloys, *Nonlinear Analysis. Modelling and control*, vol. 10(3), 2005.

## Proceedings

- E. Javierre, C. Vuik, F.J. Vermolen, A. Segal and S. van der Zwaag, Higher dimensional numerical simulations of precipitate dissolution in multi-component aluminium alloys, In P. Wesseling, E. Oñate and J. Périaux, editors, *Proceedings European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2006*, CD-ROM, The Netherlands, 2006, TU Delft.
- E. Javierre, C. Vuik, F.J. Vermolen, A. Segal and S. van der Zwaag, The Level Set Method for Solid-Solid Phase Transformations, In A. Bermúdez de Castro, D. Gómez, P. Quintela and P. Salgado, editors, *Numerical Mathematics and Advanced Applications. Proceedings of ENUMATH 2005*, pages 712-719, Berlin, 2006, Springer.

## Reports

- E. Javierre , C. Vuik, F.J. Vermolen, A. Segal, *A level set method for particle dissolution in a binary alloy*, DIAM TU Delft, R 05-07
- E. Javierre, C. Vuik, F.J. Vermolen, S. van der Zwaag, *A comparison of numerical models for one-dimensional Stefan problems*, DIAM TU Delft, R 05-03
- E. Javierre-Pérez, *Literature Study: Numerical methods for solving Stefan problems*, DIAM TU Delft, R 03-16

## Talks

- *Higher dimensional numerical simulations of precipitate dissolution in multi-component aluminium alloys*. European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006, September 5-8, 2006, Egmond aan Zee, The Netherlands.
- *Numerical solution of a three-dimensional vector Stefan problem arising in homogenization of Aluminum alloys*. 7th World Congress on Computational Mechanics, July 16-22, 2006, Los Angeles
- *Numerical solution of higher dimensional vector Stefan problems. Simulation of precipitate dissolution*. 42e Nederlands Mathematisch Congres, March 27-28, 2006, Delft.
- *Numerical simulation of precipitate dissolution in multi-component aluminium alloys*. Third BRICKS Scientific Computing Meeting, February 16, 2006, Amsterdam.
- *The Level-Set method for Stefan models in Homogenization of Aluminum Alloys*. Poster presentation in The Thirtieth Conference of the Dutch&Flemish Numerical Analysis Communities, October 12-14, 2005, Zeist, The Netherlands.
- *The Level Set Method for Solid-Solid Phase Transformations*. The Sixth European Conference on Numerical Mathematics and Advanced Applications, ENUMATH 2005, July 18-22, 2005, Santiago de Compostela, Spain.
- *The Level Set Method for solid-solid phase transformations*. Interphase 2004, Numerical Methods for Free Boundary Problems, 13-16 September 2004, Rome.

---

# Curriculum Vitae

- September 2002 - September 2006, Delft University of Technology, PhD student in the Delft Institute of Applied Mathematics, section of Scientific Computing, granted by the Dutch Technology Foundation (STW), project: *Vector Stefan models for the homogenization in aluminium alloys*, promotor Prof.dr.ir P. Wesseling (TUD/DIAM).
- November 2001 - June 2002, Department of Statistical Methods, Faculty of Science, University of Zaragoza, research grant in the field of computational geometry.
- September 1997 - July 2002, University of Zaragoza, MSc degree in Mathematics, speciality of Applied Mathematics.
- September 1993 - June 1997, High School IES Ramón J. Sender, Fraga, Spain.
- September 1984 - June 1993, Primary School CP Ramón J. Sender, Belver de Cinca, Spain.
- Born on 29-08-1979, Almudafar (province of Huesca), Spain.