

Computational Materials Engineering

An Introduction to Microstructure Evolution

Koenraad Janssens, Ph.D. • George Frans, Ph.D.

Dierk Raabe, Ph.D. • Britta Nestler, Ph.D.

Ernst Kozeschnik, Ph.D. • Mark A. Miodownik, Ph.D.



COMPUTATIONAL MATERIALS ENGINEERING

An Introduction to Microstructure Evolution

This page intentionally left blank

COMPUTATIONAL MATERIALS ENGINEERING

An Introduction to Microstructure Evolution

KOENRAAD G. F. JANSSENS
DIERK RAABE
ERNST KOZESCHNIK
MARK A. MIODOWNIK
BRITTA NESTLER




ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Academic Press is an imprint of Elsevier



Elsevier Academic Press
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
525 B Street, Suite 1900, San Diego, California 92101-4495, USA
84 Theobald's Road, London WC1X 8RR, UK

This book is printed on acid-free paper. 

Copyright © 2007, Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone: (+44) 1865 843830, fax: (+44) 1865 853333, E-mail: permissions@elsevier.co.uk. You may also complete your request on-line via the Elsevier homepage (<http://elsevier.com>), by selecting "Customer Support" and then "Obtaining Permissions."

Library of Congress Cataloging-in-Publication Data

Computational materials engineering: an introduction to microstructure evolution/editors Koenraad G. F. Janssens ... [et al.].

p. cm.

Includes bibliographical references and index.

ISBN-13: 978-0-12-369468-3 (alk. paper)

ISBN-10: 0-12-369468-X (alk. paper)

1. Crystals--Mathematical models. 2. Microstructure--Mathematical models. 3. Polycrystals--Mathematical models. I. Janssens, Koenraad G. F., 1968-

TA418.9.C7C66 2007

548'.7--dc22

2007004697

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library

ISBN 13: 978-0-12-369468-3

ISBN 10: 0-12-369468-X

For all information on all Elsevier Academic Press publications

visit our Web site at www.books.elsevier.com

Printed in the United States of America

07 08 09 10 11 12 10 9 8 7 6 5 4 3 2 1

Working together to grow
libraries in developing countries

www.elsevier.com | www.bookaid.org | www.sabre.org

ELSEVIER

BOOK AID
International

Sabre Foundation

to Su

This page intentionally left blank

Table of Contents

Preface	xiii
1 Introduction	1
1.1 Microstructures Defined	1
1.2 Microstructure Evolution	2
1.3 Why Simulate Microstructure Evolution?	4
1.4 Further Reading	5
1.4.1 On Microstructures and Their Evolution from a Noncomputational Point of View	5
1.4.2 On What Is Not Treated in This Book	6
2 Thermodynamic Basis of Phase Transformations	7
2.1 Reversible and Irreversible Thermodynamics	8
2.1.1 The First Law of Thermodynamics	8
2.1.2 The Gibbs Energy	11
2.1.3 Molar Quantities and the Chemical Potential	11
2.1.4 Entropy Production and the Second Law of Thermodynamics	12
2.1.5 Driving Force for Internal Processes	15
2.1.6 Conditions for Thermodynamic Equilibrium	16
2.2 Solution Thermodynamics	18
2.2.1 Entropy of Mixing	19
2.2.2 The Ideal Solution	21
2.2.3 Regular Solutions	22
2.2.4 General Solutions in Multiphase Equilibrium	25
2.2.5 The Dilute Solution Limit—Henry’s and Raoult’s Law	27
2.2.6 The Chemical Driving Force	28
2.2.7 Influence of Curvature and Pressure	30
2.2.8 General Solutions and the CALPHAD Formalism	33
2.2.9 Practical Evaluation of Multicomponent Thermodynamic Equilibrium	40

3	Monte Carlo Potts Model	47
3.1	Introduction	47
3.2	Two-State Potts Model (Ising Model)	48
3.2.1	Hamiltonians	48
3.2.2	Dynamics (Probability Transition Functions)	49
3.2.3	Lattice Type	50
3.2.4	Boundary Conditions	51
3.2.5	The Vanilla Algorithm	53
3.2.6	Motion by Curvature	54
3.2.7	The Dynamics of Kinks and Ledges	57
3.2.8	Temperature	61
3.2.9	Boundary Anisotropy	62
3.2.10	Summary	64
3.3	Q -State Potts Model	64
3.3.1	Uniform Energies and Mobilities	65
3.3.2	Self-Ordering Behavior	67
3.3.3	Boundary Energy	68
3.3.4	Boundary Mobility	72
3.3.5	Pinning Systems	75
3.3.6	Stored Energy	77
3.3.7	Summary	80
3.4	Speed-Up Algorithms	80
3.4.1	The Boundary-Site Algorithm	81
3.4.2	The N -Fold Way Algorithm	82
3.4.3	Parallel Algorithm	84
3.4.4	Summary	87
3.5	Applications of the Potts Model	87
3.5.1	Grain Growth	87
3.5.2	Incorporating Realistic Textures and Misorientation Distributions	89
3.5.3	Incorporating Realistic Energies and Mobilities	92
3.5.4	Validating the Energy and Mobility Implementations	93
3.5.5	Anisotropic Grain Growth	95
3.5.6	Abnormal Grain Growth	98
3.5.7	Recrystallization	102
3.5.8	Zener Pinning	103
3.6	Summary	107
3.7	Final Remarks	107
3.8	Acknowledgments	108
4	Cellular Automata	109
4.1	A Definition	109
4.2	A One-Dimensional Introduction	109
4.2.1	One-Dimensional Recrystallization	111
4.2.2	Before Moving to Higher Dimensions	111
4.3	+2D CA Modeling of Recrystallization	116
4.3.1	CA-Neighborhood Definitions in Two Dimensions	116
4.3.2	The Interface Discretization Problem	118
4.4	+2D CA Modeling of Grain Growth	123

4.4.1	Approximating Curvature in a Cellular Automaton Grid	124
4.5	A Mathematical Formulation of Cellular Automata	128
4.6	Irregular and Shapeless Cellular Automata	129
4.6.1	Irregular Shapeless Cellular Automata for Grain Growth	131
4.6.2	In the Presence of Additional Driving Forces	135
4.7	Hybrid Cellular Automata Modeling	136
4.7.1	Principle	136
4.7.2	Case Example	137
4.8	Lattice Gas Cellular Automata	140
4.8.1	Principle—Boolean LGCA	140
4.8.2	Boolean LGCA—Example of Application	142
4.9	Network Cellular Automata—A Development for the Future?	144
4.9.1	Combined Network Cellular Automata	144
4.9.2	CNCA for Microstructure Evolution Modeling	145
4.10	Further Reading	147
5	Modeling Solid-State Diffusion	151
5.1	Diffusion Mechanisms in Crystalline Solids	151
5.2	Microscopic Diffusion	154
5.2.1	The Principle of Time Reversal	154
5.2.2	A Random Walk Treatment	155
5.2.3	Einstein's Equation	157
5.3	Macroscopic Diffusion	160
5.3.1	Phenomenological Laws of Diffusion	160
5.3.2	Solutions to Fick's Second Law	162
5.3.3	Diffusion Forces and Atomic Mobility	164
5.3.4	Interdiffusion and the Kirkendall Effect	168
5.3.5	Multicomponent Diffusion	171
5.4	Numerical Solution of the Diffusion Equation	174
6	Modeling Precipitation as a Sharp-Interface Phase Transformation	179
6.1	Statistical Theory of Phase Transformation	181
6.1.1	The Extended Volume Approach—KJMA Kinetics	181
6.2	Solid-State Nucleation	185
6.2.1	Introduction	185
6.2.2	Macroscopic Treatment of Nucleation—Classical Nucleation Theory	186
6.2.3	Transient Nucleation	189
6.2.4	Multicomponent Nucleation	191
6.2.5	Treatment of Interfacial Energies	194
6.3	Diffusion-Controlled Precipitate Growth	197
6.3.1	Problem Definition	199
6.3.2	Zener's Approach for Planar Interfaces	202
6.3.3	Quasi-static Approach for Spherical Precipitates	203
6.3.4	Moving Boundary Solution for Spherical Symmetry	205
6.4	Multiparticle Precipitation Kinetics	206
6.4.1	The Numerical Kampmann–Wagner Model	206
6.4.2	The SFFK Model—A Mean-Field Approach for Complex Systems	209
6.5	Comparing the Growth Kinetics of Different Models	215

7	Phase-Field Modeling	219
7.1	A Short Overview	220
7.2	Phase-Field Model for Pure Substances	222
7.2.1	Anisotropy Formulation	224
7.2.2	Material and Model Parameters	226
7.2.3	Application to Dendritic Growth	226
7.3	Case Study	228
7.3.1	Phase-Field Equation	229
7.3.2	Finite Difference Discretization	229
7.3.3	Boundary Values	231
7.3.4	Stability Condition	232
7.3.5	Structure of the Code	232
7.3.6	Main Computation	233
7.3.7	Parameter File	236
7.3.8	MatLab Visualization	237
7.3.9	Examples	238
7.4	Model for Multiple Components and Phases	241
7.4.1	Model Formulation	241
7.4.2	Entropy Density Contributions	242
7.4.3	Evolution Equations	245
7.4.4	Nondimensionalization	248
7.4.5	Finite Difference Discretization and Staggered Grid	249
7.4.6	Optimization of the Computational Algorithm	252
7.4.7	Parallelization	253
7.4.8	Adaptive Finite Element Method	253
7.4.9	Simulations of Phase Transitions and Microstructure Evolution	253
7.5	Acknowledgments	265
8	Introduction to Discrete Dislocations Statics and Dynamics	267
8.1	Basics of Discrete Plasticity Models	267
8.2	Linear Elasticity Theory for Plasticity	268
8.2.1	Introduction	268
8.2.2	Fundamentals of Elasticity Theory	269
8.2.3	Equilibrium Equations	273
8.2.4	Compatibility Equations	274
8.2.5	Hooke's Law—The Linear Relationship between Stress and Strain	275
8.2.6	Elastic Energy	280
8.2.7	Green's Tensor Function in Elasticity Theory	280
8.2.8	The Airy Stress Function in Elasticity Theory	283
8.3	Dislocation Statics	284
8.3.1	Introduction	284
8.3.2	Two-Dimensional Field Equations for Infinite Dislocations in an Isotropic Linear Elastic Medium	285
8.3.3	Two-Dimensional Field Equations for Infinite Dislocations in an Anisotropic Linear Elastic Medium	287
8.3.4	Three-Dimensional Field Equations for Dislocation Segments in an Isotropic Linear Elastic Medium	289

8.3.5	Three-Dimensional Field Equations for Dislocation Segments in an Anisotropic Linear Elastic Medium	292
8.4	Dislocation Dynamics	298
8.4.1	Introduction	298
8.4.2	Newtonian Dislocation Dynamics	299
8.4.3	Viscous and Viscoplastic Dislocation Dynamics	307
8.5	Kinematics of Discrete Dislocation Dynamics	310
8.6	Dislocation Reactions and Annihilation	311
9	Finite Elements for Microstructure Evolution	317
9.1	Fundamentals of Differential Equations	317
9.1.1	Introduction to Differential Equations	317
9.1.2	Solution of Partial Differential Equations	320
9.2	Introduction to the Finite Element Method	321
9.3	Finite Element Methods at the Meso- and Macroscale	322
9.3.1	Introduction and Fundamentals	322
9.3.2	The Equilibrium Equation in FE Simulations	324
9.3.3	Finite Elements and Shape Functions	324
9.3.4	Assemblage of the Stiffness Matrix	327
9.3.5	Solid-State Kinematics for Mechanical Problems	329
9.3.6	Conjugate Stress–Strain Measures	331
	Index	335

This page intentionally left blank

Preface

Being actively involved since 1991 in different research projects that belong under the field of computational materials science, I always wondered why there is no book on the market which introduces the topic to the beginning student. In 2005 I was approached by Joel Stein to write a book on this topic, and I took the opportunity to attempt to do so myself. It was immediately clear to me that such a task transcends a mere copy and paste operation, as writing for experts is not the same as writing for the novice. Therefore I decided to invite a respectable collection of renowned researchers to join me on the endeavor. Given the specific nature of my own research, I chose to focus the topic on different aspects of computational *microstructure evolution*. This book is the result of five extremely busy and active researchers taking a substantial amount of their (free) time to put their expertise down in an understandable, self-explaining manner. I am very grateful for their efforts, and hope the reader will profit. Even if my original goals were not completely met (atomistic methods are missing and there was not enough time to do things as perfectly as I desired), I am satisfied with—and a bit proud of—the result.

Most chapters in this book can be pretty much considered as stand-alones. Chapters 1 and 2 are included for those who are at the very beginning of an education in materials science; the others can be read in any order you like.

Finally, I consider this book a work in progress. Any questions, comments, corrections, and ideas for future and extended editions are welcome at comp.micr.evol@mac.com. You may also want to visit the web site accompanying this book at <http://books.elsevier.com/companions/9780123694683>.

Koen Janssens,
Linn, Switzerland,
December 2006

This page intentionally left blank

1 Introduction

—Koen Janssens

The book in front of you is an introduction to *computational* microstructure evolution. It is not an introduction to microstructure evolution. It does assume you have been sufficiently introduced to materials science and engineering to know what microstructure evolution is about.

That being said we could end our introduction here and skip straight to the computational chapters. However, if you intend to learn more about the science of polycrystalline materials but want to learn about them through their computational modeling, then this chapter will give you the bare-bones introduction to the secrets of microstructures. Just remember that the one law binding any type of computational modeling is equally valid for computational microstructure evolution:

$$\text{garbage-in} = \text{garbage-out} \quad (1.1)$$

You have been warned—we wash our hands in innocence.

1.1 Microstructures Defined

The world of materials around us is amazingly diverse. It is so diverse that scientists feel the need to classify materials into different types. Like the authors of this book, your typical technologist classifies materials based on their technical properties. Fundamental groups are, for example, metals and alloys, ceramics, polymers, and biomaterials. More specific classifications can be, for example, semiconductors, nanomaterials, memory alloys, building materials, or geologic minerals.

In this book the focus is on materials with *polycrystalline* microstructures. For the sake of simplicity let us consider a pure metal and start with atoms as the basic building blocks. The atoms in a metal organize themselves on a crystal lattice structure. There are different types of lattice structures each with their defining symmetry. Auguste Bravais was the first to classify them correctly in 1845, a classification published in 1850–1851 [Bra, Bra50, Bra51]. The lattice parameters define the length in space over which the lattice repeats itself, or in other words the volume of the unit cell of the crystal. The organization of atoms on a lattice with a specific set of lattice parameters is what we call a solid state *phase*. Going from pure metals to alloys, ceramics, polymers, and biomaterials, the structures get more and more complex and now consist of lattices of groups of different atoms organized on one or more different lattice structures.

For completeness it should be mentioned that not in all materials are the atoms organized on lattices; usually the larger the atom groups the less it becomes probable—like for most polymers—and we end up with amorph or glassy structures in the material.

The most simple microstructure is a perfect single crystal. In a perfect single crystal the atoms are organized on a crystal lattice without a single defect. What crystal structure the atoms organize on follows from their electronic structure. *Ab initio* atomistic modeling is a branch of computational materials science concerning itself with the computation of equilibrium crystal structures. Unfortunately we do not treat any form of molecular dynamics in this book—or, to be more honest would be to admit we did not make the deadline. But keep your wits up, we are considering it for a future version of this book, and in the meantime refer to other publications on this subject (see Section 1.4 at the end of this chapter).

When the material is not pure but has a composition of elements¹, the lattice is also modified by solute or interstitial atoms that are foreign to its matrix. Depending on temperature and composition, a material may also have different phases, meaning that the same atoms can be arranged in different crystal lattice structures, the equilibrium phase being that one which has the lowest Gibbs free energy at a particular temperature and pressure. If you lost me when I used the words “Gibbs free energy” you may want to read Chapter 2 in this book (and needless to say, if you are an expert on the matter you may wish to skip that chapter, unless of course you feel like sending us some ideas and corrections on how to make this book a better one). In any case, but especially in view of equation (1.1), it is important that you have a minimum of understanding of materials thermodynamics.

Any deviation from a material’s perfect single-crystal structure increases the energy stored in the material by the introduction of *crystal defects*. These defects are typically classified according to their dimensions in space: point defects, line defects, and surface defects. Important in the context of microstructures is that the energy stored in these defects is a driving force for microstructure transformation. For example, a grain boundary is a defect structure, and the microstructure is thereby driven to minimize its free energy by minimizing the surface area of grain boundaries in itself, hence the process of grain growth. In a deformed microstructure of a metal, the dislocations can be removed from the microstructure by recovery, in which dislocations mutually annihilate, but also by the nucleation and growth of new, relatively dislocation-free grains, hence the process of recrystallization. Phase transformations are similar in that they also involve nucleation and growth of grains, but are different in the origin of the driving force, which is now the difference of the free energy density of different phases of the same crystal. Once again, you can read more about all the underlying thermodynamics in Chapter 2. Another point to keep in mind is that other properties, such as grain boundary mobility, may be coupled to the presence of defects in a crystal.

At this point we have defined most of the relevant concepts and come back to what the microstructure of a polycrystalline material is:

A microstructure is a spatially connected collection of arbitrarily shaped *grains* separated by *grain boundaries*, where each grain is a (possibly-non-defect-free) single crystal and the grain boundaries are the location of the interfaces between grains.

1.2 Microstructure Evolution

Now that you have some idea of what microstructures are, we can start talking about microstructure *evolution*. Evolution is actually an unfortunate word choice, as microstructures

¹ A real material is *always* composed of a multitude of elements, leading to the saying that “materials science is the physics of dirt” [Cah02]—but that is another story

do not evolve in the way living materials do. The use of the word evolution probably originated from an attempt to find generalized wording for the different transformation processes that are observed in changing microstructures. The word “transformation” is somewhat reserved to “phase transformation” when speaking about microstructures. A better term possibly would have been *microstructure transmutation*, as it points more precisely at what is really meant: under the influence of external heat and/or work, a microstructure transmutes into another microstructure.

But let us keep it simple and explain what microstructure evolution is by illustration with an example from the metal processing industry shown in Figure 1-1. The figure shows how a metal is continuously cast and subsequently hot rolled. Many different microstructure transformations come into action in this complex materials processing line:

Solidification: Solidification is the process which defines the casting process at the microstructure scale. Solidification is a phase transformation, special because a liquid phase transforms into a solid phase. In this book you can find examples of the simulation of solidification using the phase-field method in Chapter 7.

Diffusion: Diffusion also is one of the main characters, for example, in the segregation of elements in front of the solidification front. But diffusion is a process which plays a role in any microstructure transformation at an elevated temperature, be it the annealing after casting or before and during hot rolling (or any other technological process

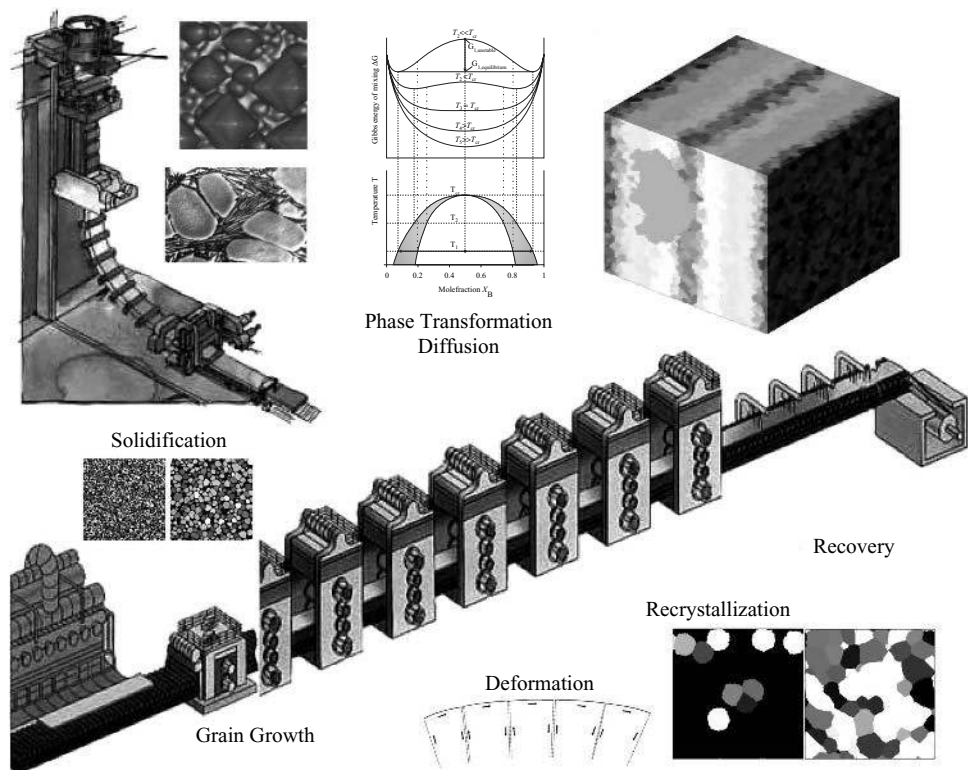


FIGURE 1-1 Schematic of the industrial processing of a metal from its liquid phase to a sheet metal, showing the different microstructure transformations which occur.

involving heat treatments). You can find information on the modeling of the diffusion process in Chapter 5.

Phase Transformation: Phase transformation is microstructural transformation in the solid state that occurs at elevated temperature heat treatments when a material has different thermodynamically stable phases at different temperatures, like iron has a face centered cubic phase (austenite) and a body centered cubic phase (ferrite). Phase transformations can be modeled computationally using a variety of methods, several of which are introduced in this book. The phase-field model is treated in Chapter 7, and although not explicitly treated, Potts-type Monte Carlo in Chapter 3 and cellular automata in Chapter 4 are a possibility. Read more about the underlying thermodynamics in Chapter 6.

Deformation: The plastic deformation of a metal is a topic that has been studied since the beginning of materials science. Plasticity can be modeled at the continuum scale, and recently the field of multiscale modeling is slowly but certainly closing the gap between the microstructure and the continuum scales of computational modeling. In this book we only touch on plasticity with the description of two computational approaches. Closer to the atomistic scale is discrete dislocation dynamics modeling, which is introduced in Chapter 8. Coming from the scale of continuum modeling, we also treat the application of finite elements to microstructure evolution modeling in Chapter 9. The *recovery* of a plastically deformed metal is in essence a process at the dislocation scale, but it is not addressed in this book.

Recrystallization and Grain Growth: Recrystallization and grain growth, on the other hand, are treated in detail as an application of cellular automata in Chapter 4 and of Potts-type Monte Carlo in Chapter 3.

1.3 Why Simulate Microstructure Evolution?

Modern materials are characterized by a wide spectrum of tailored mechanical, optical, magnetic, electronic, or thermophysical properties. Frequently these properties can be attributed to a specially designed microstructure.

A dedicated microstructure of a metal promoting its strength and toughness could be one with small and homogeneous grains, minimum impurity segregation, and a high number density of small, nanometer-sized precipitates to stabilize grain boundaries and dislocations. To obtain this particular microstructure in the course of the material manufacturing processes, advantage is oft taken of different microstructural transformation processes that have the power of producing the desired microstructures in a reproducible way, such as depicted in Figure 1-1: phase transformation, diffusion, deformation, recrystallization, and grain growth.

Today, computational modeling is one of the tools at the disposal of scientific engineers, helping them to better understand the influence of different process parameters on the details of the microstructure. In some cases such computational modeling can be useful in the optimization of the microstructure to obtain very specific material properties, and in specific cases modeling may be used directly to design new microstructures. In spite of the fact that the latter is frequently used as an argument in favor of computational materials science, the true strength of the computational approach is still its use as a tool for better understanding. Technologically relevant microstructures are four-dimensional (in space and time) creatures that are difficult for the human mind to grasp correctly. That this understanding is relevant becomes obvious when one reads the way Martin, Doherty, and Cantor view microstructures [MDC97]: a microstructure is a meta-stable structure that is kinetically prevented to evolve into a minimum free-energy

configuration. This means that to produce a specific microstructure, one must understand the kinetic path along which it evolves, and be able to stop its evolution at the right moment in processing time.

With the help of computational modeling, the scientist is able to dissect the microstructure in space and in its evolution in time, and can, for example, perform different parameter studies to decide how to ameliorate the manufacturing process. Building such a computational tool always needs three components:

1. Having correct models defining the underlying physics of the different subprocesses that act in the play of microstructure evolution, for example, diffusion equations or laws for grain boundary migration or dislocation motion.
2. A computational model, which is capable of simulating the evolution of the microstructure using the underlying laws of physics as input. A major concern in such a model is always that the different sub-models compute on the same scale. As an example, it is relatively straightforward to model recovery of a deformed metal analytically, fitting the parameters to experimental data. It is already much more complex to model recrystallization using cellular automata, as it is not so straightforward to calibrate the length of an incremental step in the model against real time. The latter is usually circumvented by normalizing the simulation and experimental data based on particular points in time (e.g., a certain amount of the volume recrystallized), but such assumes that the relation between time and simulation step is linear, which is not always true. Combining both the analytical recovery model and the computational recrystallization model requires a true time calibration so this trick can no longer be applied, resulting in tedious time calibration experiments and simulations that need to be performed with great care if one aims to transcend mere qualitative predictions.
3. Finally, unless one is studying the microstructure itself, one needs additional modeling, which relates the (simulated) microstructures on the one side, to the target properties on the other side of the equation. Such a model would, for example, compute the plastic yield locus of a metal based on characteristics of the microstructure such as the grain size distribution. It should need little imagination to realize that such a computation can easily be equally complex as the microstructure evolution model itself.

This book focuses entirely on item 2 in the preceding list. For the less sexy topics the reader is referred to other monographs—see further reading.

1.4 Further Reading

1.4.1 On Microstructures and Their Evolution from a Noncomputational Point of View

The book by Humphreys and Hatherly [HH96] is certainly one of the most referenced books on this topic and gives a good overview. Other important monographs I would consider are the work of Gottstein and Shvindlerman [GS99] on the physics of grain boundary migration in metals, the book of Martin, Doherty, and Cantor [MDC97] on the stability of microstructures, and Sutton and Balluffi [SB95] on interfaces in crystalline materials. Finally, if you need decent mathematics to compute crystal orientations and misorientations, Morawiec's work [Mor04] may help.

1.4.2 On What Is Not Treated in This Book

Unfortunately we did not have time nor space to treat all methods you can use for microstructure evolution modeling. If you did not find your taste in our book, here are some other books we prudently suggest.

Molecular Dynamics: Plenty of references here. Why not start with a classic like Frenkel and Smit [FS96]?

Level Set: Level set methods and fast marching methods are in the book of Sethian [Set99].

Continuum Plasticity of Metals: Yes, continuum theory also finds its application in the simulation of microstructures, especially when it concerns their deformation. Actually, you can find its application to microstructure evolution in this book in Chapter 9. Other monographs on the subject in general are plenty. Lemaitre and Chaboche [LC90] certainly gives a very good overview, but it may be on the heavy side for the beginning modeler. An easier point of entry may be Han and Reddy [HR99] on the mathematics of plasticity, and Dunne and Petrinic [DP05] on its computational modeling.

Particle Methods: See Liu and Liu [LL03].

Genetic Algorithms: Because you never know when you may need these, the book by Haupt and Haupt [HH04] describes the basic ideas really well.

The Meshless Local Petrov-Galerkin (MLPG) Method, S. N. Atluri and S. Shen, Tech Science Press, Forsyth, GA, 2002. [Atl02]

See Torquato [Tor02] on methods for computational modeling of the relation between microstructure and materials properties!

Bibliography

- [Bra] M. A. Bravais. http://en.wikipedia.org/wiki/Bravais_lattice.
- [Bra50] M. A. Bravais. *J. Ecole Polytechnique*, 19:1–128, 1850.
- [Bra51] M. A. Bravais. *J. Ecole Polytechnique*, 20:101–278, 1851.
- [Cah02] R. W. Cahn. The science of dirt. *Nature Materials*, 1:3–4, 2002.
- [DP05] F. Dunne and N. Petrinic. *Introduction to Computational Plasticity*. Oxford University Press, Oxford, 2005.
- [FS96] D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Academic Press, San Diego, 1996.
- [GS99] G. Gottstein and L. S. Shvindlerman. *Grain Boundary Migration in Metals*. CRC Press LLC, Boca Raton, FL, 1999.
- [HH96] F. J. Humphreys and M. Hatherly. *Recrystallization and Related Annealing Phenomena*. Pergamon, Oxford, 1996.
- [HH04] R. L. Haupt and S. E. Haupt. *Practical Genetic Algorithms*. Wiley, Hoboken, NJ, 2004.
- [HR99] W. Han and B. D. Reddy. *Plasticity—Mathematical Theory and Numerical Analysis*. Springer-Verlag, Berlin, 1999.
- [LC90] J. Lemaitre and J.-L. Chaboche. *Mechanics of Solid Materials*. Cambridge University Press, Cambridge, 1990.
- [LL03] G. R. Liu and M. B. Liu. *Smoothed Particle Hydrodynamics*. World Scientific, London, 2003.
- [MDC97] J. W. Martin, R. D. Doherty, and B. Cantor. *Stability of Microstructures in Metallic Systems*. Cambridge University Press, Cambridge, 1997.
- [Mor04] A. Morawiec. *Orientations and Rotations: Computations in Crystallographic Textures*. Springer-Verlag, Berlin, 2004.
- [SB95] A. P. Sutton and R. W. Balluffi. *Interfaces in Crystalline Materials*. Clarendon Press, Oxford, 1995.
- [Set99] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, 1999.
- [Tor02] S. Torquato. *Random Heterogeneous Materials: Microstructure and Macroscopic Properties*. Springer-Verlag, Berlin, 2002.
- [Atl02] S. N. Atluri and S. Shen. *The Meshless Local Petrov-Galerkin (MLPG) Method*. Tech Science Press, Forsyth, GA, 2002.
-

2 Thermodynamic Basis of Phase Transformations

—Ernst Kozeschnik

Many of the models that are discussed in this book rely on the knowledge of thermodynamic quantities, such as solution enthalpies, chemical potentials, driving forces, equilibrium mole fractions of components, etc. These quantities are needed as model input parameters and they are often not readily available in experimental form when dealing with special or complex systems. However, in the last decades, suitable theoretical models have been developed to assess and collect thermodynamic and kinetic data and store them in the form of standardized databases. Thus, essential input data for modeling and simulation of microstructure evolution is accessible on the computer.

Although thermodynamics is covered in numerous excellent textbooks and scientific publications, we nevertheless feel the strong necessity to introduce the reader to the basic concepts of thermodynamics, and in particular to *solution thermodynamics* (Section 2.2), which we will be most concerned with in computational modeling of microstructure evolution. The basics are discussed at least to a depth that the theoretical concepts of the modeling approaches can be understood and correctly applied and interpreted as needed in the context of this book. Some of the material that is presented subsequently is aimed at giving the reader sufficient understanding of the underlying approaches to apply theory in the appropriate way. Some of it is aimed at providing reference material for later use.

Thermodynamics provides a very powerful methodology for describing macroscopic observables of materials on a quantitative basis. In the last decades, mathematical and computational methods have been developed to allow extrapolation of known thermodynamic properties of binary and ternary alloys into frequently unexplored higher-order systems of technical relevance. The so-called method of *computational thermodynamics* (CT) is an indispensable tool nowadays in development of new materials, and it has found its way into industrial practice where CT assists engineers in optimizing heat treatment procedures and alloy compositions. Due to the increasing industrial interest, comprehensive thermodynamic databases are being developed in the framework of the CALPHAD (CALculation of PHase Diagrams) technique, which in combination with commercial software for Gibbs energy minimization can be used to predict phase stabilities in almost all alloy systems of technical relevance [KEH⁺00]. More and more students become acquainted with commercial thermodynamic software packages such

as ThermoCalc [SJA85], MTDData [DDC⁺89], F*A*C*T [PTBE89], ChemSage [EH90], or PANDAT [CZD⁺03] already at universities, where CT is increasingly taught as an obligatory part of the curriculum.

Traditionally, *computational thermodynamics* is connected to the construction of phase diagrams on the scientist's and engineer's desktop. There, it can provide information about which stable phases one will find in a material in thermodynamic equilibrium at a given temperature, pressure, and overall chemical composition. This knowledge is already of considerable value to the engineer when trying to identify, for instance, solution temperatures of wanted and unwanted phases to optimize industrial heat treatments. Moreover, and this is of immediate relevance for the present textbook: although the thermodynamic parameters that are stored in the thermodynamic databases have been assessed to describe equilibrium conditions, these data also provide information on thermodynamic quantities in the nonequilibrium state. For instance, chemical potentials of each element in each phase can be evaluated for given temperature, pressure, and phase composition. From these data, the chemical driving forces can be derived and finally used in models describing kinetic processes such as phase transformations or precipitate nucleation and growth.

It is not the intent of the present book to recapitulate solution thermodynamics in scientific depth, and we will restrict ourselves to an outline of the basic concepts and methods in order to provide the reader with the necessary skills to apply these theories in appropriate ways. For a more comprehensive treatment, the reader is referred to some of the many excellent textbooks on solution thermodynamics (e.g., refs. [Hil98, SM98, Cal85, Hac96, MA96, Wag52, FR76]).

2.1 Reversible and Irreversible Thermodynamics

2.1.1 The First Law of Thermodynamics

Thermodynamics is a macroscopic art dealing with energy and the way how different forms of energy can be transformed into each other. One of the most fundamental statements of thermodynamics is related to the *conservation of energy* in a closed system, that is, a system with a constant amount of matter and no interactions of any kind with the surrounding. When introducing the internal energy U as the sum of all kinetic, potential, and interaction energies in the system, we can define U formally as a part Q coming from the heat that has flown into the system and a part W coming from the work done on the system:

$$U = Q + W \quad (2.1)$$

It is important to recognize that this definition does not provide information about the absolute value of U and, in this form, we are always concerned with the problem of defining an appropriate reference state. Therefore, instead of using the absolute value of the internal energy, it is often more convenient to consider the change of U during the transition from one state to another and to use equation (2.1) in its differential form as

$$dU = dQ + dW \quad (2.2)$$

By definition, the internal energy U of a closed system is constant. Therefore, equation (2.2) tells us that, in systems with constant amount of matter and in the absence of interactions with the surrounding, energy can neither be created nor destroyed, although it can be converted from one form into another. This is called the *first law of thermodynamics*.

The internal energy U is a *state function* because it is uniquely determined for each combination of the *state variables* temperature T , pressure P , volume V , and chemical composition \mathbf{N} . The vector \mathbf{N} contains the numbers N_i of moles of components i . Any thermodynamic property that is independent of the size of the system is called an *intensive property*. Examples for intensive quantities are T and P or the chemical potential μ . An intensive state variable or function is also denoted as a *thermodynamic potential*. A property that depends on the size of the system is called an *extensive property*. Typical examples are the state variable V or the state function U .

The value of a state function is always independent of the way how a certain state has been reached, and for the internal energy of a closed system we can write

$$\oint dU = 0 \quad (2.3)$$

A necessary prerequisite for the validity of equation (2.3) is that the variation of the state variables is performed in infinitesimally small steps, and the process thus moves through a continuous series of equilibria. In other words, after variation of any of the state variables, we are allowed to measure any thermodynamic quantity only after the system has come to a complete rest.

It is also important to realize that the state variables introduced before are not independent of each other: If we have c independent components in the system, only $c + 2$ state variables can be chosen independently. For instance, in an ideal one-component gas ($c = 1$), we have the four state variables P , T , V , and N . Any three of these variables can be chosen independently, while the fourth parameter is determined by the ideal gas law $PV = NRT$. R is the *universal gas constant* ($R = 8.3145 \text{ J}(\text{mol K})^{-1}$). The choice of appropriate state variables is dependent on the problem one is confronted with. In solution thermodynamics, a natural choice for the set of state variables is T , P , and \mathbf{N} .

The quantities Q and W are *not* state functions because the differentials dQ and dW simply describe the interaction of the system with its surrounding or the interaction between two sub-systems that are brought into contact. Depending on the possibilities of how a system can exchange thermal and mechanical energy with its surrounding, different expressions for dQ and dW will be substituted into equation (2.2). For instance, a common and most important path for mechanical interaction of two systems is the work done against hydrostatical pressure. For convenience, a new function H is introduced first with

$$H = U + PV \quad (2.4)$$

which is called *enthalpy*. H is also a state function and in its differential form we have

$$dH = dU + PdV + VdP \quad (2.5)$$

Now consider an insulated cylinder filled with ideal gas and a frictionless piston on one side. If the temperature of the gas is increased by an infinitesimal amount dT and the pressure of the gas is held constant, the piston must move outwards because the volume of the gas has increased by the infinitesimal amount dV . In the course of this process, work dW is done against the hydrostatic pressure P and we have

$$dW = -PdV \quad (2.6)$$

The minus sign comes from the fact that dW is defined as the mechanical energy received by the system. Substituting equations (2.2) and (2.6) into the general definition (2.5) leads to

$$dH = dQ + VdP \quad (2.7)$$

Under constant pressure and constant chemical composition ($dP = 0$, $dN_i = 0$), equation (2.7) reduces to

$$(dH)_{P,\mathbf{N}} = dQ \quad (2.8)$$

thus manifesting that the addition of any amount of heat dQ to the system under these conditions is equal to the increase dH . If we further assume a proportionality between dH and dT , we can write

$$(dH)_{P,\mathbf{N}} = C_P \cdot dT \quad (2.9)$$

The proportionality constant C_P is called *specific heat capacity*, and it is commonly interpreted as the amount of heat that is necessary to increase the temperature of one mole of atoms by one degree. Formally, the definition of the specific heat capacity is written

$$C_P = \left(\frac{\partial H}{\partial T} \right)_{P,\mathbf{N}} \quad (2.10)$$

The enthalpy H has been introduced for conditions of constant pressure, constant chemical composition, and under the assumption that dW in equation (2.2) is representing a work done against a hydrostatic pressure. An analogy will now be sought for the incremental heat dQ .

In the previous example we have expressed the mechanical work input as $-\Delta W = \Delta(PV)$ and used the differential form with

$$-dW = PdV + VdP \quad (2.11)$$

In analogy to the mechanical part, we assume that the stored heat in the system can be expressed by a product $\Delta Q = \Delta(TS)$. For the differential form we can write

$$dQ = TdS + SdT \quad (2.12)$$

If heat is added under conditions of constant temperature, we finally arrive at the so-called *thermodynamic definition of entropy*:

$$dS = \frac{dQ}{T} \quad (2.13)$$

The concept of *entropy* was introduced by the German physicist Rudolf Clausius (1822–1888). The word *entropy* has Greek origin and means transformation. Similar to U and H , entropy S is a state function. Its value is only dependent on the state variables T , P , V , \mathbf{N} and it is independent of the way how the state was established. We can therefore also write

$$\oint dS = \oint \frac{dQ}{T} = 0 \quad (2.14)$$

2.1.2 The Gibbs Energy

In solution thermodynamics, it is convenient to use the state variables T , P , and \mathbf{N} (or \mathbf{X} , which will be introduced in the next section) to describe the state of a system. This selection of variables is mainly driven by practical considerations: In an experimental setup, which is related to microstructure transformation problems, P , T , and \mathbf{N} are most easily controlled. Based on this selection of variables, the entire thermodynamic properties of a system can be described by the so-called *Gibbs energy* G , which is given as

$$G(T, P, \mathbf{N}) = H(T, P, \mathbf{N}) - T \cdot S(T, P, \mathbf{N}) \quad (2.15)$$

G is a state function and we can therefore write

$$\oint dG = 0 \quad (2.16)$$

The Gibbs energy G provides a unique description of the state of a system, and many properties of a thermodynamic system can be obtained from its partial derivatives. The following relations hold:

$$\left(\frac{\partial G(T, P, \mathbf{N})}{\partial P} \right)_{T, \mathbf{N}} = V \quad (2.17)$$

$$\left(\frac{\partial G(T, P, \mathbf{N})}{\partial T} \right)_{P, \mathbf{N}} = -S \quad (2.18)$$

$$\left(\frac{\partial G(T, P, \mathbf{N})}{\partial N_i} \right)_{T, P, N_{i \neq j}} = \mu_i \quad (2.19)$$

Each of these derivatives is evaluated with the subscript quantities held constant. μ_i is called the *chemical potential* of element i . Finally, under constant temperature and pressure, application of the total derivative of G with respect to the composition variables N_i and equation (2.19) delivers the important relation

$$G = \sum N_i \cdot \mu_i \quad (2.20)$$

The Gibbs energy and chemical potentials play an important role in modeling of kinetic processes. This will be discussed in more detail in subsequent chapters.

2.1.3 Molar Quantities and the Chemical Potential

In thermodynamic and kinetic modeling, for convenience and for practical reasons, the size of the system is frequently limited to a constant amount of matter. In thermodynamics, this measure is commonly one mole of atoms, whereas in kinetics, usually unit amount of volume is regarded. Accordingly, the Gibbs energy G of one mole of atoms can be expressed in terms of T , P , and a new variable \mathbf{X} , which represents the vector of mole fractions X_i of elements i , as

$$G_m(T, P, \mathbf{X}) = H_m(T, P, \mathbf{X}) - T \cdot S_m(T, P, \mathbf{X}) \quad (2.21)$$

The subscript m indicates the use of molar quantities. G_m is denoted as the *molar Gibbs energy*. For the sum of all mole fractions, the following constraint applies:

$$\sum X_i = 1 \quad (2.22)$$

The *chemical potential* μ_i has already been formally introduced in the previous section as the partial derivative of the Gibbs energy with respect to the number of moles N_i . From a practical point of view, the chemical potential represents a measure for the change of Gibbs energy when infinitesimal amount of element i is added to the system.

When investigating chemical potentials in the framework of the new set of variables T , P , and \mathbf{X} , we have to be aware of the fact that the composition variables X_i are not independent from each other and the derivative of G in molar quantities has to be evaluated under the constraint (2.22). Substituting G by $N \cdot G_m$ leads to

$$\mu_i = \frac{\partial G}{\partial N_i} = \frac{\partial}{\partial N_i} (N \cdot G_m) = 1 \cdot G_m + N \cdot \frac{\partial G_m}{\partial N_i} \quad (2.23)$$

Since we have

$$\frac{\partial}{\partial N_i} = \sum_j \frac{\partial}{\partial X_j} \frac{\partial X_j}{\partial N_i} \quad (2.24)$$

and

$$X_i = 1 - \sum_{j \neq i} X_j, \quad \frac{\partial X_j}{\partial N_j} = \frac{N - N_j}{N^2}, \quad \frac{\partial X_j}{\partial N_k} = -\frac{N_j}{N^2} \quad (2.25)$$

we finally obtain

$$\mu_i = G_m + \frac{\partial G_m}{\partial X_i} - \sum_j X_j \frac{\partial G_m}{\partial X_j} \quad (2.26)$$

Equation (2.26) is of considerable value in practical CT and Computational Microstructure Evolution (CME) because it relates the chemical potential μ_i of an element i to the molar Gibbs energy G_m . Both quantities are thermodynamic potentials and both quantities can be used for a definition of equilibrium. It will be shown later that chemical potentials can, moreover, conveniently be used to define the driving force for internal reactions, such as phase transformations or diffusion. Based on equation (2.20), the relation between molar Gibbs energy and the chemical potentials can be written as

$$G_m = \sum X_i \cdot \mu_i \quad (2.27)$$

It must be emphasized, finally, that equation (2.26) should be used with some care, since this expression is obtained by variation of one mole fraction component while holding all others constant. In terms of mole fractions this is of course not possible and the physical meaning of the chemical potential derived in this way is questionable (see, e.g., Hillert [Hil98]). However, it will be demonstrated later (see, e.g., diffusion forces, Section 5.3.3) that, in most cases, the chemical potential is applied in a form where one component of a mixture is exchanged against some reference component(s). The difference ($\mu_i - \mu_{\text{ref}}$) does not inherit this conceptual difficulty and can be used without this caution.

2.1.4 Entropy Production and the Second Law of Thermodynamics

So far, we have considered thermodynamic processes as infinitesimal variations of state variables that lead through a continuous series of equilibria. We have manifested the properties of some thermodynamic state functions under equilibrium conditions. We have found that mechanical work and heat can be converted into each other without loss of energy as long as the variation of state variables occurs infinitely slowly and the system can come to a rest at all stages of the process. Under these conditions, we have found that the nature of thermodynamic processes

is *reversible*. In the following sections, the grounds of so-called *equilibrium thermodynamics* are left behind and processes are analyzed, where the variation of state variables is performed outside the convenient—but impracticable—assumption of continuous equilibrium. The branch of science dealing with these phenomena is called *irreversible thermodynamics*.

In his fundamental work on the thermodynamic properties of entropy, Rudolph Clausius (see Section 2.1.1) was strongly influenced by the ideas of the French physicist Nicolas Carnot (1796–1832). The latter investigated the efficiency of steam machines and introduced the famous thought experiment of an idealized machine that converts thermal into mechanical energy and vice versa. When going through the so-called *Carnot cycle*, a system can deliver mechanical work as a result of heat transport from a warmer to a cooler heat reservoir. Consider the following closed thermodynamic process (see Figure 2-1), which operates between two heat reservoirs at T_a and T_b , with $T_a < T_b$:

1. Let the system be in contact with the cooler reservoir at a temperature T_a . Perform an *isothermal compression* from V_1 to V_2 . During compression, the work W_1 is done on the system and, simultaneously, the system gives away the heat $-Q_1$.
2. Decouple the system from the reservoir and perform an *adiabatic compression* from V_2 to V_3 . Since no heat is exchanged with the surroundings, $Q_2 = 0$. Continue with compression until the temperature of the system has increased to T_b . Let the work done on the system be W_2 .
3. Put the system into contact with the warmer reservoir at T_b . Perform an *isothermal expansion* from V_3 to V_4 . During expansion, the work $-W_3$ is done by the system and, simultaneously, the system picks up the heat Q_3 from the warmer reservoir.
4. Decouple the system from the reservoir and perform an *adiabatic expansion* from V_4 back to V_1 . There is no heat exchange, that is, $Q_4 = 0$ and the work done by the system is $-W_4$.

Let us now investigate the net work and heat of this idealized closed cycle. In the first two steps of the Carnot cycle, the work $W_{\text{in}} = W_1 + W_2$ is performed on the system and the

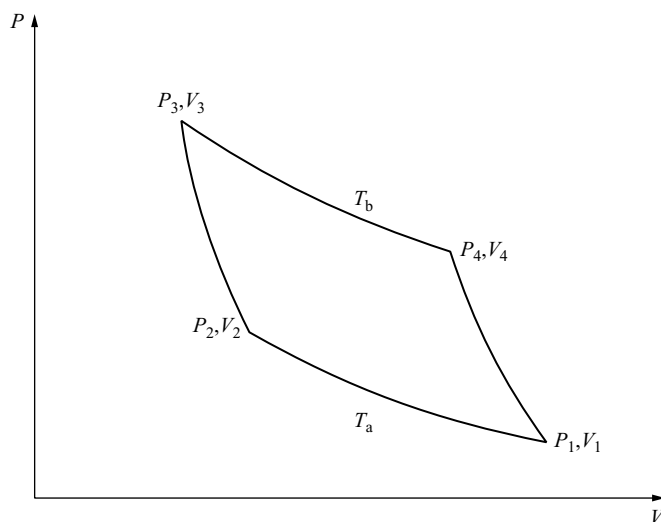


FIGURE 2-1 Schematic representation of the Carnot cycle.

heat $Q_1 = \Delta S \cdot T_a$ is transferred into the cooler heat reservoir. Mathematically, $W = \int PdV$ and W_{in} thus corresponds to the area below the first two segments of the P - V diagram. In the next two steps, the heat $Q_3 = \Delta S \cdot T_b$ is taken from the warmer reservoir and the work $W_{out} = W_3 + W_4$ is given back by the system. From comparison of $W = W_1 + W_2$ we immediately find that more work is released in steps 3 and 4 than was expended in steps 1 and 2. Graphically, the net work corresponds to the inscribed area in the P - V diagram. For the transfer of energy we finally obtain

$$-W = Q_3 - Q_1 = \Delta S \cdot (T_b - T_a) \quad (2.28)$$

Each of the individual steps 1–4 of the Carnot cycle are by themselves of reversible nature. For instance, the compressive step 1 with heat release $-Q_1$ and work W_1 can be reversed by isothermal expansion, where the heat Q_1 is picked up again from the reservoir and the mechanical work $-W_1$ is given back into the system. One could thus quickly conclude that, since all individual steps in the process are reversible, the entire process is reversible and, therefore, the process should convert between mechanical work and heat with an efficiency of $\eta = 1$.

Whereas the first statement is true (the Carnot process is indeed an idealized *reversible* process), the second statement is not. Remember that the total input of energy was the heat Q_3 taken from the warmer reservoir. This energy was converted into the work W , while the heat Q_1 was given to the cooler reservoir. Consequently, the total efficiency of conversion between heat and mechanical work is

$$\eta = \frac{-W}{Q_3} = \frac{Q_1 + Q_3}{Q_3} = 1 + \frac{Q_1}{Q_3} \quad (2.29)$$

Since Q_1 is negative and its absolute value is always less than Q_3 , the efficiency of the process is always equal to or less than one and we can alternatively write

$$\eta = \frac{Q_1 + Q_3}{Q_3} = \frac{T_b - T_a}{T_b} \leq 1 \quad (2.30)$$

An important conclusion from the Carnot process is that, in the description of a closed cycle that converts heat into mechanical work or vice versa, two reservoirs have to be considered, and the efficiency of such a process is directly proportional to the difference in temperature of the two heat reservoirs (equation 2.30).

If a process has an efficiency $\eta < 1$ and only part of the thermal energy Q_3 is converted into mechanical work, we must ask ourselves where has the missing part of the free energy gone? The answer is that the amount Q_1 was transferred from the warmer to the cooler reservoir without having been converted into mechanical work. This process can be interpreted as an *internal process* that transfers heat from the warmer to the cooler reservoir very similar to heat conduction. The entropy change ΔS_{ip} for this process is

$$\Delta S_{ip} = -\frac{\Delta Q}{T_a} + \frac{\Delta Q}{T_b} = \Delta Q \cdot \frac{T_b - T_a}{T_a T_b} \quad (2.31)$$

The transfer of Q_1 from the reservoir with higher temperature T_b to the reservoir with lower temperature T_a produces entropy, and we thus find that heat conduction is an *irreversible process*. The internal entropy production dS_{ip} in differential form reads

$$dS_{ip} = dQ \cdot \frac{T_b - T_a}{T_a T_b} \quad (2.32)$$

and it is a measure for the amount of free energy that cannot be used to produce mechanical energy. The fraction of the free energy that is used for internal entropy production is permanently lost during the process.

The efficiency of any machine converting heat into mechanical work or vice versa is not only restricted by the theoretical limit given by equation (2.29). In reality, *all* processes of heat conduction inside the machine and into the surroundings produce entropy and thus further limit the amount of work that can be produced in the thermomechanical process. The efficiency of the Carnot cycle represents the theoretical upper limit for the efficiency.

Although the Carnot cycle is in principle a reversible process, it produces entropy and makes part of the free energy unavailable for any further production of work. Within our universe, an almost infinite number of entropy producing processes occur at any time, and the total entropy of the universe is steadily increasing. The possible sources for the production of mechanical work are therefore decreasing, and the universe is heading toward a state of perfect disorder. Luckily, the estimated time to arrive there is sufficiently long, so that this collapse is irrelevant for the time being.

Thermodynamic processes have irreversible character if observable macroscopic fluxes of heat and/or matter between different regions of a system or between the system and the surroundings are involved. Typical examples of irreversible processes are heat conduction or atomic diffusion, both of which occur in a preferred direction. Experience tells us that the heat flux always occurs from the warmer to the cooler side. We never observe the macroscopic transport of heat in the opposite direction. Analogously, in diffusion, matter is transported downwards concentration gradients (more exactly: downwards chemical potential gradients). We do not observe diffusion in the opposite direction. From experience, we conclude that heat conduction and diffusion are strictly *irreversible processes*.

All spontaneous processes have a preferred direction and they are irreversible because the reverse process occurs with lower probability. For any spontaneous process we have

$$dS_{ip} \geq 0 \quad (2.33)$$

This is the *second law of thermodynamics*. This law represents a vital link between the worlds of reversible and irreversible thermodynamics and it tells us that all processes that occur spontaneously are accompanied by the production of entropy. The part of the free energy that is *dissipated* (consumed) by the process of internal entropy production is no longer available for the production of mechanical work.

Interestingly, on a microscopic scale, *uphill transport of heat* and *uphill diffusion* occur on a regular basis in the form of *thermal* and *compositional fluctuations*. In nucleation theory, the concept of fluctuations is a vital ingredient of theory (see the Section 6.2 on solid-state nucleation). In a real solution, atoms are never arranged in a perfectly homogeneous way. Instead, one will always observe more or less severe *local* deviations from the average value. Locally, the concentration of one component of a solution can have a significantly different-than-average value and, thus, one could think of a violation of the second law of thermodynamics. However, thermodynamics is a macroscopic art and on a macroscopic basis, there will neither be a net transport of heat nor a net transport of matter against the corresponding potential gradient. Although individual processes can decrease entropy, we will always observe a net production of entropy on a net global scale.

2.1.5 Driving Force for Internal Processes

Heat exchange between two reservoirs will proceed as long as there is a difference in temperature, namely, a temperature gradient. The process stops as soon as both reservoirs are at the

same temperature. Analogously, in atomic diffusion, the *macroscopic* net transport of atoms will proceed until all *macroscopic* concentration gradients are leveled out.

Consider a system consisting of multiple chemical components and multiple phases. We can introduce a new variable ξ , which defines the degree of any internal process that can occur in this system with $0 \leq \xi \leq 1$. Such internal processes are, for instance, the exchange of some amount of element i against element j or the increase of the amount of one phase β at the expense of another phase α . The latter process is known as a *phase transformation* and it frequently occurs simultaneously with an exchange of elements.

Consider a system with one possible internal process. The entropy production caused by this internal process is the *internal entropy production* dS_{ip} (see also previous Section 2.1.4). The *driving force* D for the occurrence of this internal process can then be defined as

$$D = T \cdot \frac{dS_{ip}}{d\xi} \quad (2.34)$$

In physical chemistry, D is often called *affinity* and it quantifies the tendency of a chemical reaction to occur. In phase transformations, the term *driving force* is commonly used. When examining equation (2.34) in more detail, we use the convention that $d\xi$ is defined positive in the direction of the process. Since T is always positive and $dS_i \geq 0$ for a spontaneous process, D must also be > 0 . If $D \leq 0$, the internal process will not occur no matter how long one waits. This fact will be utilized in the subsequent section as a definition of the equilibrium state of a system.

If we consider a closed system under constant temperature and constant pressure, it can be shown (see for instance ref. [Hil98]) that the differential form of the Gibbs energy including contributions from internal processes can be expressed as

$$dG = -SdT + VdP - Dd\xi \quad (2.35)$$

From equation (2.35) and at constant T and P , the driving force for an internal process D is determined as the partial derivative of the Gibbs energy G with respect to the internal variable ξ by

$$D = - \left(\frac{\partial G}{\partial \xi} \right)_{T,P} \quad (2.36)$$

In the present section we have shown that the driving force for an internal process and the internal entropy production are directly related [equation (2.34)] and that the absolute value of D can be obtained as a partial derivative of the Gibbs energy [equation (2.36)]. The former finding will be of great value when we derive evolution equations for the growth of complex precipitates in multicomponent alloys (see Section 6.4.2).

2.1.6 Conditions for Thermodynamic Equilibrium

Based on the Gibbs energy G and given a constant number of atoms in the system, a sufficient condition for *thermodynamic equilibrium* can be given with

$$G(T, P, \mathbf{N}) = \min \quad (2.37)$$

The minimum of G defines a state where no spontaneous reaction will occur in the system because each variation of any state parameter (T, P, \mathbf{N}) will increase the Gibbs energy of the

system and bring it into an unstable state with a positive driving force for at least one internal process. Equation (2.37) together with the definition of the Gibbs energy (2.15) also show that neither a minimum of enthalpy H nor a maximum of entropy S alone can define any such criterion. The ability of a system to produce spontaneous reactions must therefore always be considered as a combined effect of H and S .

Analysis of equation (2.37) directly leads to an alternative condition for thermodynamic equilibrium. Since in equilibrium, G is an extremum, the partial derivative with respect to all state variables must be zero. At constant T and P , this condition reads

$$\left(\frac{\partial G}{\partial \xi}\right)_{T,P} = 0 \quad (2.38)$$

Combination of equations (2.38) and (2.36) yields yet another criterion for equilibrium, which is

$$D \leq 0 \quad (2.39)$$

for all possible internal processes ξ . Or in other words: A system is in equilibrium if the driving forces for all internal processes are less than or equal to zero.

Finally, we want to investigate the important case of an exchange of N_i atoms between two regions I and II of a thermodynamic system. Therefore, we start with equation (2.38). In a thought experiment, N_i atoms are taken from region II and entered into region I. If the system is in equilibrium, for the change of Gibbs energy, we can write

$$\left(\frac{\partial G}{\partial N_i}\right)_{T,P} = \left(\frac{\partial G^I}{\partial N_i}\right)_{T,P} - \left(\frac{\partial G^{II}}{\partial N_i}\right)_{T,P} = 0 \quad (2.40)$$

Since the partial derivatives represent the chemical potentials of the atomic species in the two regions [see equation (2.19)], we can further write

$$\mu_i^I - \mu_i^{II} = 0 \quad (2.41)$$

or

$$\mu_i^I = \mu_i^{II} \quad (2.42)$$

In equilibrium, equation (2.42) must be true for all components and, more general, also for all other thermodynamic potentials. If any thermodynamic potential differs between two regions of the system, there exists a positive driving force for an internal process that causes a reduction of this potential difference. If any potential in the system varies in space, the system is not in equilibrium.

For calculation of multicomponent thermodynamic equilibrium, any of these conditions (2.37), (2.39), or (2.42) can be used. We must be aware, however, that practical evaluation of the preceding formulas is usually more involved than expected from the simplicity of the preceding formulations for equilibrium conditions. The reason for this is the fact that internal processes frequently require a simultaneous variation of multiple state variables due to restrictions of the thermodynamic models, such as mass conservation or stoichiometric constraints. The strategy for minimizing the Gibbs energy in the framework of the sublattice model is outlined in Section 2.2.9 later.

2.2 Solution Thermodynamics

More than 100 years ago, in a single two-part scientific paper [Gib61], Josiah Willard Gibbs (1839–1903) developed the fundamentals of modern solution thermodynamics. The paper titled *On the Equilibrium of Heterogeneous Substances* appeared in 1876 (part II 2 years later) and it is nowadays considered as a giant milestone in this field of science. Most of the physical relations and theoretical concepts of Gibbs' work are now widely used still in their original form, and only minor modifications to his relations have been suggested since.

Solution thermodynamics is concerned with mixtures of multiple components and multiple phases. Consider an experiment where you bring into contact N_A moles of macroscopic pieces of pure substance A and N_B moles of macroscopic pieces of pure substance B (see Figure 2-2, top). The pure substances have molar Gibbs energies of ${}^0G_m^A$ and ${}^0G_m^B$, respectively. After compressing the two substances until no voids exist between the pieces, this conglomerate is called a *mechanical mixture*. When ignoring effects of interfaces between the A and B regions in a first approximation, the total Gibbs energy of the mixture is simply given as the sum of the individual components with ${}^{MM}G = N_A {}^0G_m^A + N_B {}^0G_m^B$. With the amounts of A and B in mole fractions X_A and X_B , the molar Gibbs energy G_m of the mechanical mixture is simply the weighted sum of its pure components

$${}^{MM}G_m = X_A {}^0G_m^A + X_B {}^0G_m^B \quad (2.43)$$

Now consider mixing X_A atoms of sort A and X_B atoms of sort B. In contrast to the previous thought experiment, where a conglomerate of macroscopic pieces was produced, mixing is now performed on the atomic scale (see Figure 2-2, bottom). This so-called *solution* (or *solid solution* for condensed matter) has considerably different properties than the mechanical mixture and we shall investigate these in detail in the following section.

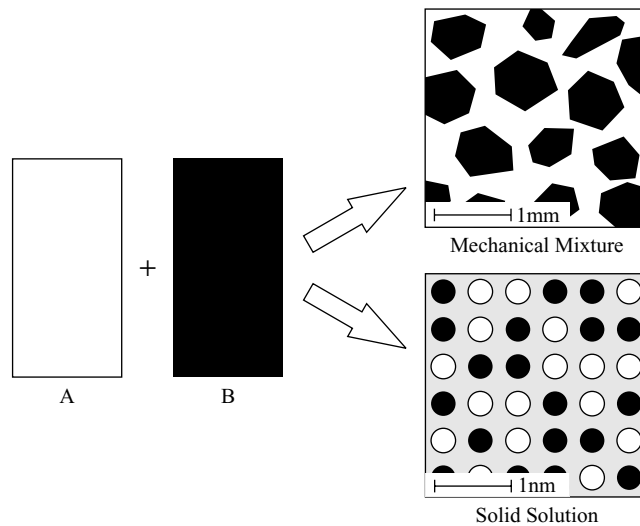


FIGURE 2-2 Two possibilities of mixing two substances A and B, (top) mechanical mixture of macroscopic pieces of the two substances, (bottom) solid solution with mixing on the atomic scale.

2.2.1 Entropy of Mixing

In Section 2.1.4, the entropy S was formally introduced in the framework of reversible thermodynamics [equation (2.13)] and we have seen that S represents an important state function. We have also seen that the concept of entropy production dS_{ip} represents a link between equilibrium thermodynamics and the thermodynamics of irreversible processes. In this section, yet another approach to the thermodynamic quantity *entropy* is presented, which is based on an analysis of the Austrian physicist and philosopher Ludwig Boltzmann (1844–1906). His work on the dynamics of an ensemble of gas particles very clearly illustrates the *irreversible* character of solution thermodynamics, and it gives a handy interpretation of entropy in the framework of *statistical thermodynamics*.

Consider a set of 100 red and 100 blue balls, which at time $t = 0$ are separated. In one hypothetical time step Δt , allow the interchange of two balls. Let this happen by (i) arbitrarily selecting one out of the 200 balls, then (ii) arbitrarily selecting a second ball, and (iii) exchanging them. When allowing the first and second ball also to be identical, the probability that a blue ball is exchanged for a red one is $P = 0.5$. In other words, the macroscopic state of perfect order at $t = 0$ evolves to a state with one ball exchanged between the two regions at $t = \Delta t$ with a probability of $P = 0.5$. Now allow for a consecutive time step: Pick again an arbitrary ball and exchange with another one. The probability that another exchange of blue and red balls occurs is still approximately $P \approx 0.5$. The probability that the exchange of the first time step is reversed is $P = 2/200 \cdot 1/199 \approx 10^{-4}$. Consequently, the probability that the macroscopic state at $t = 0$ is re-established at $t = 2\Delta t$ is much smaller than the probability of finding the system in a new state with two red balls in the blue domain and vice versa.

After sufficient time steps, the probability of the system being in a particular state is equal to the number of possibilities of how to arrange the set of particles in a particular configuration. For instance, the number of possibilities to establish the state at $t = 0$ is equal to 1. There is only one possibility to set up a configuration with all red and blue balls separated. The number of possibilities to establish a state with one ball exchanged between the regions is equal to $100 \cdot 100 = 10^4$. The number of possibilities to establish a state with two balls exchanged is approximately $100 \cdot 99 \cdot 100 \cdot 99 \approx 10^8$ and so forth. Generally, if we have N_B B atoms and N_A A atoms, with $N = N_A + N_B$, the number of possibilities how to arrange this set of atoms is

$$W = \frac{N!}{N_A! \cdot N_B!} \quad (2.44)$$

Consider the same thought experiment, however, this time with only one red and one blue ball. The probability to exchange red against blue in the first time step is $P = 0.5$. The probability for the reverse transformation is also $P = 0.5$. If we consider the two-ball system as a *microsystem*, we can easily find that the principle of time reversibility is fully valid since the probabilities for transformation and reverse transformation are equal. In *macrosystems*, that is, systems that consist of a large number of microsystems, the probability for a process and the corresponding reverse process is *not* equal, and the process has thus a *preferred direction*. The process of exchanging balls in the thought experiment with a large number of balls is an *irreversible* process, although the process of exchanging balls in the microsystem is reversible.

The random exchange of red and blue balls brings the system from an ordered state into a disordered state. Experience tells us that the process never goes in the opposite direction. In atomic diffusion, the probability of an atom to switch position with a particular neighbor is equal to the probability of the atom to switch back to the initial position in the following time step. On a microscopic scale, diffusion is therefore a reversible process. On a macroscopic scale, diffusion tends to reduce concentration gradients and thus brings the system into a state

with a higher degree of disorder. If we bring N_A moles of pure substance A into contact with N_B moles of pure substance B, the irreversible process of diffusion of A atoms into the B-rich region and vice versa will finally lead to a homogeneous solid solution of A atoms and B atoms. We will never observe the reverse process of spontaneous unmixing of a solution and separation of atoms in pure A and pure B containing regions. And we have seen that this is not because it is *impossible*, but because it is *fantastically unlikely*.

In order to quantify this macroscopic irreversibility, Ludwig Boltzmann introduced the term entropy S (which in this context is sometimes also called *Boltzmann entropy*) as being proportional to the natural logarithm of the number of possible states with

$$S = k_B \cdot \ln W \quad (2.45)$$

The proportionality factor k_B is known as the Boltzmann constant ($k_B = 1.38065 \cdot 10^{-23}$ J/K). If we now apply Stirling's approximation ($\ln N! \approx N \ln N - N$, for large N) to equation (2.44), we obtain

$$S = k_B \cdot (N \ln N - N_A \ln N_A - N_B \ln N_B) \quad (2.46)$$

With the relations $X_A = N_A/N$ and $X_B = N_B/N$, the entropy of an A–B solution becomes

$$S = -k_B N \cdot (X_A \ln X_A + X_B \ln X_B) \quad (2.47)$$

Since this definition of entropy is based on the number of possible configurations of a system, it is also called *configurational entropy*. Figure 2-3 shows the entropy contribution of a two-component mixture with $X_B = 1 - X_A$. The curve is symmetric with a maximum entropy at $X_A = X_B = 0.5$.

When considering a solution with one mole of atoms and using the relation $R = k_B N_A$ ($N_A = 6.022142 \cdot 10^{23}$ is the Avogadro constant), the entropy contribution S_i of each component with mole fraction X_i is then given as

$$S_i = -R \cdot X_i \ln X_i \quad (2.48)$$

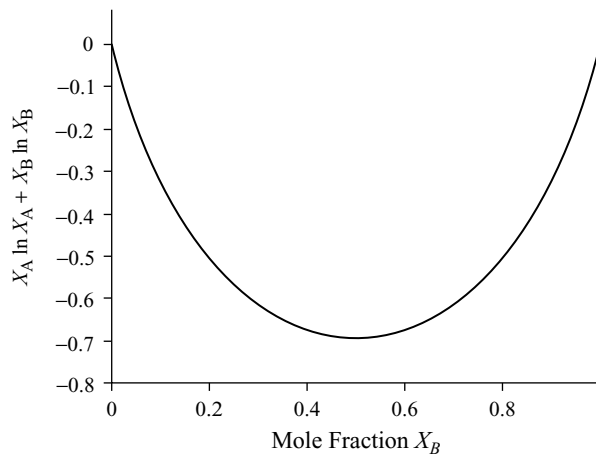


FIGURE 2-3 Configurational entropy contribution in a binary A–B alloy.

and the total molar *ideal entropy of mixing* is

$${}^{\text{IS}}S_{\text{m}} = \sum S_i = - \sum R \cdot X_i \ln X_i \quad (2.49)$$

2.2.2 The Ideal Solution

An *ideal solution* is defined as a solution with zero *enthalpy of mixing* ($\Delta H = 0$) and *ideal entropy of mixing* ${}^{\text{IS}}S$.

The molar Gibbs energy of an ideal solution ${}^{\text{IS}}G_{\text{m}}$ is given by the weighted sum of the molar Gibbs energies of the pure substances ${}^0G_{\text{m}}^i$ and the molar entropy ${}^{\text{IS}}S_{\text{m}}$ as

$${}^{\text{IS}}G_{\text{m}} = \sum \left(X_i \cdot {}^0G_{\text{m}}^i - T \cdot {}^{\text{IS}}S_{\text{m}}^i \right) = \sum X_i \cdot \left({}^0G_{\text{m}}^i + RT \ln X_i \right) \quad (2.50)$$

For the simple case of a binary A–B system with X_{A} atoms of kind A and $(1 - X_{\text{A}})$ atoms of kind B, the molar Gibbs energy is

$$\begin{aligned} {}^{\text{IS}}G_{\text{m}}^{\text{AB}} = & X_{\text{A}} \cdot {}^0G_{\text{m}}^{\text{A}} + (1 - X_{\text{A}}) \cdot {}^0G_{\text{m}}^{\text{B}} + \\ & + RT (X_{\text{A}} \ln X_{\text{A}} + (1 - X_{\text{A}}) \ln(1 - X_{\text{A}})) \end{aligned} \quad (2.51)$$

Figure 2-4 shows the Gibbs energy of the ideal two-component A–B solution. The left end of the diagram represents the properties of pure substance A, the right end represents pure B. The straight line connecting the molar Gibbs energy of the pure substances ${}^0G_{\text{m}}^{\text{A}}$ and ${}^0G_{\text{m}}^{\text{B}}$ represents the molar Gibbs energy of a mechanical mixture ${}^{\text{MM}}G_{\text{m}}$ as described previously. The curved solid line represents the molar Gibbs energies of the mechanical mixture plus the contribution of the configurational entropy, that is, the molar Gibbs energy of an ideal solution.

The *Gibbs energy diagram* shown in Figure 2-4 nicely illustrates the relation between the molar Gibbs energy and the chemical potentials. For a given composition X , the tangent to the Gibbs energy curve is displayed. The intersections of this tangent with the pure A and B

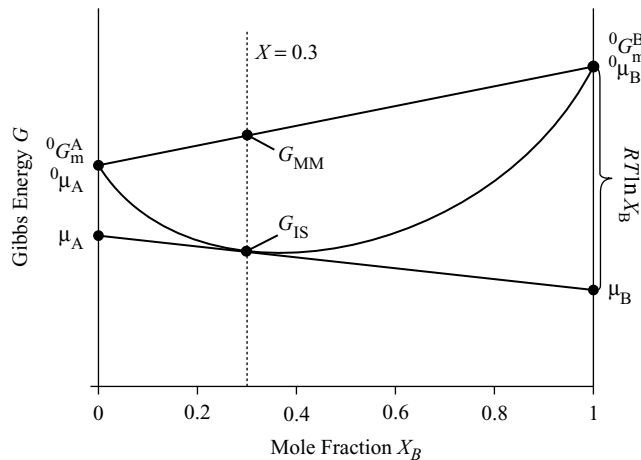


FIGURE 2-4 Molar Gibbs energy of an ideal solution ${}^{\text{IS}}G_{\text{m}}$. ${}^{\text{MM}}G_{\text{m}}$ is the molar Gibbs energy of the mechanical mixture.

sides mark the chemical potentials μ_A and μ_B . Furthermore, from the graph and from equation (2.50), we can see that the difference between the molar Gibbs energy of the pure component ${}^0G_m^i$ and the chemical potential μ_i is equal to $RT \ln X_i$. Moreover, we can identify the molar Gibbs energy of a solution as the weighted sum of the individual chemical potentials [compare with equation (2.27)].

In this context, the molar Gibbs energy of the pure component is often alternatively denoted as ${}^0\mu_i$ and for the chemical potentials ${}^{\text{IS}}\mu_i$ in an ideal solution we have

$${}^{\text{IS}}\mu_i = {}^0\mu_i + RT \ln X_i \quad (2.52)$$

In an ideal solution, the Gibbs energy of mixing ${}^{\text{IS}}\Delta G_{\text{mix}}$ is always negative:

$${}^{\text{IS}}\Delta G_{\text{mix}} = RT \sum (X_i \ln X_i) < 0 \quad (2.53)$$

which means that complete mixing of the pure substances is energetically favorable at all temperatures.

Most mixtures do not behave like ideal solutions. The thermodynamic properties of real solutions are more complex, and we must rely on more complex approaches to capture their behavior. Fortunately, most thermodynamic models take the ideal solution model as a reference, and the behavior of the real mixture is approximated in terms of corrections to ideal solution behavior. In the next section, a common first-order correction is discussed.

2.2.3 Regular Solutions

The ideal solution was introduced as a mixture with zero enthalpy of mixing $\Delta H = 0$ and ideal entropy of mixing ${}^{\text{IS}}S = -\sum RT X_i \ln X_i$. In real solutions, the enthalpy of mixing is almost never zero because this requires, for instance, that the atomic radii of the components are equal (otherwise we have lattice distortion and thus introduce mechanical energy) and that the components behave chemically identical. The latter means that the atomic bond energy between atoms of different kind must be identical to the bond energy for atoms of the same sort.

Consider two pure substances A and B. In a state where A and B are separated, all atomic bonding is of either A–A or B–B type. The sum of all bond energies E in pure A and B are then

$$E_{AA} = \frac{1}{2} Z N_A \cdot \epsilon_{AA} \quad \text{and} \quad E_{BB} = \frac{1}{2} Z N_B \cdot \epsilon_{BB} \quad (2.54)$$

Z is the coordination number and it represents the average number of nearest-neighbor bonds of a single atom. The factor $1/2$ avoids counting bonds between atoms twice. On mixing the substances, some A–A and B–B bonds are replaced by A–B bonds. In a solution of A and B atoms with mole fractions X_A and X_B , the probability of an A atom being a nearest neighbor of a B atom is $P_{AB} = NX_B$ and the probability an A atom neighboring another A atom is $P_{AA} = NX_A$. Since we have NX_A A atoms, we have $ZNX_A X_B$ bonds between A and B atoms. Accordingly, for all bond energies of an A–B solution we have

$$\begin{aligned} E'_{AA} &= \frac{1}{2} Z N_A \cdot X_A \cdot \epsilon_{AA} \\ E'_{BB} &= \frac{1}{2} Z N_B \cdot X_B \cdot \epsilon_{BB} \\ E'_{AB} &= Z N \cdot X_A X_B \cdot \epsilon_{AB} \end{aligned} \quad (2.55)$$

Since the enthalpy of mixing ΔH is inherently related to the change of energy during mixing, that is, the difference in the bond energies before and after mixing, we have

$$\Delta H = E - E' = \frac{1}{2} N X_A X_B Z \cdot (\epsilon_{AA} + \epsilon_{BB} - 2\epsilon_{AB}) \quad (2.56)$$

A mixture where the deviation from ideal solution behavior is described by the enthalpy of mixing according to equation (2.56) is called a *regular solution*. With $\epsilon_{AA} = \epsilon_{BB} = \epsilon_{AB}$, the regular solution model simply reduces to the ideal solution model. It is convenient now to introduce a parameter

$$\omega = Z \cdot (\epsilon_{AA} + \epsilon_{BB} - 2\epsilon_{AB}) \quad (2.57)$$

The enthalpy of mixing ΔH is then

$$\Delta H = \frac{1}{2} X_A X_B N \cdot \omega \quad (2.58)$$

and for the molar Gibbs energy of a regular solution ${}^{\text{RS}}G_m^{\text{AB}}$, with $N = 1$, we finally have

$$\begin{aligned} {}^{\text{RS}}G_m^{\text{AB}} &= {}^0G_m^{\text{AB}} - T \cdot {}^{\text{IS}}S_m + \Delta H_m \\ &= X_A \cdot {}^0G_m^{\text{A}} + (1 - X_A) \cdot {}^0G_m^{\text{B}} \\ &\quad + RT (X_A \ln X_A + (1 - X_A) \ln(1 - X_A)) \\ &\quad + \frac{1}{2} X_A (1 - X_A) \cdot \omega \end{aligned} \quad (2.59)$$

In regular solutions, the mixing characteristics of the two substances depend on the values of temperature T and ω . If the like A–A bonds and B–B bonds are stronger than the unlike A–B bonds, unlike atoms repel each other. The more the difference between the like and unlike bonds, the higher the tendency for unmixing and formation of two separate phases.

When looking closer at equation (2.59), we find that from the last two terms of this equations, the first term, which corresponds to the ideal entropy of mixing, is linearly depending on temperature T . The last term, which is the contribution of regular solution behavior, is independent of T . Consequently, the influence of A–B bonds will be stronger at lower temperature and weaker at higher T .

Figure 2-5 shows the influence of temperature on the Gibbs energy of mixing ${}^{\text{RS}}\Delta G_{\text{mix}}$ assuming a positive enthalpy of mixing $\Delta H > 0$. The upper part of the figure displays the ΔG curves for different temperatures, whereas the lower part shows the corresponding phase diagram. Let us consider an A–B mixture with composition $X_A = X_B = 0.5$. At higher temperature, for example, T_4 or T_5 , the Gibbs energy of mixing is negative because the entropy contribution [right-hand term in equation (2.59)], which favors mixing, dominates over the influence of a positive ΔH , which favors unmixing. We will therefore observe a solid solution of the two substances. With decreasing temperature, the entropy contribution also becomes weaker and weaker until a critical temperature T_{cr} is reached, where the two contributions balance. At this point, we observe a change in curvature of ΔG . At even lower temperatures (T_1 or T_2), the repulsion between unlike atoms becomes dominant over the entropy, and we arrive at a situation where separation of the solution into two phases with different composition is energetically favorable over complete mixing.

Consider now a situation where you hold the A–B mixture above the critical temperature T_{cr} , until the two substances are in complete solution. Now bring the solution to temperature T_1 so fast that no unmixing occurs during cooling. In Figure 2-5, the Gibbs energy of the solution

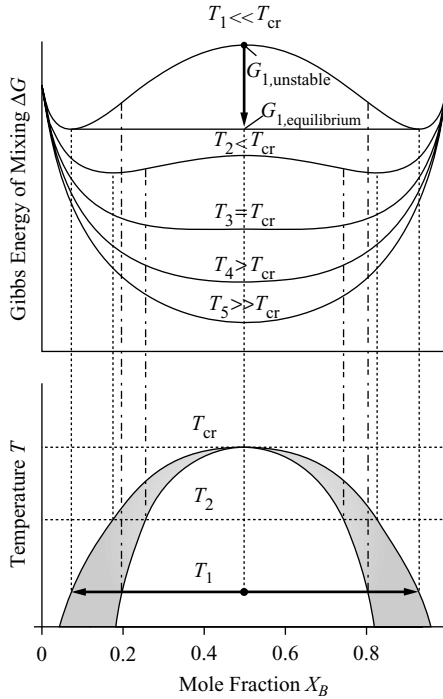


FIGURE 2-5 Molar Gibbs energy of mixing ${}^{\text{RS}}\Delta G_{\text{mix}}$ and phase diagram of a regular solution with $\Delta H > 0$.

in this state is denoted with $G_{1,\text{unstable}}$, which indicates that the system is not in equilibrium because the internal process of unmixing of the two substances can decrease its Gibbs energy. This is indicated by bold arrows in the diagrams. The Gibbs energy of the unmixed state is denoted as $G_{1,\text{equilibrium}}$ and it represents the weighted sum of the Gibbs energies of the two unmixed phases as indicated by the solid horizontal line.

If unmixing of the solution occurs, the two new phases have compositions that are given by the intersections of the common tangent with the ΔG curve. Note that the common tangent represents the lowest possible Gibbs energy that the two coexisting phases can achieve. The compositions obtained by this graphical procedure are indicated by the vertical dashed lines connecting the upper and lower diagrams. The dash-dotted lines mark the inflection points of the ΔG curves. These are important in the theory of *spinodal decomposition*. We will now derive an expression for the critical temperature T_{cr} .

According to Figure 2-5, the critical temperature below which phase separation occurs is characterized by a horizontal tangent and an inflection point at $X = 0.5$. The latter is defined as the point where the second derivative of the ΔG curve is zero. From equation (2.59), for an A–B regular solution, we obtain

$$\begin{aligned}\Delta G &= \frac{1}{2}X_A(1 - X_A)Z \cdot \omega - RT \cdot (X_A \ln X_A + (1 - X_A) \ln(1 - X_A)) \\ \frac{\partial \Delta G}{\partial X_A} &= \frac{1}{2}(1 - 2X_A)Z \cdot \omega - RT \cdot (\ln X_A - \ln(1 - X_A)) \\ \frac{\partial^2 \Delta G}{\partial X_A^2} &= -Z \cdot \omega + RT \cdot \left(\frac{1}{X_A} + \frac{1}{1 - X_A} \right)\end{aligned}\quad (2.60)$$

The critical temperature of a regular solution is evaluated from setting the second derivative zero at a composition $X_A = X_B = 0.5$. We get

$$T_{\text{cr}} = \frac{Z\omega}{4R} \quad (2.61)$$

Let us shift our focus back to the enthalpy of mixing ΔH and the chemical potentials, and let us introduce the quantity $^{\text{RS}}\Delta G^{\text{ex}}$, which represents the *excess Gibbs energy of mixing* of a regular solution, with

$$^{\text{RS}}\Delta G^{\text{ex}} = \Delta H \quad (2.62)$$

From equation (2.56), using $X_A = N_A/(N_A+N_B)$, $X_B = N_B/(N_A+N_B)$ and $N = N_A+N_B$, we have

$$^{\text{RS}}\Delta G^{\text{ex}} = \frac{1}{2} \frac{N_A N_B}{N_A + N_B} \cdot \omega \quad (2.63)$$

For the additional contribution to the chemical potential μ_A^{ex} , we obtain

$$\begin{aligned} \mu_A^{\text{ex}} &= \left(\frac{\partial ^{\text{RS}}\Delta G_{\text{m}}^{\text{ex}}}{\partial N_A} \right)_{N_B} \\ &= \frac{1}{2} \left(\frac{N_B}{N_A + N_B} - \frac{N_A N_B}{(N_A + N_B)^2} \right) \cdot \omega \\ &= \frac{1}{2} \omega X_B^2 \end{aligned} \quad (2.64)$$

and the chemical potential of a regular solution can be expressed as

$$^{\text{RS}}\mu_A = {}^0\mu_A + RT \ln X_A + \frac{1}{2} \omega X_B^2 \quad (2.65)$$

2.2.4 General Solutions in Multiphase Equilibrium

The formalism of the regular solution model, which has been presented in Section 2.2.3 for binary A–B solutions, is symmetric with respect to the composition variables X_A and $X_B = 1 - X_A$. In general (“real”) solutions, the Gibbs energy – composition ($G - X$) curves have nonsymmetric shape, and a single phenomenological parameter such as ω is not sufficient to describe more complex atomic interactions on thermodynamic grounds.

In a traditional approach to describe the Gibbs energy of *general solutions*, the *chemical activity* a is introduced. The activity a_i of a component i and the chemical potential μ_i are related by

$$\mu_i = {}^0\mu_i + RT \ln a_i \quad (2.66)$$

Comparison of the chemical potential in an ideal solution (equation (2.52)) with equation (2.66) suggests introduction of an additional quantity, the *activity coefficient* f_i , which is related to the mole fraction X_i and the activity a_i with

$$a_i = f_i X_i \quad (2.67)$$

According to the definitions (2.66) and (2.67), the activity coefficient f_i can be considered as the thermodynamic quantity that contains the deviation of the thermodynamic properties of a general solution from ideal solution behavior. The activity a and the activity coefficient f are

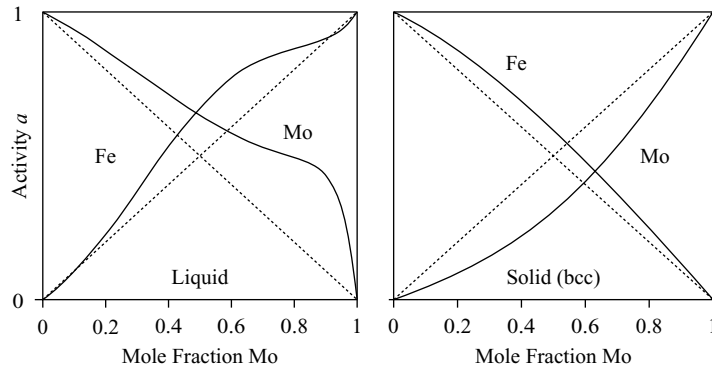


FIGURE 2-6 Activity of Fe and Mo in the liquid and solid (bcc) state of an Fe–Mo solution at 1700°C calculated from computational thermodynamics.

usually complex functions of temperature T and composition \mathbf{X} . Figure 2-6 shows the activities of iron and molybdenum in the liquid and solid phases at 1700°C as obtained from computational thermodynamics. The difference of real solution and ideal solution behavior is observable as the deviation of the activity curves from the straight dotted diagonals representing the ideal solution behavior.

Activities of elements in a solution can be obtained from suitable experiments, and the phenomenological coefficients of thermodynamic models can be optimized on these data to give a good representation of the atomic interactions in the solution. Based on this information, multiphase equilibria can be determined using computational techniques such as Gibbs energy minimization (see Section 2.2.9) or equilibration of chemical potentials.

In general multiphase equilibrium with m stable phases $\alpha_1, \alpha_2, \dots, \alpha_m$, the chemical potentials of all elements are identical in each phase (compare also Section 2.1.6). Accordingly, for each component i , we can write

$$\mu_i^{\alpha_1} = \mu_i^{\alpha_2} = \dots = \mu_i^{\alpha_m} \quad (2.68)$$

For two phases α_r and α_s , after insertion of equation (2.67), we find

$$\frac{{}^0\mu_i^{\alpha_s} - {}^0\mu_i^{\alpha_r}}{RT} = \ln \frac{a_i^{\alpha_r}}{a_i^{\alpha_s}} \quad (2.69)$$

Equation (2.69) can be used to evaluate the partitioning behavior of an element between two phases in equilibrium. With equation (2.67), the ratio between the mole fractions X_i in the two phases α_r and α_s is given with

$$\ln \frac{X_i^{\alpha_r}}{X_i^{\alpha_s}} = \frac{{}^0\mu_i^{\alpha_s} - {}^0\mu_i^{\alpha_r}}{RT} - \ln \frac{f_i^{\alpha_r}}{f_i^{\alpha_s}} \quad (2.70)$$

Figure 2-7 shows the Gibbs energy diagram of the liquid and solid Fe–Mo phases in thermodynamic equilibrium at a temperature of 1700°C. Again, the curves have been calculated from computational thermodynamics. On the left side, that is, the iron-rich side of the $G - X$ diagram, the Gibbs energy of the solid bcc phase is lower than the liquid. Therefore, the solid phase is stable. In a composition region of approximately $0.46 < X_{\text{Mo}} < 0.82$, liquid and solid Fe–Mo are in two-phase equilibrium. The Gibbs energy of the two-phase mixture is given by

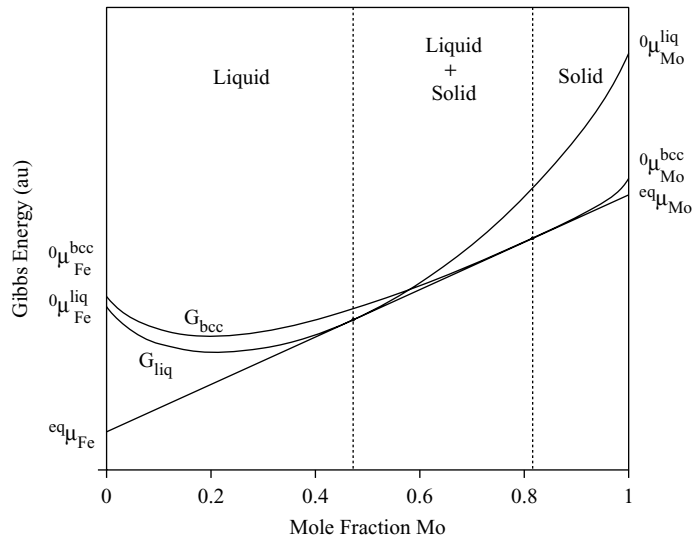


FIGURE 2-7 Calculated Gibbs energy–composition diagram for liquid and solid (bcc) Fe–Mo at 1700°C.

the common tangent to the G – X curve of the two phases, which represents the state of lowest possible Gibbs energy. The composition of the solid and liquid phases are determined by the intersections of the tangent with the Gibbs energy curves. In equilibrium, the chemical potentials ${}^{\text{eq}}\mu_{\text{Fe}}$ and ${}^{\text{eq}}\mu_{\text{Mo}}$ are read at the intersections of the common tangent with the ordinate axes. At highest Mo-content, solid bcc molybdenum is stable.

2.2.5 The Dilute Solution Limit—Henry’s and Raoult’s Law

In a dilute solution, *solute* atoms are dissolved in a matrix of *solvent* atoms. The concentration of solute atoms is small compared to the solvent atoms, that is, $c_{\text{solute}} \ll c_{\text{solvent}}$. In an A–B system, where the concentration of B is small compared to A, and B thus represents the solute, the probability that two B atoms occur next to each other is assumed to be so low that the interactions of two B atoms can be neglected.

The dilute solution approximation is a popular simplification and assumption in theoretical modeling because, on one hand, many alloys systems of technical relevance have dilute solution character (e.g., microalloyed steels) and, on the other hand, theoretical models for precipitation kinetics, etc., can be substantially simplified compared to the general solutions for concentrated alloys.

In this section, thermodynamic properties of a solution in the dilute solution limit will be explored briefly. For the theoretical discussion of these properties, the regular solution model, which has been introduced in Section 2.2.3, is employed. Accordingly, comparison of the general expressions for the activity (2.66) and the activity coefficient (2.67) with the regular solution chemical potential (2.65) for the solute B yields

$${}^0\mu_{\text{B}} + RT \ln a_{\text{B}} = {}^0\mu_{\text{B}} + RT \ln X_{\text{B}} + \frac{1}{2}\omega X_{\text{A}}^2 \quad (2.71)$$

and for the activity coefficient f_{B} in the regular solution model we have

$$RT \ln f_{\text{B}} = \frac{1}{2}\omega X_{\text{A}}^2 \quad (2.72)$$

In the limiting case of a dilute solution where $X_B \ll 1$, it follows that $X_A \approx 1$ and we have

$$RT \ln f_B \approx \frac{1}{2}\omega \quad \text{and} \quad f_B \approx \exp \frac{\omega}{2RT} \quad (2.73)$$

Thus, the activity coefficient f of a *solute* in the dilute solution limit is approximately constant (independent of composition), and the *activity* is approximately linear proportional to its mole fraction with the proportionality constant given by equation (2.73). This is called *Henry's law*, after the English chemist William Henry (1775–1836). Figure 2-8 shows the activity of a regular solution with $\Delta H > 0$ indicating Henry's law by a bold dashed arrow.

When looking at the activity coefficient f_A of the *solvent* in the dilute solution limit, with $X_A \approx 1$ and $X_B^2 \approx 0$ we find

$$RT \ln f_A \approx 0 \quad \text{and} \quad f_A \approx 1 \quad (2.74)$$

Accordingly, for the solvent, the excess contribution to the Gibbs energy of mixing disappears and the activity of the solvent is approximately equal to its mole fraction. The activity of the solvent is only depending on its own properties, and it is independent of the properties of the solute (see Figure 2-8). This is called *Raoult's law*, according to Francois-Marie Raoult, a French physicist and chemist (1830–1901).

2.2.6 The Chemical Driving Force

Consider a binary system A–B with two phases α and β . Let α be a solution phase and β be a precipitate phase with limited stoichiometry, that is, the α phase is stable over the entire composition range and β exists only within a limited region. Figure 2-9 shows the Gibbs energy diagram of this system.

Let us assume that the overall composition is $X_B = 0.3$ and, initially, the system consists of only α . The overall composition is marked by a dashed line. The molar Gibbs energy of this

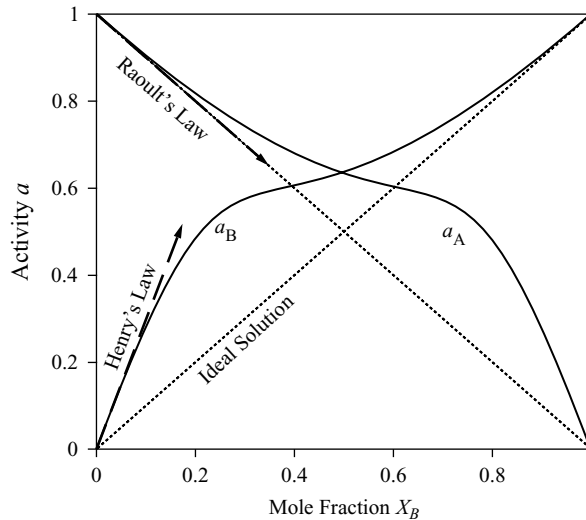


FIGURE 2-8 Activities and activity coefficients of ideal and regular solution with Raoult's and Henry's law in the dilute solution limit.

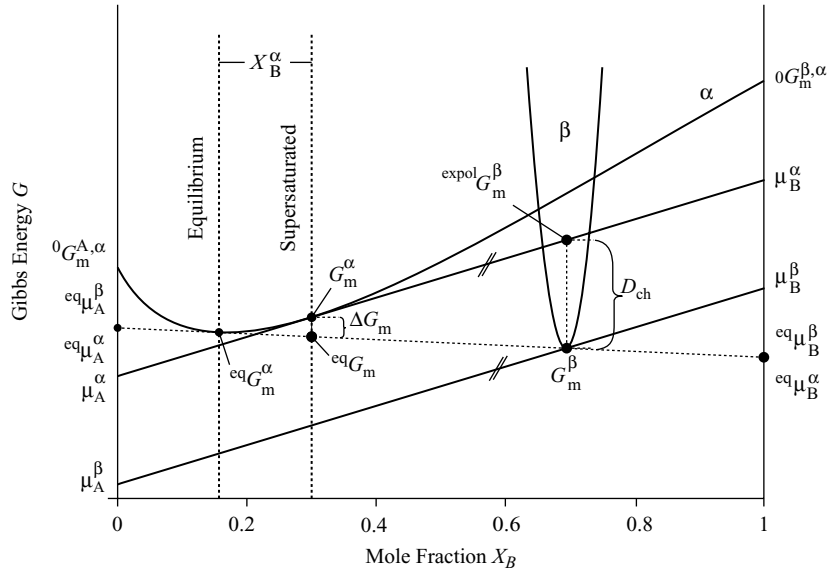


FIGURE 2-9 Calculation of the chemical driving force D_{ch} of a binary solution A–B with two phases α and β .

configuration is indicated by G_m^α and the chemical potentials in the α phase are μ_A^α and μ_B^α . From Figure 2-9 it is evident that the total Gibbs energy of the system (which is G_m^α in the initial state) can be decreased by ΔG_m if a certain amount of the phase β is formed such that the final (equilibrium) state is determined by a common tangent to the molar Gibbs energy curve of both phases. The decrease is given by the difference between the Gibbs energy of the initial state ΔG_m^α and the final state ${}^{eq}\Delta G_m$ with

$$\Delta G_m = {}^{eq}G_m - G_m^\alpha \quad (2.75)$$

Consider again the initial configuration of only α phase. Since α has more B atoms in solution than it would have in equilibrium, that is, the Gibbs energy of the system can be decreased by formation of B-rich β , this configuration is denoted as a *supersaturated solution*. Now imagine taking out some A and B atoms from the solution to form a small amount of the new phase β , and consider the α phase as an infinite reservoir that does not change its composition in this process. The change in Gibbs energy ΔG_m^β counted per one mole of atoms is then given as the difference between the Gibbs energy of the atoms taken from the solution $X_A^\beta \mu_A^\alpha + X_B^\beta \mu_B^\alpha$ and the energy G_m^β after transformation into β with

$$\Delta G_m^\beta = -(X_A^\beta \mu_A^\alpha + X_B^\beta \mu_B^\alpha - G_m^\beta) \quad (2.76)$$

Graphically, we can identify ΔG_m^β as the difference between the extrapolated Gibbs energy of the α phase to the composition of the β phase ${}^{expol}G_m^\beta$ and the molar Gibbs energy G_m^β of β .

According to equation (2.36), the driving force D for an internal reaction is equal to the derivative of the molar Gibbs energy with respect to the internal variable ξ , which represents the

extent of the reaction. Since the extent of β formation is determined by the number of moles of the new phase formed N^β , we have

$$d\xi = dN^\beta \quad (2.77)$$

and

$$D = - \left(\frac{\partial G}{\partial N^\beta} \right)_{T,P,\mathbf{N}} = -\Delta G_m^\beta. \quad (2.78)$$

Finally, we obtain the general form of the *chemical driving force* D_{ch} for formation of a new phase β in an α matrix as

$$D_{\text{ch}} = - \left(\sum X_i^\beta \mu_i^\alpha - G_m^\beta \right) \quad (2.79)$$

So far, we have used the Gibbs energy of the new phase G_m^β without looking closer into how this quantity is defined. From Figure 2-9 we recognize that the β phase has a finite compositional range of stability. Therefore, a variety of possible “choices” for the composition of the new phase exists, and we could assign any composition to it as long as we make sure that its Gibbs energy G_m^β is lower than the corresponding extrapolated Gibbs energy $G_m^{\text{extrapol}}^\beta$. A common and pragmatic approximation to fix this ambiguity is to assume that the β phase has exactly the composition that yields the highest tendency for formation, that is, the composition with maximum chemical driving force D_{ch} .

From graphical considerations, the maximum driving force is obtained by the parallel tangent to the two phases in the Gibbs energy diagram. This procedure is called *tangent construction*. It should be emphasized that the tangent construction also leads to equal differences between the chemical potentials for all components $\mu_i^\beta - \mu_i^\alpha$ in the two phases. This fact can be utilized in practical calculation of X_B^β .

Approximation of the composition of the new phase with the maximum chemical driving force criterion allows us to identify a unique composition that can be used in the analysis of phase transformation processes. This selection criterion is reasonable at least in the context of equilibrium thermodynamics, where all internal processes have sufficient time to come to a rest in every time increment. However, during many dynamic processes, such as solid-state precipitation or solidification, the new phases often form with compositions that can significantly deviate from the maximum driving force composition. These deviations are due to kinetic constraints and the actual reaction path is determined by alternative processes, such as maximum Gibbs energy dissipation, which is introduced in Section 6.4.2, where the thermodynamic extremal principle is introduced and utilized to develop evolution equations for multicomponent precipitate growth.

2.2.7 Influence of Curvature and Pressure

In this section we will investigate the influence of pressure on the equilibrium between two phases. The pressure P can be applied in two ways: On one hand, it can act on the two phases in the form of a *hydrostatic pressure* and thus affect both phases equally. Since a small hydrostatic pressure will influence the thermodynamic properties of both phases in approximately the same way, its influence on the two-phase equilibrium is usually weak and we will not consider this case further. On the other hand, pressure can act on the phases in the form of *curvature induced pressure*. In solid matter, this pressure originates from the interfacial energy of the curved interface between a precipitate and the embedding matrix, and it mainly affects the precipitate phase while the thermodynamic properties of the matrix phase remain almost unaltered. The influence of *curvature induced pressure* on the equilibrium state can be substantial, and it is particularly large when the precipitates are very small, that is, their radius is in the order of a few nanometers or less. Curvature induced pressure is the driving mechanism behind a number of important

metallurgical processes, such as Ostwald ripening or grain/precipitate coarsening, and its origin will be briefly explored now.

Consider a sphere with radius ρ . Cut the sphere in half. The length around the sphere is $2\pi\rho$ and the interfacial tension is thus $2\pi\gamma\rho$. γ is the specific interfacial energy in units of J/m^2 and it denotes the energy that is stored in unit area of interface. It can also be expressed in units of force per length, N/m , and it thus also represents a specific force, namely, force per unit length. The force due to surface tension must be compensated by a pressure force inside the sphere, which is $\pi\rho^2P$. The extra pressure P inside a sphere due to the curvature of the interface then is

$$P = \frac{2\gamma}{\rho} \quad (2.80)$$

This simple relation tells us that the pressure difference between a precipitate and the surrounding matrix is inversely proportional to the precipitate radius. It is thus the larger the smaller the precipitate is. Let us now look at the influence of this pressure on the thermodynamic properties of the precipitate phase. If we assume that the phases are incompressible, the Gibbs energy of the precipitate phase will be increased by $P^\beta V_m^\beta$. V_m^β is the molar volume of β and P^β is the extra pressure acting on the β particle. The Gibbs energy of the precipitate phase β is then

$$G_m^\beta = H_m^\beta - TS_m^\beta + \Delta G_m^P \quad (2.81)$$

and the excess Gibbs energy ΔG_m^P due to interfacial curvature is

$$\Delta G_m^P = P^\beta V_m^\beta = \frac{2\gamma V_m^\beta}{\rho} \quad (2.82)$$

Figure 2-10 shows the Gibbs energy curves of the α matrix and the β precipitate with and without the effect of curvature induced pressure. Accordingly, the excess Gibbs energy ΔG_m^P shifts the Gibbs energy curve of the precipitate to higher values. The solid curve in the diagram

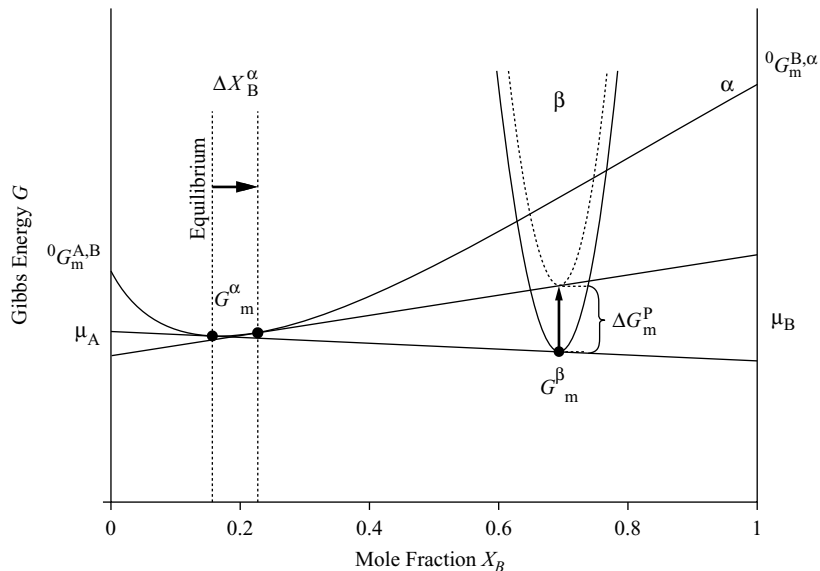


FIGURE 2-10 Influence of curvature induced pressure on a precipitate β in an α matrix.

represents the unstressed Gibbs energy of β , and the dashed line is the Gibbs energy including the effect of P . An important result of this analysis is the observation that, due to the increase of internal pressure in the β phase, simultaneously, the equilibrium concentration of B in the solution phase α is shifted to the right, that is, to higher mole fraction X_B . Apparently, this effect is stronger the higher the extra pressure P is and, consequently, the equilibrium concentration of B around a small particle is higher than the equilibrium concentration of B around a large particle. If a small and a large precipitate are located next to each other, a composition gradient will exist between the two, which will cause a net flux of B atoms from the smaller to the larger particle. This effect is known as coarsening or Ostwald ripening.

Finally, we explore the relation between the pressure increase in the precipitate and the change in equilibrium concentration of component B in the surrounding matrix. From Figure 2-10 we can see that the slope of the tangent to the Gibbs energy curves of the stressed and unstressed β phases can be approximated with

$$\frac{\partial G_m^\alpha}{\partial X_B} = \frac{G_m^\beta - G_m^\alpha}{X_B^\beta - X_B^\alpha} \quad \text{and} \quad \frac{\partial G_m^{\alpha,P}}{\partial X_B} = \frac{G_m^{\beta,P} - G_m^{\alpha,P}}{X_B^{\beta,P} - X_B^{\alpha,P}} \quad (2.83)$$

with the superscript “ P ” denoting the variables under the influence of pressure. Furthermore, we can approximate the curvature of the Gibbs energy curve with

$$\frac{\partial^2 G_m^\alpha}{\partial X_B^2} = \frac{\frac{\partial G_m^{\alpha,P}}{\partial X_B} - \frac{\partial G_m^\alpha}{\partial X_B}}{X_B^{\alpha,P} - X_B^\alpha} \quad (2.84)$$

On substitution of equation (2.83) into (2.84) and with the assumption that the distance $X_B^\beta - X_B^\alpha$ is sufficiently large compared to the shift in compositions in the individual phases, we find

$$X_B^{\alpha,P} - X_B^\alpha = \left[(X_B^\beta - X_B^\alpha) \cdot \frac{\partial^2 G_m^\alpha}{\partial X_B^2} \right]^{-1} \cdot PV_m^\beta \quad (2.85)$$

Equation (2.85) provides an approximation of the equilibrium composition of B in the vicinity of a precipitate, if matrix and precipitate composition are sufficiently different from each other. However, the equation still contains the second derivative of the Gibbs energy, a quantity that might not always be readily available. If the solution behaves approximately like an ideal solution, which it does at least in the dilute solution limit of low X_B , we can substitute the derivative by

$$\frac{\partial^2 G_m^\alpha}{\partial X_B^2} \approx \frac{RT}{X_A} + \frac{RT}{X_B} = \frac{RT}{X_A X_B} \approx \frac{RT}{X_B} \quad (2.86)$$

With this approximation and equation (2.82), we finally have

$$X_B^{\alpha,P} - X_B^\alpha = \frac{X_B^\alpha}{X_B^\beta - X_B^\alpha} \cdot \frac{2\gamma V_m^\beta}{RT} \cdot \frac{1}{\rho} \quad (2.87)$$

or

$$X_B^{\alpha,P} = X_B^\alpha \cdot \left(1 + \frac{2\gamma V_m^\beta}{(X_B^\beta - X_B^\alpha) \cdot RT} \cdot \frac{1}{\rho} \right) \quad (2.88)$$

Equation (2.88) is the linearized form of the well-known *Gibbs–Thomson equation*. In this form, it represents a reasonable approximation for larger precipitate radii. In a more general approach, it can be shown that this equation becomes

$$X_B^{\alpha,P} = X_B^\alpha \cdot \exp\left(\frac{2\gamma V_m^\beta}{(X_B^\beta - X_B^\alpha) \cdot RT} \cdot \frac{1}{\rho}\right) \quad (2.89)$$

This version of the *Gibbs–Thomson equation* gives a better approximation for small precipitates.

2.2.8 General Solutions and the CALPHAD Formalism

The energetic interactions between individual atoms (bonding energies) in a multicomponent alloy are extremely complex, and treating each of these on a rigorous basis is still out of reach of current computational capabilities. Instead, solution thermodynamics tries to treat the properties of solutions on a macroscopic, *phenomenological* scale and utilizes statistical methods and average quantities to bypass treating the individual atomic interactions.

It was already emphasized that *general* (or *real*) solutions rarely behave like ideal or regular solutions. It is convenient, however, to take the ideal solution model as a reference state and express the properties of the real solution in terms of excess quantities. The Gibbs energy of a *real* solution can thus be written as

$$G_m = {}^0G_m - T^{\text{IS}}S_m + {}^{\text{ex}}G_m \quad (2.90)$$

${}^{\text{ex}}G_m$ is the *excess Gibbs energy* and it contains all interactions between atoms in excess to ideal solution behavior.

From a mathematical point of view, one could take arbitrary functions to represent ${}^{\text{ex}}G_m$ as long as it is ensured that the functions go to zero at the limits of the pure components. A very popular formalism to describe the excess Gibbs energy is based on a polynomial series proposed by Redlich and Kister in 1948. Accordingly, we have

$${}^{\text{ex}}G_m = \sum_{i \neq j}^k L_{ij} \cdot X_i X_j (X_i - X_j)^k \quad (2.91)$$

The indices i and j represent two components and the interaction parameters ${}^kL_{ij}$ describe the intensity of the excess interaction between components i and j . The exponent $k \geq 0$ is an integer and defines the order of the so-called Redlich–Kister polynomial (see Figure 2-11).

It is important to note that the Redlich–Kister polynomials are not symmetric with respect to i and j . Consequently, we must be careful not to exchange the order of the components when evaluating the interaction terms. If we only consider interactions of zeroth order, equation (2.91) reduces to

$${}^{\text{ex}}G_m = {}^0L_{ij} \cdot X_i X_j \quad (2.92)$$

By comparison with equation (2.58), we can identify the relation between the zeroth-order interaction parameter and the regular solution parameter ω with

$${}^0L_{ij} = \frac{1}{2}N \cdot \omega \quad (2.93)$$

Consequently, if the thermodynamic description of the solution involves only zeroth-order interactions, the solution behaves like a regular solution. Higher-order interactions between atoms are described by Redlich–Kister polynomials with $k \geq 1$.

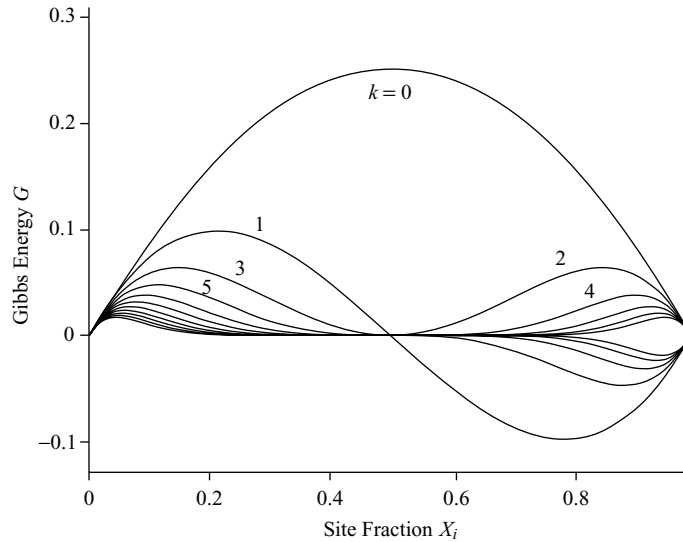


FIGURE 2-11 Shape of Redlich–Kister polynomials for $k \geq 0$, $L_{ij} = 1$ and $X_i = 1 - X_j$.

Solution Phases with Multiple Sublattices

In the previous chapters we have introduced a model that describes the thermodynamic properties of general multicomponent solutions. However, we have assumed that the atoms are arranged on a single lattice and that all lattice sites are equivalent. In a real crystal, this is rarely the case and certain types of atoms are located on separate *sublattices*. A typical example of a multisublattice phase is, for instance, the iron-based carbide cementite, which has the fixed stoichiometry Fe_3C and where the C atoms occupy a different sublattice than the Fe atoms. In higher-order systems, part of the Fe atoms on the substitutional sublattice of the cementite phase can be substituted by elements such as Cr or Mn. These atoms will never replace C atoms, which reside on a different sublattice, but only occupy sites on the Fe sublattice. The stoichiometry of the M_3C carbide, where “M” stands for the elements Fe, Cr, and Mn, is therefore usually written as $(\text{Fe}, \text{Cr}, \text{Mn})_3\text{C}$, to indicate that some components share common sublattices.

In thermodynamic modeling of phases with multiple sublattices, the following assumptions are made:

1. On each sublattice, consider a number of one formula unit of atoms, that is, one mole of atoms.
2. Assume random mixing on each sublattice, but no mixing across the sublattices.
3. The amount of an element in a phase is described by the *site fraction* variable y_i^s , which denotes the number of moles of atoms of type i on each sublattice s .

For the site fractions on each sublattice we have

$$\sum_i y_i^s = 1 \quad (2.94)$$

and

$$0 \leq y_i^s \leq 1 \quad (2.95)$$

Equations (2.94) and (2.95) can be viewed as constraints of the thermodynamic model, namely, the *sublattice model*. They have to be taken into account when manipulating thermodynamic quantities or when evaluating phase equilibria.

Most metals have fcc, bcc, or hcp crystal structure, and they are modeled by two sublattices to take into account that interstitial atoms, such as C or N, can occupy *interstitial* sites between the *substitutional* sites of the metal atoms. For instance, pure fcc or bcc Fe can dissolve a certain amount of interstitial C, and the Fe solution phases are modeled such that the first sublattice holds all substitutional elements whereas the interstitial sublattice holds the interstitial atoms. Since equation (2.94) must be generally valid, for the interstitial sublattice, an additional, hypothetical component named *vacancy* must be introduced. Vacancies in the sublattice model represent empty *interstitial* lattice positions and they are denoted by the symbol “Va.” For the relation between the mole fractions X_i in a solution and the site fractions y_i^s we then have

$$X_i = \frac{\sum_s (b^s \cdot y_i^s)}{\sum_s (b^s \cdot (1 - y_{\text{Va}}^s))} \quad (2.96)$$

The factors b^s denote the number of lattice sites on each sublattice and they thus define the stoichiometry of the phase. Equation (2.96) relates the total mole fraction X_i of a component in a phase to the sum of its individual site fractions y_i^s on each sublattice. Since we have assumed that each sublattice holds one mole of atoms, the total mole fraction sum must be divided by the total number of moles in the phase. The term in parentheses $(1 - y_{\text{Va}}^s)$ takes into account that interstitial vacancies are only hypothetical components and, therefore, do not contribute to the total number of atoms.

When calculating the Gibbs energy for a multisublattice phase, we have to be aware of the fact that the Gibbs energy of the pure components now has different meaning than in a one-sublattice solution model. Take, for instance, the usual sublattice model for the iron fcc phase in the Fe–C system, which is $(\text{Fe})_1(\text{C}, \text{Va})_1$. The substitutional sublattice is entirely occupied by Fe, whereas the interstitial sublattice contains C atoms and/or vacancies. When writing the Gibbs energy of Fe, we have to define which component, C or Va, should occupy the interstitial sublattice. We therefore have two configurations $(\text{Fe})_1(\text{C})_1$ and $(\text{Fe})_1(\text{Va})_1$ instead of only one previously, both of them representing a “pure” component. When using the colon symbol “:” as a separator for sublattices, we have the Gibbs energies ${}^0G_{\text{Fe:Va}}^{\text{fcc}}$ and ${}^0G_{\text{Fe:C}}^{\text{fcc}}$. When further assuming ideal entropy of mixing on the interstitial sublattice, the Gibbs energy of one mole of Fe–C solution in the sublattice model becomes

$$G^{\text{fcc}} = y_{\text{Fe}}^0 y_{\text{Va}}^1 \cdot {}^0G_{\text{Fe:Va}}^{\text{fcc}} + y_{\text{Fe}}^0 y_{\text{C}}^1 \cdot {}^0G_{\text{Fe:C}}^{\text{fcc}} + RT(y_{\text{C}}^1 \ln y_{\text{C}}^1 + y_{\text{Va}}^1 \ln y_{\text{Va}}^1) + {}^{\text{ex}}G^{\text{fcc}} \quad (2.97)$$

The general expression for the Gibbs energy of a phase α in the two-sublattice model is

$$G_{\text{m}}^{\alpha} = \sum_{t \neq s, i, j} y_i^s y_j^t \cdot {}^0G_{i:j}^{\alpha} + \sum_{s, i} RT y_i^s \ln y_i^s + {}^{\text{ex}}G^{\text{fcc}} \quad (2.98)$$

and the excess Gibbs energy in terms of Redlich–Kister polynomials

$${}^{\text{ex}}G_{\text{m}} = \sum_{i \neq j, m} {}^k L_{i,j:m} \cdot y_i y_j y_m (y_i - y_j)^k + \sum_{i, j \neq m} {}^k L_{i,j:m} \cdot y_i y_j y_m (y_j - y_m)^k \quad (2.99)$$

Extension of equations (2.98) and (2.99) to more sublattices is straightforward and will not further be discussed in this book. The reader is referred to references [Hil98, SM98].

Magnetic Excess Energy

In the previous sections we have introduced a thermodynamic model for solutions with multiple sublattices. Based on the ideal solution model, we have defined the excess Gibbs energy $^{\text{ex}}G_m$, which contains all deviations from ideal solution behavior. However, not all physical effects can easily be incorporated into this mathematical formalism and we have to model these effects separately. A typical example for this is magnetic ordering in ferromagnetic substances such as Fe, Co, or Ni. Figure 2-12 shows the specific heat capacity of pure Fe with and without contribution from magnetic ordering.

Among a number of others, a nowadays widely used model for magnetic ordering has been suggested by G. Inden [Ind76]. It describes the transition from the disordered state to the magnetically ordered state by a series based on the normalized temperature $\tau = T/T_C$ and the magnetic moment β_m . T_C is the Curie temperature and it is defined by the inflection point of the magnetic entropy. For the Gibbs energy contribution of magnetic ordering $^{\text{mo}}G_m$, above the Curie temperature $\tau > 1$, we have

$$^{\text{mo}}G_m = -RT \ln(\beta_m + 1) \cdot \left(\frac{\tau^{-5}}{10} + \frac{\tau^{-15}}{315} + \frac{\tau^{-25}}{1500} \right) / \left(\frac{518}{1125} + \frac{11692}{15975} \left(\frac{1}{p} - 1 \right) \right) \quad (2.100)$$

and for $\tau < 1$, we have

$$^{\text{mo}}G_m = RT \ln(\beta_m + 1) \left\{ 1 - \left[\frac{79\tau^{-1}}{140p} + \frac{474}{497} \left(\frac{1}{p} - 1 \right) \left(\frac{\tau^3}{6} + \frac{\tau^9}{135} + \frac{\tau^{15}}{600} \right) \right] \right\} / \left(\frac{518}{1125} + \frac{11692}{15975} \left(\frac{1}{p} - 1 \right) \right) \quad (2.101)$$

The parameter p is 0.28 for fcc metals and 0.40 for bcc metals.

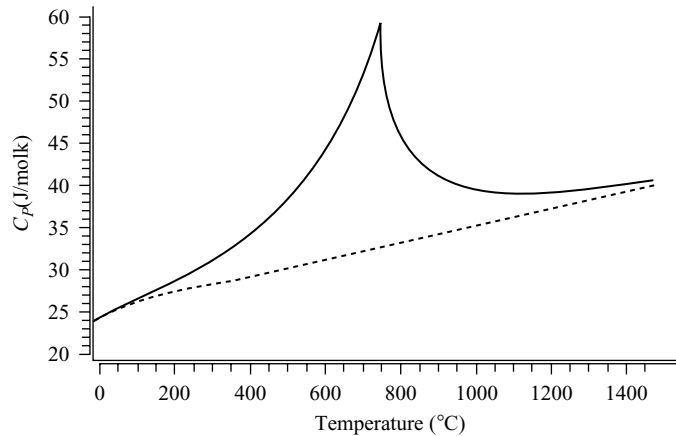


FIGURE 2-12 Molar specific heat capacity of pure iron. Calculated from computational thermodynamics and using the database of Section 2.2.8. Solid line: C_p including effect of ferromagnetic ordering. Dashed line: C_p without magnetic ordering.

A CALPHAD Thermodynamic Database for Fe–Cr–C

In recent times, a number of thermodynamic data have become available in the framework of the so-called CALPHAD (CALculation of PHase Diagrams) method. Some of these databases have been developed in large research projects such as the European initiative COST or the Scientific Group Thermodynamique Europe (SGTE) and within these projects, most of the thermodynamic parameters describing the Gibbs energy of particular alloy systems have been published in literature. They are thus (almost) freely available to the scientific community. Typically, one can find thermodynamic assessments of alloy systems in the *CALPHAD* journal or the *Journal of Phase Equilibria*. Many other alloy databases are available on a commercial basis.

When evaluating thermodynamic data to obtain Gibbs energies of solution phases or chemical potentials, it is advantageous to briefly review the basics of one of the standard database formats, here the format used by, for example, the thermodynamic software package ThermoCalc. More details are given in Saunders and Miodownik [SM98].

Each line of the database is delimited by an exclamation mark “!”. Multiple spaces and line breaks are treated as white spaces and are ignored. By convention, the dollar sign “\$” is used to indicate a comment. The entire text until the end of the line is then considered as a comment and ignored. Subsequently, we present a basic thermodynamic database for the Fe–Cr–C system, containing the solution phase BCC_A2 (bcc structure of A2 type) and the three carbide phases cementite (M_3C), $M_{23}C_6$, and M_7C_3 . The database lines are displayed in typewriter font.

```
$ Thermodynamic database for Fe-Cr-C
$ 2005-09-01

ELEMENT VA    VACUUM          0.0000E+00  0.0000E+00  0.0000E+00!
ELEMENT C     GRAPHITE         1.2011E+01  1.0540E+03  5.7400E+00!
ELEMENT CR    BCC_A2          5.1996E+01  4.0500E+03  2.3560E+01!
ELEMENT FE    BCC_A2          5.5847E+01  4.4890E+03  2.7280E+01!
```

The first few lines include a comment to the origin of the database and four lines that define the elements that are included in the database. The three numeric values represent the molar weight of the element in grams as well as the enthalpy and entropy at a temperature of 298 K and a pressure of 1 atm.

```
$ definition of some functions
FUNCTION GHSERCC  2.98150E+02  -17368.441+170.73*T-24.3*T*LN(T)
-4.723E-04*T**2+2562600*T**(-1)-2.643E+08*T**(-2)+1.2E+10*T**(-3);
6.00000E+03  N !
FUNCTION GHSERCR  2.98140E+02  -8856.94+157.48*T-26.908*T*LN(T)
+.00189435*T**2-1.47721E-06*T**3+139250*T**(-1); 2.18000E+03  Y
-34869.344+344.18*T-50*T*LN(T)-2.88526E+32*T**(-9);
6.00000E+03  N !
FUNCTION GHSERFE  2.98140E+02  +1225.7+124.134*T-23.5143*T*LN(T)
-.00439752*T**2-5.8927E-08*T**3+77359*T**(-1); 1.81100E+03  Y
-25383.581+299.31255*T-46*T*LN(T)+2.29603E+31*T**(-9);
6.00000E+03  N !
FUNCTION GFECERM  2.98150E+02  -10745+706.04*T-120.6*T*LN(T);
6.00000E+03  N !
FUNCTION GFEFCC  2.98140E+02  -1462.4+8.282*T-1.15*T*LN(T)+
6.4E-04*T**2+GHSERFE#; 1.81100E+03  Y
-27098.266+300.25256*T-46*T*LN(T)+2.78854E+31*T**(-9);
6.00000E+03  N !
```

```

FUNCTION GCRFCC      2.98150E+02  +7284+.163*T+GHSERCR#;
  6.00000E+03  N !
FUNCTION GCRM23C6   2.98150E+02  -521983+3622.24*T-620.965*T*LN(T)
  -.126431*T**2;   6.00000E+03  N !
FUNCTION GFEM23C6   2.98150E+02  +7.666667*GFECM#
  -1.666667*GHSERCC#+66920-40*T;   6.00000E+03  N !
FUNCTION GCRM7C3    2.98150E+02  -201690+1103.128*T
  -190.177*T*LN(T)-.0578207*T**2;   6.00000E+03  N !

```

Each of these functions starts with the keyword “FUNCTION” and its name. Functions are defined for convenience, if quantities are used multiple times, or for clearer structuring. In the preceding case they represent the pure element states as previously described in this Section. The function name is followed by the lower temperature limit for the next polynomial, which describes the quantity as a function of temperature and/or pressure. The polynomial ends with a semicolon “;”. Next is the upper temperature limit and an “N” if no further data follows or “Y” if data for another temperature interval is defined.

```

TYPE_DEFINITION & GES A_P_D BCC_A2 MAGNETIC -1.0      0.4 !

```

This line defines the magnetic properties of the bcc solution phase. The last number represents the parameter p from equations (2.100) and (2.101).

```

$ definition of the bcc phase
PHASE BCC_A2  %&  2 1  3 !
  CONSTITUENT BCC_A2  :CR%,FE% : C,VA% :  !

```

The keyword “PHASE” starts the phase definition. The phase name is followed by symbols, which have been defined by the keyword “TYPE_DEFINITION,” then the number of sublattices and the number of moles on these sublattices. The keyword “CONSTITUENT” defines the elements on the individual sublattices separated with a colon “:”. The percent sign “%” after an element indicates *major constituents*, which are elements that occur on this sublattice in significant amounts.

```

$ thermodynamic parameters of the bcc phase: unary parameters
PARAMETER G(BCC_A2,CR:VA;0)  2.98150E+02  +GHSERCR#;
  6.00000E+03  N !
PARAMETER G(BCC_A2,FE:VA;0)  2.98150E+02  +GHSERFE#;
  6.00000E+03  N !
PARAMETER G(BCC_A2,CR:C;0)  2.98150E+02  +GHSERCR#+3*GHSERCC#
  +416000;   6.00000E+03  N !
PARAMETER G(BCC_A2,FE:C;0)  2.98150E+02  +322050+75.667*T
  +GHSERFE#+3*GHSERCC#;   6.00000E+03  N !

$ thermodynamic parameters for magnetic ordering
$ TC: Curie temperature
$ BMAGN: Bohr magneton number
PARAMETER TC(BCC_A2,FE:C;0)  2.98150E+02  1043;
  6.00000E+03  N !
PARAMETER BMAGN(BCC_A2,FE:C;0)  2.98150E+02  2.22;
  6.00000E+03  N !
PARAMETER TC(BCC_A2,CR:C;0)  2.98150E+02  -311.5;
  6.00000E+03  N !

```

```

PARAMETER BMAGN(BCC_A2,CR:C;0) 2.98150E+02 -.008;
6.00000E+03 N !
PARAMETER TC(BCC_A2,CR:VA;0) 2.98150E+02 -311.5;
6.00000E+03 N !
PARAMETER BMAGN(BCC_A2,CR:VA;0) 2.98150E+02 -.01;
6.00000E+03 N !
PARAMETER TC(BCC_A2,FE:VA;0) 2.98150E+02 1043;
6.00000E+03 N !
PARAMETER BMAGN(BCC_A2,FE:VA;0) 2.98150E+02 2.22;
6.00000E+03 N !

$ interaction parameters
PARAMETER G(BCC_A2,CR,FE:C;0) 2.98150E+02 -1250000+667.7*T;
6.00000E+03 N !
PARAMETER BMAGN(BCC_A2,CR,FE:C;0) 2.98150E+02 -.85;
6.00000E+03 N !
PARAMETER TC(BCC_A2,CR,FE:C;0) 2.98150E+02 1650;
6.00000E+03 N !
PARAMETER TC(BCC_A2,CR,FE:C;1) 2.98150E+02 550;
6.00000E+03 N !
PARAMETER G(BCC_A2,CR:C,VA;0) 2.98150E+02 -190*T;
6.00000E+03 N !
PARAMETER G(BCC_A2,FE:C,VA;0) 2.98150E+02 -190*T;
6.00000E+03 N !
PARAMETER G(BCC_A2,CR,FE:VA;0) 2.98150E+02 +20500-9.68*T;
6.00000E+03 N !
PARAMETER BMAGN(BCC_A2,CR,FE:VA;0) 2.98150E+02 -.85;
6.00000E+03 N !
PARAMETER TC(BCC_A2,CR,FE:VA;0) 2.98150E+02 1650;
6.00000E+03 N !
PARAMETER TC(BCC_A2,CR,FE:VA;1) 2.98150E+02 550;
6.00000E+03 N !

```

The keyword “PARAMETER” is followed by the type of parameter (G, TC, or BMAGN), with the phase name and the elements. Again, sublattices are separated by a colon. After the semicolon comes the exponent k of the Redlich–Kister polynomials (see earlier in this Section). This parameter is only of relevance for interaction parameters.

Finally, the definition of the thermodynamic data for the carbide phases:

```

$ data for cementite
PHASE CEMENTITE % 2 3 1 !
CONSTITUENT CEMENTITE :CR,FE% : C : !

PARAMETER G(CEMENTITE,CR:C;0) 2.98150E+02 +3*GHSERCR#+GHSECC#
-48000-9.2888*T; 6.00000E+03 N !
PARAMETER G(CEMENTITE,FE:C;0) 2.98150E+02 +GFECM#;
6.00000E+03 N !
PARAMETER G(CEMENTITE,CR,FE:C;0) 2.98150E+02 +25278-17.5*T;
6.00000E+03 N !

PHASE M23C6 % 3 20 3 6 !
CONSTITUENT M23C6 :CR%,FE% : CR%,FE% : C : !

```

```

PARAMETER G (M23C6,CR:CR:C;0) 2.98150E+02 +GCRM23C6#;
6.00000E+03 N !
PARAMETER G (M23C6,FE:CR:C;0) 2.98150E+02 +.1304348*GCRM23C6#
+.8695652*GFEM23C6#; 6.00000E+03 N !
PARAMETER G (M23C6,FE:FE:C;0) 2.98150E+02 +GFEM23C6#;
6.00000E+03 N !
PARAMETER G (M23C6,CR:FE:C;0) 2.98150E+02 +.8695652*GCRM23C6#
+.1304348*GFEM23C6#; 6.00000E+03 N !
PARAMETER G (M23C6,CR,FE:CR:C;0) 2.98150E+02 -205342+141.6667*T;
6.00000E+03 N !
PARAMETER G (M23C6,CR,FE:FE:C;0) 2.98150E+02 -205342+141.6667*T;
6.00000E+03 N !

PHASE M7C3 \% 2 7 3 !
CONSTITUENT M7C3 :CR%,FE : C : !

PARAMETER G (M7C3,CR:C;0) 2.98150E+02 +GCRM7C3#;
6.00000E+03 N !
PARAMETER G (M7C3,FE:C;0) 2.98150E+02 +7*GHSEFE#+3*GHSECC#
+75000-48.2168*T; 6.00000E+03 N !
PARAMETER G (M7C3,CR,FE:C;0) 2.98150E+02 -4520-10*T;
6.00000E+03 N !

```

2.2.9 Practical Evaluation of Multicomponent Thermodynamic Equilibrium

Formulation of the Equilibrium Condition

In this section, the numerical algorithm for calculation of thermodynamic equilibrium based on the condition of minimum Gibbs free energy (compare Section 2.1.6) is elucidated. The algorithm is implemented in the present form, for example, in the thermodynamic engine of the thermodynamic/kinetic software package MatCalc (<http://matcalc.tugraz.at>). In evaluating thermodynamic equilibrium between l different phases, we first write down the general expression for the total Gibbs free energy G of the system with

$$G = NG_m = G = N \sum_l f^l G_m^l \quad (2.102)$$

where the subscript “m” denotes molar quantities. N is the number of moles of atoms in the system and f is the fraction of each of the participating phases. When limiting the further treatment to one mole of atoms, at a given temperature T and a given pressure P , we first assume that each phase is described by the mole fractions X_i^l of each of its constituents i . If the system is in thermodynamic equilibrium, each variation of any of the system variables, being either a phase fraction variable f^l or a composition variable X_i^l , will increase the total Gibbs free energy and lead to a less stable thermodynamic configuration. According to the discussion in the previous sections, the equilibrium criterion for the system can be manifested in terms of a minimum in its molar Gibbs free energy

$$G_m(T, P, f^l, X_i^l)|_{equ} = \sum_l f^l G_m^l(T, P, X_i^l) = \min \quad (2.103)$$

In the general case of phases with multiple sublattices, the mole fraction variables X_i^l are replaced by their corresponding site fraction variables ${}^l y_i^s$, where the index s is the

corresponding sublattice index. The relation between mole fraction and site fraction variables has already been presented in equation (2.96) and is repeated here for convenience:

$$X_i^l = \frac{\sum_s (b^s \cdot y_i^s)}{\sum_s (b^s \cdot (1 - y_{\text{Va}}^s))} \quad (2.104)$$

b^s are stoichiometry factors describing the number of sites that are available on each sublattice s . y_{Va}^s are the site fractions of vacant interstitial sublattice sites.

In view of equation (2.104), it is apparent that a unique phase constitution with respect to the sublattice model will require knowledge of all site fractions y_i^s rather than only the mole fractions X_i^l . Consequently, the mathematical statement representing thermodynamic equilibrium in the frame of the sublattice model finally reads

$$\sum_l f^l G_m^l(T, P, y_i^s) = \min \quad (2.105)$$

Implicit Constraints

An important issue that must not be ignored in solving equation (2.105) is the fact that various boundary conditions are implicitly given by the mathematical formalism of the thermodynamic model. In case of the sublattice model, valid solutions must satisfy the mass balance equation for each species i

$$\sum_l X_i^l \cdot f^l = X_i^0 \quad (2.106)$$

where X_i^0 is the total system mole fraction, the global conservation of phase fractions

$$\sum_l f^l = 1 \quad (2.107)$$

and, with particular regard to the sublattice model, the site fraction balances

$$\sum_i y_i^s = 1 \quad (2.108)$$

Finally, all system variables y_i^s and f^l must range between 0 and 1, which can be expressed by the following inequalities:

$$\begin{aligned} 0 &\leq y_i^s \leq 1 \\ 0 &\leq f^s \leq 1 \end{aligned} \quad (2.109)$$

Mole Fraction and Site Fraction Constraints

The system constraints (2.106)–(2.109) are mandatory and need to be satisfied in all cases, that is, both in unconstrained and compositionally constrained thermodynamic equilibrium. The constraints that are discussed subsequently apply to equilibria with additional compositional restrictions, namely, restrictions that control some or all of the composition variables within a particular phase. Formally, compositional constraints can be introduced in various ways.

The following generic variant simply limits the mole fraction of an element j in a phase l to a constant value c_j^l with

$$X_j^l - c_j^l = 0 \quad (2.110)$$

On the other hand, one can also consider the ratios among the constrained elements rather than fixing the individual mole fractions. Accordingly, an alternative type of composition variable u_j^l can be introduced with

$$u_j^l = \frac{X_j^l}{\sum_k X_k^l} \quad (2.111)$$

Normally, the summation k is performed over the substitutional sublattices. However, in a more general context, the so-called u -fraction variable should be defined in terms of constrained and unconstrained phase components rather than interstitial and substitutional sublattices. Thus, for application to compositionally constrained equilibrium analysis, the summation is defined to include all system components, which are restricted by a compositional constraint. This more general definition allows for a convenient representation of the mutual relations among the constrained (less mobile) elements, independent of the amount of unconstrained (fully mobile) components. The u -fraction constraint is set as

$$u_j^l - d_j^l = 0 \quad (2.112)$$

where d_j^l is an arbitrary constant.

For numerical equilibrium analysis, it is important to recognize that the two constraints (2.110) and (2.112) represent completely different types of restrictions, which apply to different practical situations. Figure 2-13 presents the differences between X - and u -fraction constraints graphically. As an example, consider the arbitrary ternary system A–B–C, where a component B is virtually immobile and therefore subject to a compositional constraint. Component A is treated as the dependent species and its amount is given by $X_A = 1 - X_B - X_C$. The left side of Figure 2-13 represents the constraint $X_B = \text{const.}$, thus corresponding to a situation where the components A and C are assumed to be fully mobile, whereas component B maintains constant mole fraction. In contrast, the constraint $u_B = \text{const.}$ (right side) denotes that only component C is considered to be mobile and the ratio between the amounts of components A and B is the same all along the line representing the constraint.

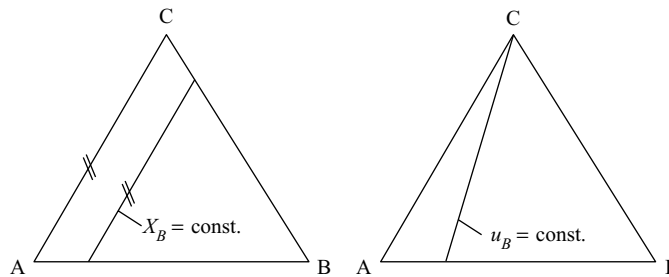


FIGURE 2-13 Interpretation of X - and u -fraction constraints in a ternary system A–B–C (schematic). Any composition, which satisfies the corresponding constraint, is located on either of the two dashed lines.

Numerical Minimization

A solution to the minimum Gibbs free energy expression (2.105) that simultaneously satisfies the mandatory constraints (2.106)–(2.109) and, optionally, additional conditions (2.110) or (2.112), can be obtained by application of the Lagrange multiplier method. By that means, the problem of finding the constrained minimum of a multivariable function is transformed into the problem of finding the solution to a set of coupled nonlinear equations. This procedure will now be demonstrated.

As a first step, a dummy variable v_j is formally introduced, representing both the phase fraction variables f^l and the site fraction variables $^l y_j^s$. Then, the equality constraints (2.106)–(2.108) are rearranged into a form $g(v_j) = 0$ and, analogously, all compositional constraints (2.110) and (2.112) are rewritten to yield $p(v_j) = 0$. The inequalities (2.109) are converted into expressions $h(v_j) = 0$. A functional F can then be defined with

$$F(v_j, \lambda_n, \eta_m, \tau_k) = G_m(v_j) + \sum_n \lambda_n g_n(v_j) + \sum_m \eta_m h_m(v_j) + \sum_k \tau_k p_k(v_k) \quad (2.113)$$

The parameters λ_n , η_m , and τ_k are Lagrange multipliers, and the functions $g_n(v_j)$, $h_m(v_j)$, and $p(v_j)$ represent the mandatory and optional constraints introduced before. The index n counts from 1 to the total number of equality constraints, m counts to the number of currently active inequalities, and k denotes the number of optional compositional X - and u -fraction constraints.

The Lagrange multiplier transformation is accomplished by requiring that in the true minimum of the functional $F(v_j, \lambda_n, \eta_m, \tau_k)$, all of its derivatives with respect to the system variables v_j and the Lagrange multipliers λ_n , η_m , and τ_k have to be equal to zero, that is,

$$\frac{\partial F(v_j)}{\partial v_j} = 0, \frac{\partial F(v_j)}{\partial \lambda_n} = 0, \frac{\partial F(v_j)}{\partial \eta_m} = 0, \frac{\partial F(v_j)}{\partial \tau_k} = 0 \quad (2.114)$$

The transformation yields a system of coupled nonlinear equations (2.114), which can be solved by standard methods. In many practical cases, an iterative procedure based on the Newton–Raphson algorithm will be the method of choice.

Finally, it is noted that if the solution of the constrained minimization problem satisfies all imposed boundary conditions, that is, $g_n(v_j) = 0$, $h_m(v_j) = 0$ and $p_k(v_j) = 0$, the multipliers λ_n , η_m , and τ_k can, in principle, take any numerical value (except zero!) without influencing the final solution. A global or local minimum of the functional $F(v_j, \lambda_n, \eta_m, \tau_k)$, therefore, simultaneously represents the global or local minimum of the total Gibbs free energy $G_m(v_j)$.

Expressions for the Derivatives of $F(v_j, \lambda_n, \eta_m, \tau_k)$

Before explicitly writing down the expressions for the derivatives of the functional F , it is advantageous to rewrite $F(v_j, \lambda_n, \eta_m, \tau_k)$ first, by separating the different types of system variables and Lagrange multipliers. From equation (2.113) and after separation, we obtain

$$\begin{aligned}
F(f^l, {}^l y_j^s, \lambda'_{n'}, \lambda''_{n''}, \lambda'''_{n'''}, \eta'_{m'}, \eta''_{m''}, \tau'_{k'}, \tau''_{k''}) = & \sum_l f^l G_m^l(v_j) + \\
& \sum_{n'} \lambda'_{n'} \left(\sum_l f^l X_{n'}^l - X_{n'}^0 \right) + \lambda''_{n''} \left(\sum_l f^l - 1 \right) + \\
& \sum_{n'''} \lambda'''_{n'''} \left(\sum_j {}^l y_j^s - 1 \right) + \\
& \sum_{m'} \eta'_{m'} v_{m'} + \sum_{m''} \eta''_{m''} (1 - v_{m''}) + \\
& \sum_{k'} \tau'_{k'} \left(X_{k'}^l - c_{k'}^l \right) + \sum_{k''} \tau''_{k''} \left(u_{k''}^l - d_{k''}^l \right)
\end{aligned} \tag{2.115}$$

The index l counts from 1 to the total number of active phases. n' counts the number of independent components in the system. As there is only one phase fraction balance in the system, n'' is always one. This index will, for that reason, be omitted in the following expressions. n''' denotes the accumulated number of sublattices in all active phases. The index j refers to the number of constituents on a particular sublattice and s designates the number of sublattices in a particular phase. The index m' denotes the number of active inequalities from the condition $v_j \geq 0$ and m'' from $v_j \leq 1$ [equations (2.109)]. The indices k' and k'' , finally, refer to the number of X - and u -fraction constraints in the system, written in terms of equation (2.110) and (2.112), respectively.

In the following, expressions are given for the partial derivatives of F [equation (2.116)] with respect to each type of system variable and Lagrange multiplier. The mandatory inequality constraints [equations (2.109)] are omitted subsequently for the benefit of clarity.

- **Phase Fraction Variable f^l** : one equation for each active phase l :

$$\frac{\partial F}{\partial f^l} = G_m^l(v_j) + \sum_{n'} \left(\lambda'_{n'} X_{n'}^l \right) + \lambda'' = 0 \tag{2.116}$$

- **Site Fraction Variable ${}^l y_j^s$** : one equation for each site fraction variable ${}^l y_j^s$ on each sublattice s in each active phase l :

$$\begin{aligned}
\frac{\partial F}{\partial ({}^l y_j^s)} = & f^l \cdot \frac{\partial G_m^l(v_j)}{\partial ({}^l y_j^s)} + \sum_{n'} \lambda'_{n'} \cdot \left(f^l \sum_l \frac{\partial X_{n'}^l}{\partial ({}^l y_j^s)} \right) + \\
& \lambda'''_{n'''} =_r + \sum_{k'} \tau'_{k'} \cdot \left(\frac{\partial X_{k'}^l}{\partial ({}^l y_j^s)} \right) + \sum_{k''} \tau''_{k''} \cdot \left(\frac{\partial u_{k''}^l}{\partial ({}^l y_j^s)} \right) = 0
\end{aligned} \tag{2.117}$$

The index r denotes the index of the particular site fraction balance that the variable ${}^l y_j^s$ is part of.

- **Lagrange Multipliers for Element Mass Balance $\lambda'_{n'}$** : one equation for each independent system component n' .

$$\frac{\partial F}{\partial \lambda'_{n'}} = \sum_l f^l X_{n'}^l - X_{n'}^0 = 0 \tag{2.118}$$

The derivatives with respect to the mass balance multipliers n^l are identical to the original mass balance statements in equation (2.107).

- **Lagrange Multiplier for Phase Fraction Balance λ''** : there is only one equation in the system.

$$\frac{\partial F}{\partial \lambda''} = \sum_l f^l - 1 = 0 \quad (2.119)$$

This derivative is identical to the phase fraction balance statement in equation (2.106).

- **Lagrange Multiplier for Site Fraction Balance $\lambda_{n'''}''$** : one equation for each sublattice s in each active phase l .

$$\frac{\partial F}{\partial \lambda_{n'''}''} = \sum_j^l y_j^s - 1 = 0 \quad (2.120)$$

The summation is performed over the site fraction variables on the particular sublattice the index n''' refers to. These derivatives are identical to the original site fraction balance statement in equation (2.108).

- **Lagrange Multipliers for X - and u -Fraction Constraints $\tau_{k'}^l$ and $\tau_{k''}^l$** : one equation for each constraint in each phase l .

$$\frac{\partial F}{\partial \tau_{k'}^l} = X_{k'}^l - c_{k'}^l = 0 \quad (2.121)$$

$$\frac{\partial F}{\partial \tau_{k''}^l} = u_{k''}^l - d_{k''}^l = 0 \quad (2.122)$$

These derivatives are identical to the original constraint statements in equations (2.110) and (2.112).

Equations (2.116)–(2.122) define the system of equations, which has to be solved simultaneously to evaluate thermodynamic equilibrium at given boundary conditions.

Bibliography

-
- [Cal85] H. B. Callen. *Thermodynamics and an Introduction to Thermostatistics*. Wiley, New York, 1985.
- [CZD+03] S.-L. Chen, F. Zhang, S. Daniel, F.-Y. Xie, X.-Y. Yan, Y. A. Chang, R. Schmid-Fetzer, and W. A. Oates. Calculating phase diagrams using PANDAT and PanEngine. *J. Metals*, 12:48–51, 2003.
- [DDC+89] R. H. Davies, A. T. Dinsdale, T. G. Chart, T. I. Barry, and M. H. Rand. Use of thermodynamic software in process modelling and new applications of thermodynamic calculations. *High Temp. Science*, 26:251–262, 1989.
- [EH90] G. Eriksson and K. Hack. ChemSage—a computer program for the calculation of complex chemical equilibria. *Met. Trans. B*, 21:1013–1023, 1990.
- [FR76] G. Falk and W. Ruppel. *Energie und Entropie*. Springer-Verlag, Berlin, 1976.
- [Gib61] J. W. Gibbs. *On the Equilibrium of Heterogeneous Substances (1876)*, pp. 55–349. Dover, New York, 1961.
- [Hac96] K. Hack, editor. *The SGTE Casebook—Thermodynamics at Work*. The Institute of Materials, London, 1996.
-

- [Hil98] M. Hillert. *Phase Equilibria, Phase Diagrams and Phase Transformations—Their Thermodynamic Basis*. Cambridge University Press, Cambridge, 1998.
- [Ind76] G. Inden. Computer calculation of the free energy contributions due to chemical and/or magnetic ordering. In *Proc. CALPHAD V, Max Planck Institut für Eisenforschung*, Düsseldorf, pp. 1–13, 1976.
- [KEH+00] U. R. Kattner, G. Eriksson, I. Hahn, R. Schmid-Fetzer, B. Sundman, V. Swamy, A. Kussmaul, P. J. Spencer, T. J. Anderson, T. G. Chart, A. Costa e Silva, B. Jansson, B. J. Lee, and M. Schalin. Use of thermodynamic software in process modelling and new applications of thermodynamic calculations. *CALPHAD*, 24:55–94, 2000.
- [MA96] S. Malanowski and A. Anderko. *Modelling Phase Equilibria*. Wiley, Canada, 1996.
- [PTBE89] A. D. Pelton, W. T. Thompson, Ch. W. Bale, and G. Eriksson. FACT thermochemical databases for calculations in materials chemistry at high temperatures. *High Temp. Science*, 26:231–250, 1989.
- [SJA85] B. Sundman, B. Jansson, and J.-O. Andersson. The Thermo-Calc databank system. *CALPHAD*, 9(2): 153–190, 1985.
- [SM98] N. Saunders and A. P. Miodownik. *CALPHAD (Calculation of Phase Diagrams): A Comprehensive Guide*. Pergamon Materials Series, Pergamon, Oxford, 1998.
- [Wag52] C. Wagner. *Thermodynamics of Alloys*. Addison-Wesley Press, Reading, MA, 1952.

3 Monte Carlo Potts Model

—Mark Miodownik

3.1 Introduction

A soap bubble is an extraordinarily beautiful thing and yet it requires virtually no skill to produce. This is because surface tension does all the work for you, making sure that a perfect spherical liquid membrane is produced every time. In fact it is impossible to blow imperfect bubbles. Even if you try to blow the bubble through a noncircular orifice you may at first achieve a temporary nonequilibrium shape (see Figure 3-1), but the end result is always a perfect sphere.

In this chapter we will show how such beautiful but tyrannical surface tension effects can be investigated using the computational equivalent of soap bubble solution: the Potts model. Like soap bubble solution it is easy to use and provides fundamental insights into surface tension phenomena; but also like soap bubble solution, it can lead to a sticky mess. The aim of this chapter is to provide a best practice implementation guide to the model. It is divided into four sections. The first deals with the Ising model, which is considerably simpler than the Potts model but will allow us to discuss the physics of boundary motion that is encapsulated in the Potts model. It also allows the reader to get some experience of coding of simple Potts-like algorithms. In the second section, we introduce the Q -state Potts model and show the diverse range of phenomena that can be simulated. Section 3 is an algorithm section, allowing the reader



FIGURE 3-1 *A soap bubble in an intermediate state, before becoming a perfect sphere.*

to improve the efficiency of the Potts code should they wish. The final section introduces the reader to a range of industrial applications of the Potts model.

3.2 Two-State Potts Model (Ising Model)

In this section we consider the case when a microstructure can be defined as consisting of two degenerate spin states. Traditionally this is called an Ising model and has been used to study the dynamics of magnetic spin systems. The model has also been used to study the microstructural evolution of grain boundaries, since the two states can code for the two crystal orientations either side of a single grain boundary, and so be used to simulate the behavior of bicrystals, such as that shown in Figure 3-2. The Ising model can capture not just the basic characteristics of a boundary, such as its interface free energy, but also the subtleties of the second derivative of energy with respect to boundary orientation and thus be used to investigate the impact of the Herring equation on curvature driven growth. Also since the system is extremely simple, consisting of only two states, it provides a clear introduction to the more complicated Q -state Potts model.

3.2.1 Hamiltonians

The Ising model describes an ideal two component system in which space is typically discretized into a regular array of lattice sites. The state of the system is described in terms of the set of components of the system, called spins, which are associated with each lattice site, $s_i \in \{0, 1\}$, where i labels the lattice site. The system defines a boundary between unlike spins and no boundary between like spins in the following way:

$$\gamma(s_i, s_j) = \begin{cases} 0 & \text{for } s_i = s_j \\ \frac{J}{2} & \text{for } s_i \neq s_j \end{cases} \quad (3.1)$$

where i represents a site and j its neighbor, and $J(> 0)$ is an interfacial energy constant of the system. Thus the energy of the system can be written as a sum over the spatial distribution of the spins as

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j) \quad (3.2)$$

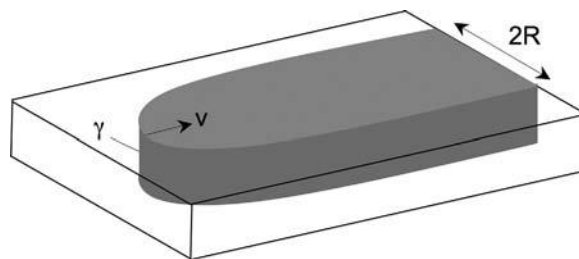


FIGURE 3-2 *The experimental setup of bicrystal experiments using the grim-reaper geometry. This experimental system yields boundaries that move at constant velocities and so can be used to measure grain boundary mobility.*

where N is the total number of lattice sites in the system, and z is the maximum number of neighbors. The definition of neighbor spin depends on the dimension and coordination of the lattice, which we will cover in Section 3.2.3.

The Hamiltonian may also include a volume energy term, $H (> 0)$, which lowers the energy of one type of spin relative to the other. This energy is associated with an external or volume driving force and results in a modified Hamiltonian

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j) - H \sum_{i=1}^N s_i \quad (3.3)$$

where the H term represents a sum over spins $s_i = 1$.

In terms of magnetic domains, the Hamiltonian describes a ferromagnetic system in which the perfectly ordered state has zero energy and H represents an external magnetic field. Under the microstructural paradigm considered here, J scales with the interfacial energy of system in which the single crystal has zero energy; all polycrystalline states have positive energy which scale with the total boundary area. H , as we will see later, may represent an external field but more frequently is used to deal with stored energies that arise in the case of deformed structures and so provides a driving force for recrystallization.

3.2.2 Dynamics (Probability Transition Functions)

The distribution of the spins in the lattice represents the state of the system. We can use a Monte Carlo method to sample different states. The method is extremely simple in principle: choose a site at random, propose a change in spin, calculate the change in energy ΔE associated with that spin swap, and accept or reject the change based on ΔE . This amounts to a biased navigation through state space. There are two main methods for performing such dynamics. The first is Glauber dynamics where the spins are unconserved. A lattice site is chosen at random, and a new spin proposed for the site. The new Hamiltonian is computed and the change is accepted or rejected depending on a probability transition function $P(\Delta E)$. The second method deals with situations where the volume fraction of each spin type is conserved; it is called Kawasaki dynamics. Here a lattice site is chosen, a neighboring site is chosen, and then a swap of the spins is proposed. Again the ΔE is computed and the change is accepted or rejected depending on $P(\Delta E)$.

Both the spin-flip dynamics, Glauber and Kawasaki, require the definition of the probability transition function. There are two common choices, the Metropolis function

$$P(\Delta E) = \begin{cases} 1 & \text{if } \Delta E \leq 0 \\ \exp \frac{-\Delta E}{kT_s} & \text{if } \Delta E > 0 \end{cases} \quad (3.4)$$

and the symmetric function

$$P(\Delta E) = \frac{1}{2} \left\{ 1 - \tanh \frac{\Delta E}{2kT_s} \right\} \quad (3.5)$$

where kT_s , defines a thermal energy of the simulation; it is analogous to the thermal energy of experimental systems but not directly related. The choice of the probability function has no effect on the thermodynamics of the system, although the choice of the functional form of $P(\Delta E)$ does affect the dynamics of boundary motion slightly. The basic algorithm to determine whether a spin change is accepted or not using the Metropolis scheme is shown in Function 3-1.

FUNCTION 3-1: Basic Glauber Dynamics Using Metropolis Probability Distribution Algorithm

$spin_{old}$ = Existing spin of site
 $spin_{new}$ = New proposed spin of site
 $likes_{old}$ = Number of like neighbor sites with spin = $spin_{old}$
 $likes_{new}$ = Number of like neighbor sites with spin = $spin_{new}$
 $\Delta E = likes_{old} - likes_{new}$
IF $\Delta E \leq 0$ **then**
 $spin_{new}$ is accepted
else if $T > 0$ **then**
 $probability = \exp^{-\Delta E/kT}$
 random = A random number uniformly distributed between 0, 1.
 if $random < probability$ **then**
 $spin_{new}$ is accepted
 end if
end if

The time variable in the Ising model is not associated with any equation of motion, nor is it associated with a fundamental dynamic frequency of the system being simulated. The time required to attempt a single spin flip whether successful or unsuccessful is defined arbitrarily as τ . On average it takes $N\tau$ to visit each site on the simulation lattice once: this is defined as one Monte Carlo time step, 1 MCS.

3.2.3 Lattice Type

There are two common types of lattice used in 2D simulations: hexagonal or square lattices. With each of these lattices an individual spin may be defined to have a number of different nearest neighbors as shown in Figure 3-3. In a simple square lattice, a site may be defined to have only the four first nearest neighbors, labeled 1–4 on Figure 3-3(a), but more often the eight first and second nearest neighbors, labeled 1–8 on Figure 3-3(a), are used. In the triangular lattice the six first nearest neighbors shown in Figure 3-3(b) are sufficient. In three dimensions the simple cubic lattice is commonly used with the 26 first, second, and third nearest neighbors used as shown in Figure 3-3(c).

Ideally the type of lattice should have no physical significance to the system being simulated and should not influence the thermodynamics or dynamics. As we shall see later, this is not always the case. The choice of neighbor coordination strongly affects the type of equilibrium boundary shapes that are favored in the system. Thus in 2D square lattices, boundaries with planes 0° , 45° , and 90° are strongly favored, and can cause facets to form. These low energy facets obviously have a corollary with the faceting of atomic planes, and although they can be studied by similar techniques, they are problematic when using the model to simulate isotropic boundary shapes since they impose an anisotropy into the simulations. In the extreme case these affects can totally invalidate the results of the simulations.

There a number of ways of mitigating against these lattice effects. First, some lattices and neighbor coordinations have less intrinsic anisotropy. For instance the triangular lattice in two dimensions using six 1st neighbors has the lowest anisotropy of any 2D regular lattice, and the simple cubic lattice using 26 nearest neighbors is the most effective in three dimensions. Another method to mitigate against the unwanted influence of the lattice is not have one at all, that is, use a random lattice [Jan03].

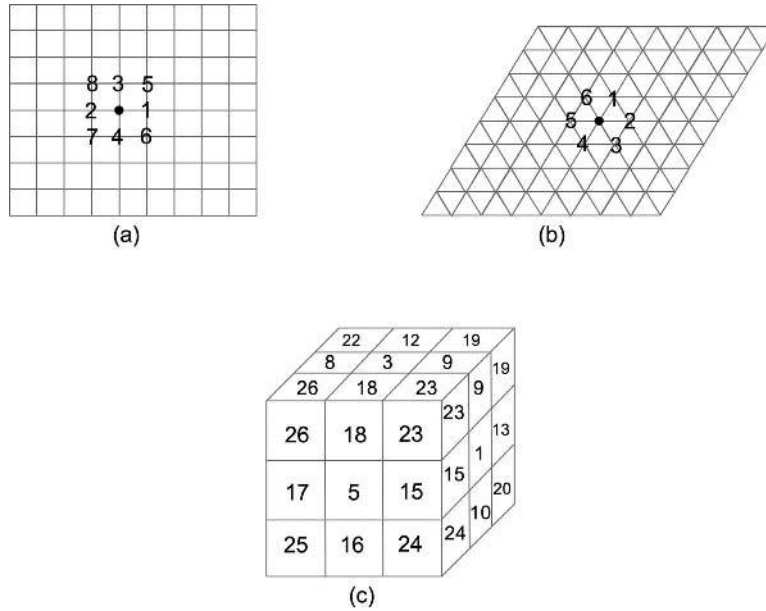


FIGURE 3-3 Different types of lattice and the neighbor coordination used in the Ising model, (a) 2D square lattice, (b) 2D triangular lattice, (c) 3D simple cubic lattice.

3.2.4 Boundary Conditions

As can be appreciated from the previous section, there exists a special set of lattice sites which lie at the boundaries of the simulation area/volume. These sites need to be treated differently than the internal sites of the lattice. In the case where the boundaries represent the edge of the simulated system, these sites will have fewer nearest neighbors. They may also have a solid–liquid or a solid–gas surface energy associated with them. Alternatively, or in addition, they may have a certain concentration of solute imposed on them as a boundary condition, and thus act as a sink or source for solute or dislocations or other variable quantities, as in Figure 3-4(a).

Another type of commonly used boundary condition is that of mirror boundary conditions. In this case the sites on a boundary are “mirrored” so that the neighbor shell of the boundary site is comprised of those defined by a reflection transformation, as in Figure 3-4(b). For example, a 2D square lattice a mirror boundary condition applied at the $x = 0$ boundary simulates the effect that the spins for all sites $x < 0$ exactly mirror those for $x > 0$. In practice this only requires the boundary nearest neighbors to be mirrored. For example, for a 2D square lattice, a mirror boundary condition applied at $x = 0$ means that the boundary neighbors shell will be $\{3, 4, 5, 1, 6, 5, 1, 6\}$.

Perhaps the most popular type of boundary condition is a periodic boundary condition. In this case the edges of the simulation effectively wrap around and contact the opposing edges creating a toroidal simulation area in the 2D case. These boundary conditions are relatively simple to impose: it simply involves assigning the nearest neighbors of each edge site to the opposite boundary edge, as in Figure 3-4(c). The boundary conditions for the simulation volume are usually implicitly encoded in the function that returns the neighbor sites of each site. For

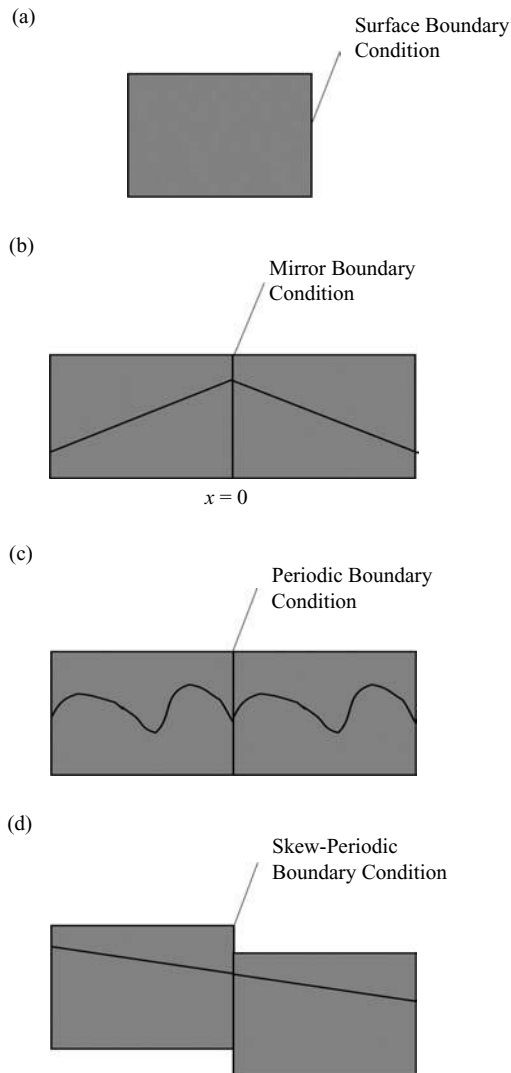


FIGURE 3-4 Different types of boundary condition used in the Ising model, (a) surface boundary condition, (b) mirror boundary condition, (c) periodic boundary condition, (d) skew-periodic boundary condition.

instance the eight neighbors of sites in a 2D lattice of size $m \times n$ may be coded as shown in Function 3-2.

FUNCTION 3-2: Calculate Neighbor Sites While Imposing Periodic Boundary Conditions

```

get coordinates  $(x, y)$  of site
neighbor1 = mod( $x + 1, m$ ),  $y$ 
neighbor2 = mod( $x + m - 1, m$ ),  $y$ 

```

$$\begin{aligned} \text{neighbor3} &= x, \text{mod}(y + n - 1, n) \\ \text{neighbor4} &= x, \text{mod}(y + 1, n) \\ \text{neighbor5} &= \text{mod}(x + 1, m), \text{mod}(y + n - 1, n) \\ \text{neighbor6} &= \text{mod}(x + 1, m), \text{mod}(y + 1, n) \\ \text{neighbor7} &= \text{mod}(x + m - 1, m), \text{mod}(y + 1, n) \\ \text{neighbor8} &= \text{mod}(x + m - 1, m), \text{mod}(y + n - 1, n) \end{aligned}$$

A variant of the periodic boundary condition is the skew-periodic boundary condition. In this case the edges of the simulation wrap around but at the boundary a vector displacement parallel to the boundary is imposed, as in Figure 3-4(d). This type of boundary condition is used when simulating flat boundaries that have a nonperpendicular intersection angle with a simulation area/volume boundary.

3.2.5 The Vanilla Algorithm

We have covered how the energetics, dynamics, lattice type, and boundaries conditions are relevant to simulating systems that contain boundaries that move by curvature growth. Now we are ready to use the model to perform some basic simulations of these systems.

The algorithm for the Ising model is relatively straightforward. It involves the setting up of a 1D, 2D, or 3D array of sites, each of which can be in one or two spin states, and is shown schematically for conserved spin Kawasaki dynamics in Figure 3-5. The array of sites as a function of time is the output of the model. It can be captured graphically as a “snapshot” at each time step and output to the screen in real time using OpenGL or other graphical libraries. Such graphical analysis is fine as a debugging tool, but slows up the code, hence it is often best to save snapshots periodically for more detailed postsimulation analysis (for more on this see Section 3.4). Before going further the reader is encouraged to get a feel for the model by attempting Problems 3-1–3-3.

PROBLEM 3-1: Minimal Surfaces

Code a 2D Ising model, of size 50×50 lattice sites, with a simple cubic lattice using eight nearest neighbor coordination, Kawasaki spin dynamics, and $kT_s = 0$. Insert a grain of ellipsoidal shape in the middle of the simulation area as the initial configuration. Save snapshots of the spin configurations every 10 MCS. Use an imaging tool to visualize these snapshots and to make a movie of the simulation. Experiment with changing the initial shape of the domain. Is the final circular shape in any way determined by the initial shape?

PROBLEM 3-2: Shrinking Circle

Code a 2D Ising model, of size 50×50 lattice sites, with a simple cubic lattice using eight nearest neighbor coordination, Glauber (Metropolis) spin dynamics, and $kT_s = 0$. Insert a circular grain of size $R = 20$ in the middle of the simulation area as the initial configuration. Save snapshots of the spin configurations every 10 MCS. Use an imaging tool to visualize these snapshots and to make a movie of the simulation. Why does the circular grain shrink?

PROBLEM 3-3: The Effect of Lattice Geometry

Code a 2D Ising model, of size 50×50 lattice sites, with a triangular lattice using six nearest neighbor coordination, Glauber spin dynamics, and $kT_s = 0$. Insert a circular grain of size $R = 20$ in the middle of the simulation area as the initial configuration. Save snapshots of the spin configurations every 10 MCS. Use an imaging tool to visualize these snapshots and make a movie of the simulation. Compare the result with simulation performed in Problem 3-2.

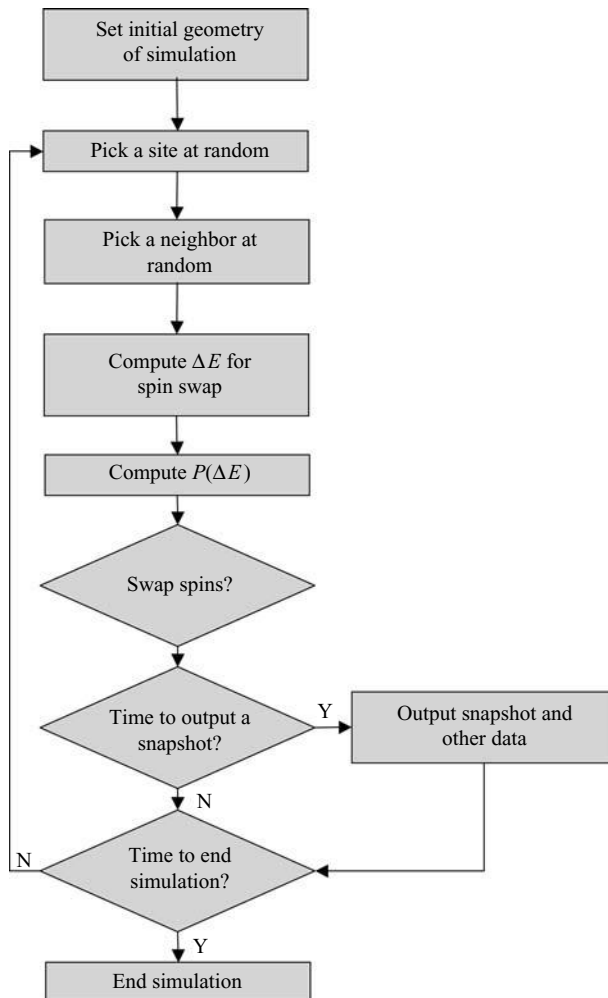


FIGURE 3-5 *The basic (vanilla) algorithm for the nonconserved spin Ising model using Kawasaki dynamics.*

The algorithm for nonconserved spin dynamics (Glauber dynamics) is shown in Figure 3-6. These algorithms require several decisions to choose the site and the new spin which are fundamentally random. It should be emphasized that for large lattices and for large run times, this requires the generation of a large number of random numbers. If the source of these random numbers has a low repeat signature or does not distribute the random number uniform on $\{0, 1\}$, then artificial patterns of behavior which are not due to curvature driving forces will be observed.

3.2.6 Motion by Curvature

Figure 3-7 shows snapshots of a 3D Ising model simulation of a spherical domain. The simulation was performed on a simple cubic lattice, periodic boundary conditions, Glauber (Metropolis) spin dynamics, and $kT_s = 0$. As we have seen in the previous section, the

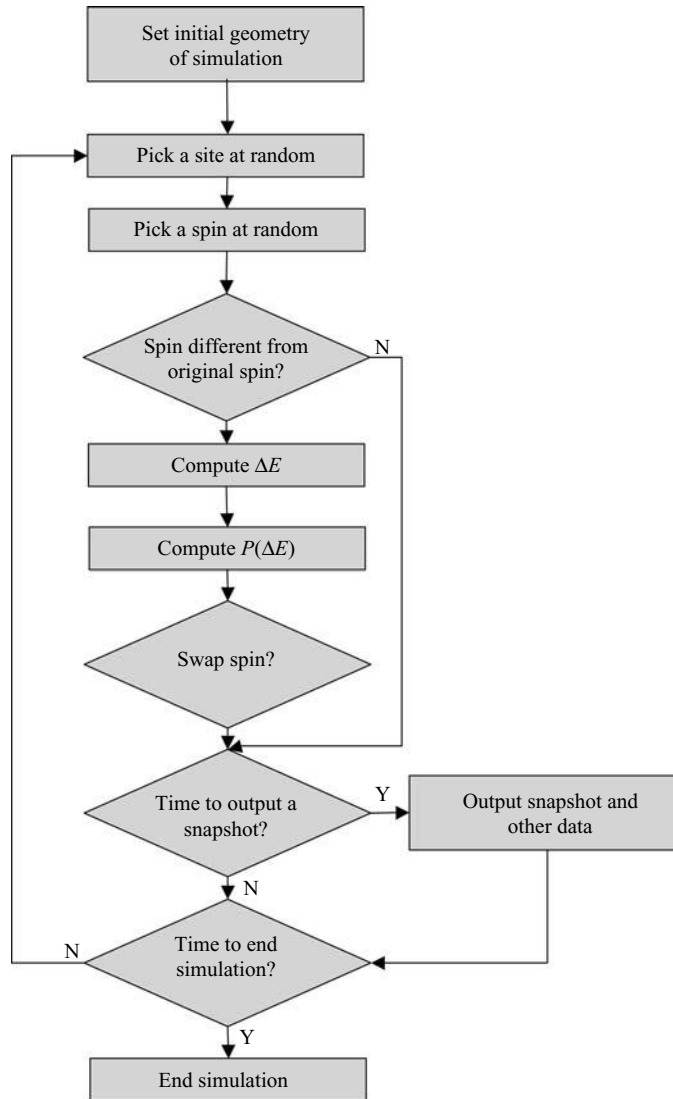


FIGURE 3-6 *The basic (vanilla) algorithm for the conserved spin Ising model using Glauber dynamics.*

initial configuration of a spherical domain is not stable in such a model, since the energy of the system can be reduced by the sphere shrinking, reducing the boundary area.

The sphere does not shrink as a perfect sphere, since the system is both discrete and stochastic. But since the sphere is shrinking under a capillary force, the kinetics of such a system should conform to those derived using rate theory of Burke and Turnbull [BT52], which assumes that the law of motion governing the curvature-driven motion is

$$v = M\gamma\kappa \quad (3.6)$$

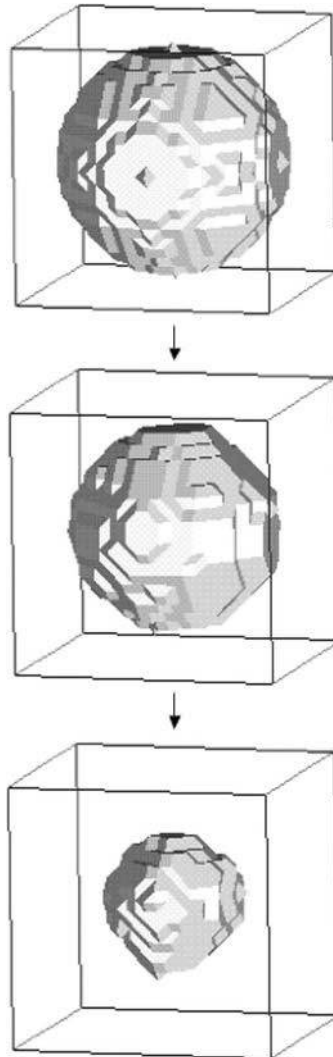


FIGURE 3-7 Sphere shrinking in an Ising model with unconserved spins, a square lattice, periodic boundary conditions, Glauber spin dynamics, and $kT_s = 0$. The net number of active sites remains constant even though the area and boundary length decrease.

where v is the velocity of the boundary, M is the mobility of the boundary, γ is the boundary energy, and κ is the boundary curvature. For a shrinking sphere of radius R , $v = dR/dt$ and $\kappa = 1/R$ and so the Burke and Turnbull growth law is

$$\frac{dR}{dt} = -\frac{M\gamma}{R} \quad (3.7)$$

By integrating equation (3.7) we arrive at the kinetic equation describing the radius of the sphere:

$$R_o^2 - R^2 = \gamma Mt \quad (3.8)$$

In three dimensions the volume of a sphere is, $V = 4/3\pi R^3$, so equation (3.8) becomes:

$$\frac{V^{2/3}}{V_o^{2/3}} = 1 - \frac{\alpha_v t}{V_o^{2/3}} \quad (3.9)$$

where $\alpha_v = (4/3\pi)^{2/3}\gamma M$. Figure 3-8 shows the plot of $V^{2/3}/V_o^{2/3}$ versus $t/V_o^{2/3}$ for the sphere. The linearity of the plots confirm that equation (3.9) is obeyed. The plot shows a departure from linearity as the domain becomes small; this is to be expected as the domain loses its spherical shape due to the discreteness of the lattice. The result is not dependant on the type of lattice, so the same result is obtained in three dimensions for a sphere shrinking on a simple cubic, fcc and/or bcc lattice.

PROBLEM 3-4: Boundary Kinetics

Derive an equivalent expression to equation (3.9) for the area of a shrinking circular grain. Validate your 2D shrinking circle code developed in Problems 3-2 and 3-3, by investigating whether their boundary kinetics obeys equation (3.6).

PROBLEM 3-5: The Effect of Non-zero Simulation Temperatures

Use your 2D shrinking circle code developed in Problems 3-2 and 3-3 to investigate the effect of non-zero values of kT_s .

3.2.7 The Dynamics of Kinks and Ledges

Although these simple cases provide evidence that the Ising model is simulating motion by curvature, it's not obvious how the model encapsulates the physics of curvature correctly. We shall consider this next.

Consider the shrinking circular grains simulated in Problems 3-2 and 3-3. In each Monte Carlo step, all sites are sampled once on average and in that time they can only do one of two

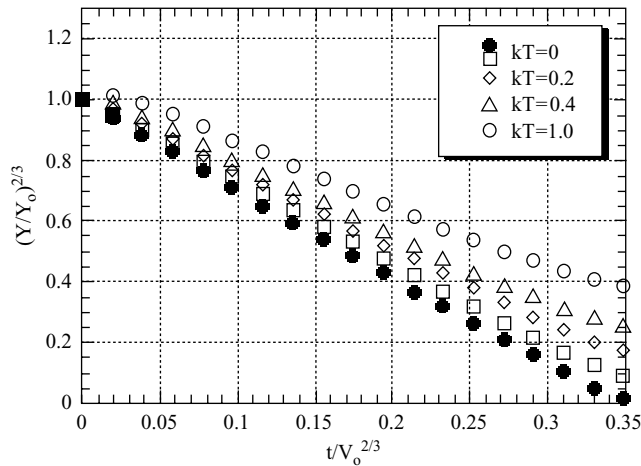


FIGURE 3-8 A plot of $V^{2/3}/V_o^{2/3}$ versus $t/V_o^{2/3}$ for the sphere shrinking with different temperatures.

things: they can swap to the opposite spin type, or remain as their own spin type. When $kT_s = 0$ these options are reduced further because most sites in the system have eight nearest neighbors of the same spin, and so for these sites any swap would be associated with $\Delta E > 0$. When $kT_s = 0$, these sites cannot swap spins. Only sites for which a swap to the opposite spin is associated with $\Delta E < 0$ have the possibility of switching spin. We can identify the sites able to change orientation if picked. These are shown in Figure 3-9, differentiated with a black dot. These sites are the only sites which are capable of being reoriented. In other words, these sites, if selected and the opposing spin chosen, would have a probability $p(\Delta E) = 1$ of changing spin. We shall call these the *active sites* of the system. All other sites, whether on the boundary or not, have a probability $p(\Delta E) = 0$ of changing spin.

Although the total number of active sites changes as the domain shrinks, the net number (the number of active sites with spin = 0 minus the number of active sites with spin = 1), A_c , remains constant ($A_c = 8$, see Figure 3-9). This means we can calculate exactly the area change per MCS. In one MCS, all the sites are chosen on average once, and therefore all active sites are chosen once. There is a probability of $p = 1/2$ that when picked, the opposite spin will be selected, and so on average the net change in area will be equal to half the net number of active sites, $dA = A_c/2$. The kinetics then follow:

$$\frac{dA}{dt} = -\frac{A_c}{2} = -4 \quad (3.10)$$

Thus we expect the linear decrease of area with time as shown in the simulation and in agreement with theory.

We can analyze the shrinking of a half-loop in a similar manner, which gives an insight into the ledge mechanisms of curvature growth in these systems. For a simulation of a half-loop, see Figure 3-10, carried out on a square lattice. Using the Metropolis probability function we can do active site analysis and notice that the number of active sites for this geometry is also constant, but half that of the sphere, $A_c = 4$. The area swept out will be equal to $dA = A_c/2$ units/MCS. If the width of the half-loop is $2R$, then the velocity of the boundary will be

$$\frac{dx}{dt} = -\frac{A_c}{4R} \quad (3.11)$$

This shows that the boundary velocity as expected is proportional to the $1/R$. The mechanism by which the migrations occur can be seen intuitively as the injection and motion of kinks from the side of the half-loop. These travel in a biased motion across to the center of the half-loop where they are annihilated by their opposite kind. The reason why A_c is on average constant can be readily appreciated by inspecting Figure 3-10, which shows that because of the geometry most of the kinks have a probability of $p = 0.5$ of traveling in either direction. It is only where the ledges are nucleated and annihilated that there is a bias.

PROBLEM 3-6: Constant Driving Force Boundary Motion

Code a 2D Potts model, of size 50×50 lattice sites, with a simple cubic lattice using eight nearest neighbor coordination, Glauber spin dynamics, and $kT_s = 0$. Use surface boundary conditions and insert a circular half-loop grain of size $R = 10$ as the initial configuration. Save snapshots of the spin configurations every 10 MCS. Use an imaging tool to visualize these snapshots and to make a movie of the simulation. Postprocess the data to plot boundary velocity versus time.

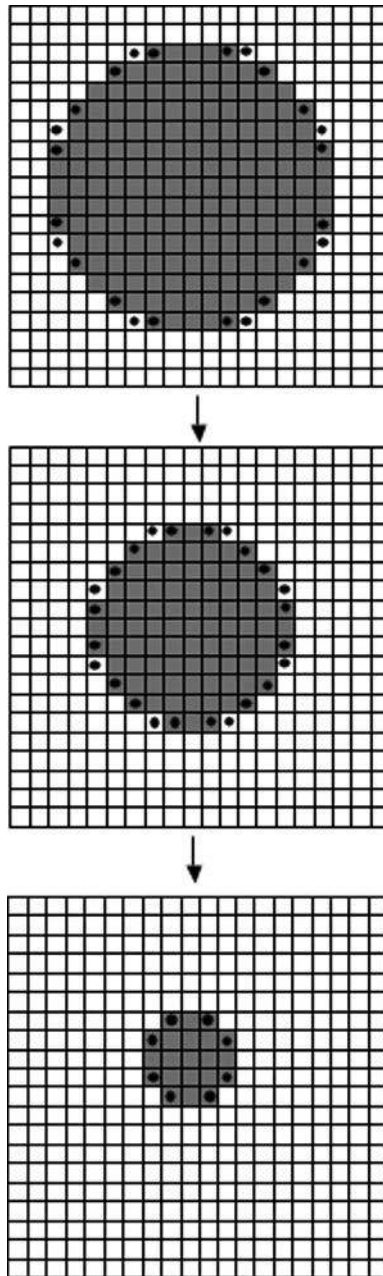


FIGURE 3-9 Circle shrinking in an Ising model with unconserved spins, a square lattice, periodic boundary conditions, Glauber spin dynamics, and $kT_s = 0$. The net number of active sites remains constant even though the area and boundary length decrease.

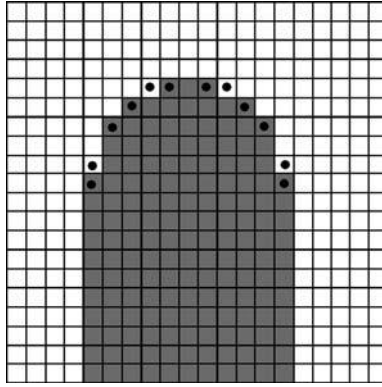


FIGURE 3-10 Half-loop, grim-reaper geometry, in an Ising model with unconserved spins, a square lattice, periodic boundary conditions, Glauber spin dynamics, and $kT_s = 0$. The net number of active sites remains constant and so the loop moves with a constant velocity.

PROBLEM 3-7: Effect of Driving Force

Use the code developed in Problem 3-6 to plot boundary velocity versus R . How can we use the plot to calculate boundary mobility?

Thus it can be appreciated that motion by curvature occurs through kink dynamics in two dimensions and ledge motion in three dimensions. A flat boundary, as shown in Figure 3-11(a), has no curvature and no kinks and so at $kT_s = 0$ will be unable to move. If a double kink is introduced, there is a possibility that if the kinks move away from each other, the boundary will be forward by one lattice site. However, since the kinks move in an unbiased random fashion, there is an equal probability that the kinks will recombine. Curved boundaries may be seen as a series of kink double kinks which have biased motion, and it is this that causes motion by curvature.

In the general case of an isolated boundary segment, see Figure 3-12. The net curvature of this boundary, θ_{net} , is the angle between the initial and final tangent vectors swept out by the left-hand rule. On a square lattice θ_{net} can only assume discrete values, $\theta_{net} = b\pi/2$, where b is an integer. Once again we can perform the active site analysis and show that there are two active sites for each $\pi/2$ so that:

$$\frac{dA}{dt} = -b \tag{3.12}$$

where the negative value of the area swept out indicates the movement of the boundary toward its center of net curvature. This is an exact discretization of the continuum law for boundary motion by mean curvature and illustrates that the boundary evolution of boundaries in the Ising model is consistent with the physics of boundary motion of equation (3.6).

PROBLEM 3-8: Triangular Lattice Active Site Analysis

Derive equation (3.10) for a 2D triangular lattice.

PROBLEM 3-9: Roughening of Boundaries

What is the impact of non-zero values of kT_s on the active site population of a boundary, and thus on boundary kinetics in the Ising model?

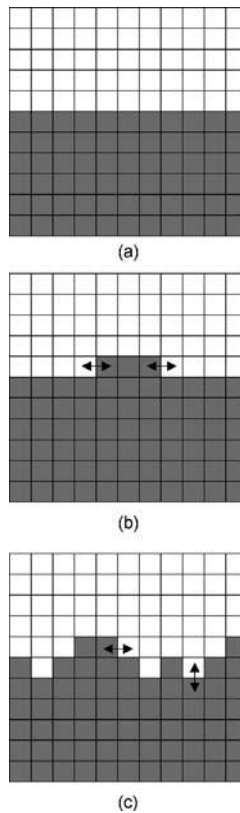


FIGURE 3-11 Flat boundary in an Ising model with unconserved spins, a square lattice, periodic boundary conditions, Glauber spin dynamics, and (a) $kT_s = 0$, (b) $kT_s = 0.2$, and (c) $kT_s = 0.8$.

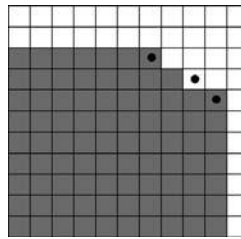


FIGURE 3-12 General case of a isolated boundary segment, in an Ising model with unconserved spins, a square lattice.

3.2.8 Temperature

The faceting effects of the previous section are temperature dependant, and the discussion referred to a low temperature regime. In order to understand the effect of temperature it is useful to again consider the shrinking circle experiment. Figure 3-8 shows the plot of $V^{2/3}/V_o^{2/3}$ versus $t/V_o^{2/3}$ for the sphere at various values of kT_s . The linearity of the plots confirm that equation (3.9) is obeyed, but for high values of kT_s there is nonuniformity at the beginning of

the plots; this is due to the size of the domain initially increasing, corresponding to an increase in boundary enthalpy. This can only occur because there is an increase in the boundary entropy associated with boundary roughness which more than compensates for the increase in enthalpy. Thus despite the increase in boundary area, the Gibbs free energy of the system is reduced.

It should be noted that in the Ising model simulations carried out at finite temperature, there is a finite probability that any site in the whole system can swap its spin. There is roughening temperature, T_r , where the system becomes disordered (often called the Curie temperature by those using the Ising model to study magnetism). Figure 3-13(a) shows a shrinking sphere at $kT_s = 0$; the shape is compact and defined by discrete ledges and facets. Figure 3-13(b) shows a shrinking sphere at $kT_s = kT_r$, the roughening temperature. The system becomes increasingly disordered. The exact roughening temperature associated with the phase transition to the disordered state depends on the lattice type and neighbors. It is possible to prevent such disordering while still obtaining high temperatures by confining swap attempts to boundary sites. When this is done the effect of roughening the boundary can be investigated independently from the system roughening, as in Figure 3-13(c).

Swaps that occur on the boundary do increase the number of active sites, but not the net curvature of the boundary, which is determined by topological considerations of the boundary. The result is that temperature affects the kinetics of the boundary but not the energetics of curvature driven growth. In effect temperature injects roughness into the boundary, by supplying kinks, thus *reducing the anisotropy of boundary mobility and energy in the system*. This has the effect of making the simulations slower as we shall see in the next sections. Figure 3-11 shows a linear boundary at different temperatures, which shows the effects of systematically increasing the temperature.

PROBLEM 3-10: Effect of Entropy

Modify your 2D shrinking circle code to disallow spin swaps away from the boundary for non-zero values of kT_s . Implement periodic boundary conditions to overcome the issues associated with the lattice boundaries. Compare the kinetics of the shrinking circular grains in this model with those measured in Problem 3-5.

3.2.9 Boundary Anisotropy

Ising models are performed on lattices, and it seems obvious that the boundary energies and boundary mobilities will have inherent anisotropies that depend on the type of the lattice. For instance there is an energy anisotropy of boundary plane which can be expressed most conveniently through a Wulff plot. The 2D triangular lattice has a lower anisotropy than the 2D square lattice, with $\gamma_{[10]}/\gamma_{[11]} = 1.07$ at $kT_s = 0.2$ [MSG02]. The presence of such anisotropy brings into question whether equation (3.6) can be used as an accurate description of the system, and perhaps it should be replaced by the more accurate Herring relation [Her49]:

$$v = M(\gamma + \gamma'')\kappa \quad (3.13)$$

where γ'' is the second derivative of the interface free energy with respect to interface inclination, and the term $\gamma + \gamma''$ is referred to as the interface stiffness. Furthermore mobility is also an anisotropic function of boundary plane. For instance in the 2D square lattice, kink motion occurs easily along the [10] directions but not along [11] directions, for example, $M_{[10]}/M_{[11]} \approx 25$ at $kT_s = 0.2$ [MSG02].

Given this inherent anisotropy of the system it seems extremely odd that shrinking circles or spheres show no obvious faceting, nor is it shown in the migration of boundaries in general.

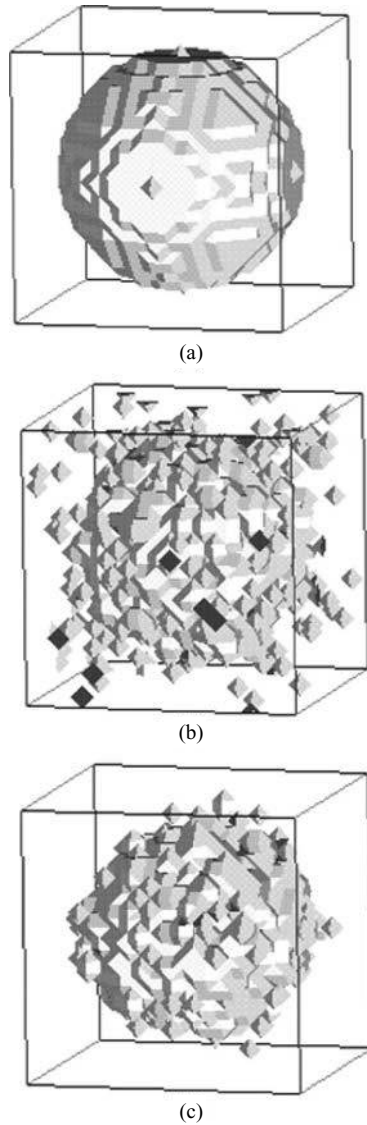


FIGURE 3-13 Snapshot of sphere shrinking under curvature on a 3D simple cubic lattice (a) $kT_s = 0$, (b) $kT_s = kT_r$, (c) $kT_s = kT_r$ disallowing grain nucleation.

The explanation is that the anisotropy of the interface stiffness and the mobility compensate for each other, producing almost isotropic reduced mobility, $M^* = M(\gamma + \gamma'')$. This result, however, only applies to the case where the driving force is due to curvature driven growth. When the driving force includes an external field, such as in equation (3.3), where H is non-zero, the reduced mobility is no longer isotropic and indeed the shrinking circles are faceted, see Figure 3-14. This result is explained by the fact that interface facets are nearly immobile at low temperatures, due to the rarity of kink nucleation, while interfaces with a high density of geometrically necessary kinks are highly mobile. As a result the grain shape reflects the

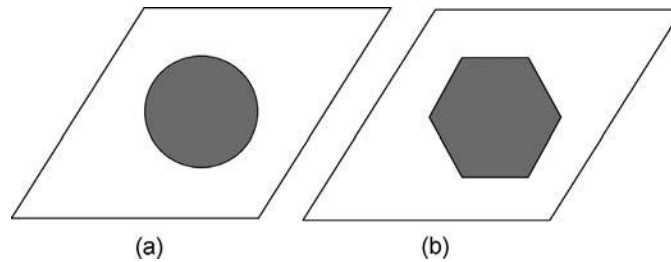


FIGURE 3-14 Snapshot of circle shrinking under curvature on a 2D triangular lattice, (a) $H = 0$, a curvature driving force only leads to isotropic shape, (b) $H = 0.2$, external and curvature driving force leads to highly faceted shape.

underlying anisotropy of the lattice. The reason why the anisotropy of the interface mobility and the interface stiffness cancel out in the case of curvature driven growth is that the entropic part of the stiffness is large for inclinations where the mobility is low and vice versa [MSG02]. What this shows is that boundary mobility is a function of the type of driving force as also observed in experimental systems [WGS02].

PROBLEM 3-11: The Effect of a Volume Driving Force

Investigate the effect of non-zero values of H on the evolution of a shrinking circular domain in the 2D Potts model using a square lattice.

3.2.10 Summary

This concludes the investigation of the Ising model. The model is simple and yet encapsulates a great deal of complex physics. Using the model provides an insight into motion by curvature without having to worry about the complex topological issues associated with multidomain systems. By the end of this section the reader should have experience with coding their own Ising models, using different lattices, using different boundary conditions, and visualizing and analyzing simple systems; a fundamental understanding of why the Potts model encapsulates the physics of boundary motion by curvature; an appreciation of lattice effects; and an appreciation of the role of simulation temperature.

3.3 Q-State Potts Model

Soap froths, such as that shown in Figure 3-15(a), are easy to make because again, surface tension does all the work for you. Take a soap bubble, add another soap bubble, and they are immediately attracted to each other because by being together they minimize their surface energy. Add a few more bubbles and you notice something else; they actually rearrange their interfaces to create a minimum internal surface area. The characteristic Y junctions where three bubbles meet have perfect 120° angles. You never observe four bubbles meeting with 90° angles. It does not matter how big or small the bubbles are, you always get these Y junctions. This behavior is a direct result of the minimization of isotropic interface energy. Thus soap froths are not just a jumble of bubbles; they have form, and the form is dictated by surface energy considerations.

The Potts model simulates the effects of interface energy on the topology of the boundary networks, and so it is a tool to investigate self-ordering behavior. Figure 3-16 shows the evolution of such a system simulated using the Potts model. Instead of bubbles we have domains

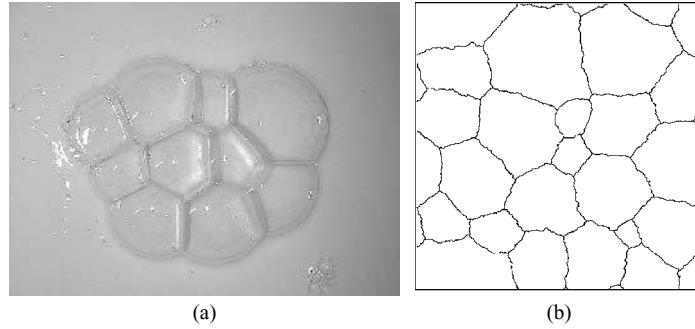


FIGURE 3-15 (a) A soap froth—the structure is self-ordering, (b) a Potts model 2D simulation of a soap froth.

and instead of liquid membranes we have domain boundaries. In this simulation the domain boundaries are associated with an isotropic excess energy, which has a profound influence on the network topology because it implies that in 2D the number of boundaries impinging on a vertex is always equal to three. In other words, only triple points with 120° vertex angles are stable. Fourfold and higher vertices, if formed, will always dissociate into the relevant number of triple points.

The Q -state Potts model is almost identical to the Ising model, except that there are more than two states; there are Q states in fact. The boundaries between these states can be treated as isotropic, thus allowing the evolution of soap froths or grain structures to be modeled. Alternatively they can be anisotropic, allowing anisotropic grain growth and abnormal grain growth to be simulated. Particles can be incorporated allowing the modeling of Zener pinning. Stored energies can also be incorporated allowing recrystallization to be modeled. The model can also be coupled with finite element models to allow general thermomechanical simulations to be carried out. The aim of this section is to not get embroiled in the details of how to apply the Potts model to such real applications; we will address this issue in Section 3.5. Rather, in this section the reader is encouraged to play, to explore the model by changing the local physics one variable at a time, and through this exploration get a feel for the self-organizing behavior that the model exhibits.

3.3.1 Uniform Energies and Mobilities

The Potts model is a generalization of the Ising model. The state of the system is described in terms of the set of Q spins, which are associated with each lattice site, $s_i \in \{0, Q\}$, where i labels the lattice site. The system defines a boundary between unlike spins and no interface between like spins. In the isotropic case the energy associated with this boundary is described by an energy function γ :

$$\gamma(s_i, s_j) = \begin{cases} 0 & \text{for } s_i = s_j \\ \frac{J}{2} & \text{for } s_i \neq s_j \end{cases} \quad (3.14)$$

Thus as in the Ising model, the energy of the system can be written as a sum over the spatial distribution of the spins as

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j) \quad (3.15)$$

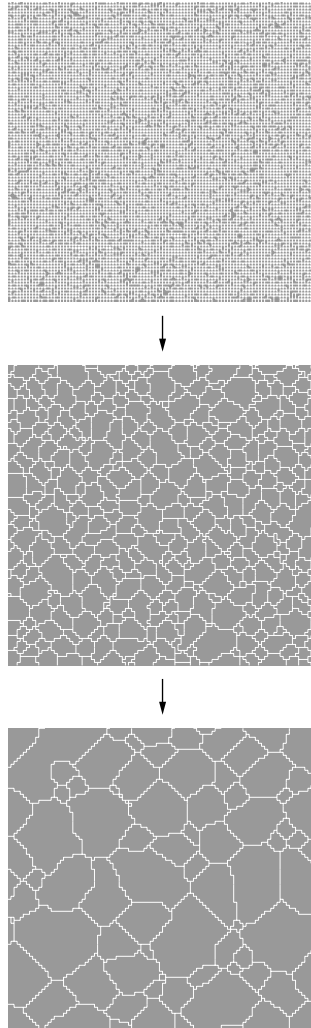


FIGURE 3-16 Microstructural evolution of an initially random distribution of spins on a 2D square lattice using the Potts model on a simple 2D square lattice, periodic boundary conditions, Metropolis spin dynamics, and $kT_s = 0$. The initial configuration of spins was set by allocating each lattice a random spin $s_i \in \{0, Q\}$.

A Monte Carlo method is used to sample different states: choosing a random change to a spin (to one of the Q other states) at a random lattice site and accepting or rejecting the change based on the change to the total energy of the system, ΔE , computed via the Hamiltonian in equation (3.15). Glauber or Kawasaki dynamics can be employed using the Metropolis or symmetric probability functions as described in Section 3.2.2. As in the Ising model, the time required to attempt a single spin flip whether successful or unsuccessful is defined arbitrarily as τ and so 1 MCS is defined as N attempted flips. The same lattice types (e.g., 2D square, 2D hexagonal, 3D simple cubic) and boundary conditions (e.g., periodic, symmetric) can be used. In fact in the case where the energy and mobility of the boundaries is isotropic, the only change

in the model is the switch from two states to Q states; thus Figure 3-6 is the basic algorithm for a vanilla Potts model using Glauber dynamics.

Nevertheless, as Figure 3-16 shows, there is an important difference in the observed behavior of the model. Note how, despite an initial random allocation of Q spins, the system self-organizes in a cellular pattern which coarsens in a self-similar manner. This is an important result, and reasons for such self-organization will be considered in the next subsection, for it concerns the competition between the minimization of interfacial energy with the need for space filling. But before going on to consider why the model works, it is important to get a feel for the model, and thus the reader is encouraged to attempt Problems 3-12 and 3-13.

PROBLEM 3-12: Cellular Systems

Write a basic Potts model code with a 2D triangular lattice, Glauber dynamics using a Metropolis probability function. Set the initial geometry of the simulation by allocating each lattice site a random spin between 1 and Q . Use periodic boundary conditions. Save snapshots of the spin configurations every 100 MCS. Use an imaging tool to visualize these snapshots and to make a movie of the simulation. Show that you get a familiar Ising system when $Q = 2$, and that as Q increases the system transitions to a self-ordering cellular system, the structure of which is independent of Q .

PROBLEM 3-13: Lattice Effects

Write a Potts model code with a 2D square lattice, Glauber dynamics using a Metropolis probability function, and periodic boundary conditions. Set the initial geometry of the simulation by allocating each lattice site a random spin, where $Q = 50$. Show that when $kT_s = 0$, you get a self-organizing network which has straight boundaries at 45° angles to the orthogonal lattice. Run a series of simulations for $kT_s = 0.25, 0.5, 0.75, 1.0$. Observe how the boundaries become rougher and less straight as temperature increases.

3.3.2 Self-Ordering Behavior

In 2D networks formed through the action of the minimization of isotropic surface energy, the average number of boundaries per grain is six. Therefore the only stable network is a hexagonal array of grains, where each grain has six neighbors and the 120° vertex angles at the triple points can be satisfied by straight boundaries. These boundaries having no curvature have no net force acting on them and so remain static. Any networks that deviate from this regular array inevitably contain some grains with less than six sides and some with more than six sides. If the triple points maintain their 120° angles then the array must contain curved boundaries. Curvature driven migration given by equation (3.6) then causes the system to evolve, as shown in Figure 3-16. The boundaries of grains with less than six sides are concave (curved toward the center of a grain), and so boundary migration makes these grains shrink. Grains with more than six sides have convex boundaries and so these grains grow. In other words, the competing requirements of space filling and surface tension cause large grains to grow and small grains to shrink. This forms the basis of a remarkable law proposed by von Neumann [vN52], which states that the growth rate of a 2D cell with area, A , and N_s sides is given by

$$\frac{dA}{dt} = c(N_s - 6) \tag{3.16}$$

where c is a constant. This result has been shown to be correct for both 2D soap froths and 2D grain structures [Mul56] and has more recently been generalized to 3D.

PROBLEM 3-14: Self-Ordering Behavior

Perform 2D Potts model simulations with $Q = 100$, and plot dA/dt versus N_s . Show that Potts model simulations obey equation (3.16) despite the fact that no topological rules are employed. Perform the simulations for lower values of Q and show that at small values of Q equation (3.16) breaks down. Why is this? Examine the movies of your simulations to discover a clue.

Employing large Q has an obvious side effect from an algorithmic point of view; it makes the simulations slower, since the probability of success of a given spin flip attempt is proportional to $1/Q$. There are alternative algorithms other than the vanilla algorithm (Figure 3-6), which allow the modeling of high Q systems without this loss of efficiency. These algorithms do not change the physics and self-ordering characteristics of the Potts model; they simply speed it up, allowing larger systems to be modeled for a given CPU time. These algorithms are discussed in Section 3.4 and the reader is encouraged to explore the implementation of these algorithms if they find that the vanilla algorithm is becoming too slow.

3.3.3 Boundary Energy

So far we have treated boundaries as if they are continuous isotropic interfaces. We can relax this condition by assuming that the energy of each grain boundary is anisotropic. Although such a system has little relevance to a soap bubble froth, it is extremely important for modeling crystalline microstructures and biological cellular structures, since in almost all cases these systems possess anisotropic interfaces.

Let us assume that γ is a function of the crystallographic misorientation across the boundary. This changes the equilibrium condition at the nodes (where three or more boundaries meet). If these are to remain in a state of local equilibrium and maintain the equilibrium angles defined by the boundary energies, then neglecting torque forces, the angles at the nodes in two dimensions are given by the relation:

$$\frac{\gamma_1}{\sin \phi_1} = \frac{\gamma_2}{\sin \phi_2} = \frac{\gamma_3}{\sin \phi_3} \quad (3.17)$$

where γ_i are boundary energies and ϕ_i the angles at the triple point as illustrated in Figure 3-17(a). What this means in practice is that triple points are no longer thermodynamically constrained to be 120° . Not just that, but triple points in two dimensions, and quadrijunctions points in three dimensions, are no longer the only stable node configurations. This makes possible a vast array of different boundary network morphologies.

Figure 3-17(b) shows how a node is represented in the Potts model on a square lattice. Note that the triple point angles are discrete quantities, which depend not just on boundary energies but also on the type of lattice. Although it may seem that there must be a problem in implementing anisotropic energies since at any one time the node angles can only be either 90° or 180° in the square lattice and 120° or 30° in the triangular lattice, in fact this type of lattice effect is not too problematic, since the effective angles are integrated over time during simulations and so do allow a wide range of node angles to be simulated. More in depth discussion of lattice effects will be dealt with in Section 3.5.

Implementing anisotropic energies into the Potts model is relatively straightforward since it only requires the modification of Hamiltonian. The easiest way to code these systems is to introduce a new identifier, which along with the spin is associated with each lattice site. Thus we may give each lattice site both a spin identifier, s_i , and component identifier, η_i , as shown in Figure 3-17(c). The component identifier carries information necessary to calculate the

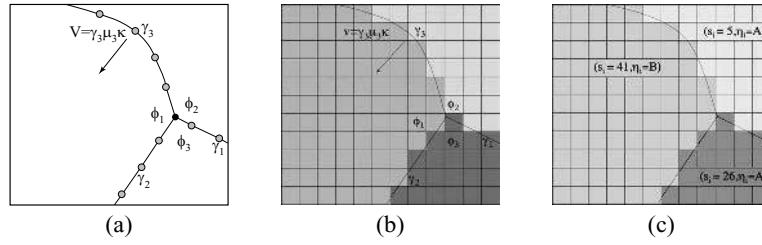


FIGURE 3-17 The relationship between boundary energy and node angle, (a) a continuum system, (b) Monte Carlo Potts model: each grain orientation is represented by a different gray scale, the boundaries are sharp being implicitly defined between sites of different orientations, (c) showing the implementation of components and spins into the model.

anisotropic nature of the boundary, while the spin identifier continues to be used to calculate whether a particular site is on a boundary. In such a system the Hamiltonian becomes:

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j, \eta_i, \eta_j) \quad (3.18)$$

This Hamiltonian represents only the boundary energy of the system and so implicitly assumes that bulk free energy of each component is the same, but their boundary energy is different. The simplest system we can consider is a single phase polycrystal with only two types of component, A and B, which then results in a system with three types of boundary, A–A, B–B, and A–B boundaries, with three boundary energies, J_{AA} , J_{BB} , and J_{AB} which gives:

$$\gamma(s_i, s_j, \eta_i, \eta_j) = \begin{cases} 0 & \text{in the grain interiors } (s_i = s_j, \eta_i = \eta_j) \\ \frac{J_{AA}}{2} & \text{for A–A boundaries } (s_i \neq s_j, \eta_i = \eta_j = A) \\ \frac{J_{BB}}{2} & \text{for B–B boundaries } (s_i \neq s_j, \eta_i = \eta_j = B) \\ \frac{J_{AB}}{2} & \text{for A–B boundaries } (s_i \neq s_j, \eta_i \neq \eta_j) \end{cases} \quad (3.19)$$

This is the simplest anisotropic system, but already it is getting complicated. The behavior of the system can be most easily understood by considering the the dimensionless parameters $R_A = J_{AA}/J_{AB}$ and $R_B = J_{BB}/J_{AB}$.

Clearly when $R_A = R_B = 1$ the system is isotropic. When $R_A = R_B > 1$ the relative energy of the A–B boundaries decreases in relation to the A–A and B–B boundaries; thus during evolution the system will try to minimize the area or length of A–A and B–B boundaries in favor of A–B boundaries and so minimize the energy of the system. Figure 3-18 shows such evolution of such a 2D system. Notice how it self-organizes into a mosaic structure which minimizes the length of A–A and B–B boundaries. The mosaic structure is itself then able to coarsen in a self-similar manner.

FUNCTION 3-3: Calculate Anisotropic Energy Dynamics Using Metropolis Scheme

$spin_{old}$ = existing spin of site
 $spin_{new}$ = new proposed spin of site
 η_{old} = existing component of site
 η_{new} = new component of site

Continued

```

energyold = sum of bond energies over neighbor shell with spin = spinold and
component = ηold
energynew = number of like neighbor sites with spin = spinnew and component = ηnew
ΔE = energyold - energynew
IF ΔE ≤ 0 then
    spinnew and ηnew is accepted
else if T > 0 then
    probability = exp-ΔE/kTs
    random = A random number uniformly distributed between 0, 1.
    if random < probability then
        spinnew and ηnew is accepted
    end if
end if

```

The basic algorithm to determine whether a spin change is accepted or not using anisotropic energy in the Metropolis scheme is shown in Function 3-3. Note that this implementation slows the algorithm up quite a bit, and that speed-up can be achieved by a more sophisticated algorithm described in Section 3.4.

PROBLEM 3-15: Anisotropic Energy

Write a Potts model code to simulate the grain growth of a two component system. Write the code in such a way that the system undergoes an initial period of isotropic growth for a time, t_{init} , to establish an equiaxed grain structure with fractions f_A and f_B of the two components. For $t > t_{init}$ allocate energies J_{AA} , J_{BB} , J_{AB} to the boundaries. Experiment with varying the ratios R_A and R_B and the fraction f_A . Note how the most interesting effects occur when $f_A = f_B = 0.5$.

Figure 3-19 shows the equilibrium structures formed under a variety of other conditions. If $R_A = R_B > 1$ the A–A and B–B boundaries are favored over A–B boundaries, and the system self-orders the phases to segregate the A and B components and thus minimize boundary energy. Figure 3-19(b) shows such a structure which orders the A component and the B component into separate enclaves and can be contrasted with Figure 3-19(a), which shows the random distribution of A and B components which comes in the isotropic case when $R_A = R_B = 1$. Figure 3-19(c) shows what happens when $R_A > R_B$; the system gets rid of the high energy A–A boundaries altogether. Figure 3-19(d) shows another example of the type of mosaic structures that are formed when $R_A = R_B < 1.0$. Figure 3-19(e) shows another example of $R_A > R_B$ but this time where $\gamma_{BB} = \gamma_{AB}$, here the A component grains are not removed because it is only the A–A boundaries which are high energy; however, they do become an isolated component. Figure 3-19(f) shows the effect of using $kT_s = 0$, with anisotropic energies. Because of the high lattice pinning present, the structure shows a very high degree of planarity in the low-boundary planes which are at 45° to the simulation lattice.

Note that in many of these 2D anisotropic energy systems, four grain junctions (quadrijunctions), are possible. The angles at which boundaries meet in a quadrijunction are not uniquely determined by an energy balance. Instead, the angle of a grain corner in a stable quadrijunction must be greater than or equal to the angle of the same corner in the trijunction formed when the quadrijunction fluctuates into two trijunctions. This angular flexibility has an important effect on the kinetics. Systems in which quadrijunctions are unstable undergo normal grain growth. When quadrijunctions are stable (due to the ratios of R_A and R_B) grain growth can stop due to the

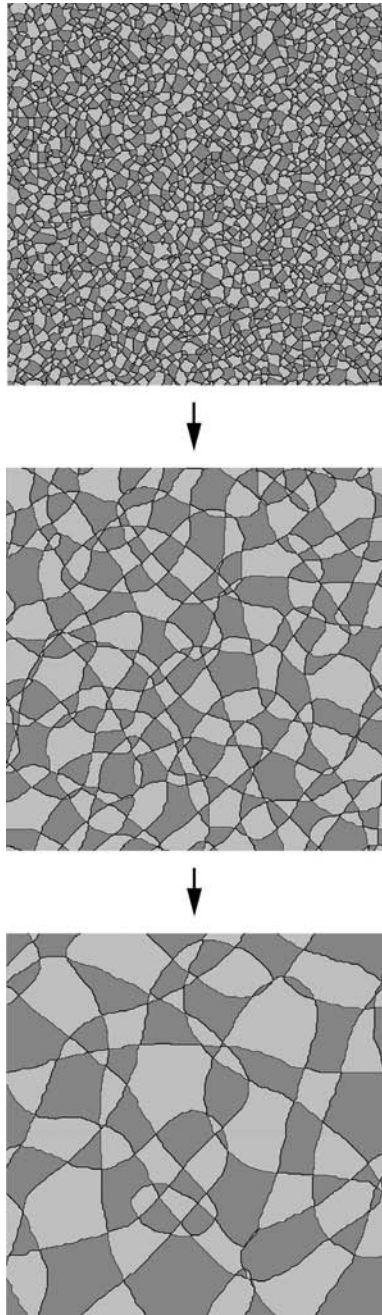


FIGURE 3-18 The evolution of microstructure during a Potts model simulation of a two component system in which the initial distribution of components is equal and $R_A = R_B = 0.5$. The A and B components are differentiated by the gray scale. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0$.

flexibility of these junctions to change their angles and thus eliminate boundary curvature. A systematic analysis of such congruent point grain growth has been carried out by Holm *et al.* [HSC93].

Both greater number of texture components and more sophisticated (and realistic) energy functions such as the Read–Shockley function can be incorporated into the model as we shall see in the applications in Section 3.5.

PROBLEM 3-16: Anisotropic Energy

Model the grain growth of a two component system in which $R_A = R_B = 0.5$ using $kT_s = 0$, a triangular lattice, Glauber dynamics, and a Metropolis transition probability function. Compare your results with the square lattice simulations shown in Figure 3-18.

PROBLEM 3-17: Anisotropic Energy

Modify your code for Problem 3-16 so that the system energies become isotropic when the average grain size reaches a 10th of the system dimensions. You should see the grain structure self-segregate and then de-segregate.

3.3.4 Boundary Mobility

To simulate the case where the mobility is also a function of the boundary character, $\mu(s_i, s_j, \eta_i, \eta_j)$, then we must modify the probability transition function so that probability of a spin flip is proportional to the mobility of that boundary. The Metropolis probability transition function then becomes:

$$P(\Delta E) = \begin{cases} p_0 & \text{if } \Delta E \leq 0 \\ p_0 \exp \frac{-\Delta E}{kT} & \text{if } \Delta E > 0 \end{cases} \quad (3.20)$$

where $p_0 = \frac{\mu(s_i, s_j, \eta_i, \eta_j)}{\mu_m}$ and μ_m is the maximum mobility in the system.

For the simplest model system with two phases or components A and B, p_0 is reduced to a simple binary function:

$$\mu(s_i, s_j, \eta_i, \eta_j) = \begin{cases} 0 & \text{in the grain interiors } (s_i = s_j, \eta_i = \eta_j) \\ M_{AA} & \text{for A–A boundaries } (s_i \neq s_j, \eta_i = \eta_j = A) \\ M_{BB} & \text{for B–B boundaries } (s_i \neq s_j, \eta_i = \eta_j = B) \\ M_{AB} & \text{for A–B boundaries } (s_i \neq s_j, \eta_i \neq \eta_j) \end{cases} \quad (3.21)$$

When $M_{AA} = M_{BB} = M_{AB}$ the system becomes isotropic and normal grain growth is observed. In the case where $M_{AA} = M_{BB} < M_{AB}$ the A–B boundaries have a higher mobility than the A–A and B–B boundaries, which means that isolated A or B grains grow rapidly if they have a size advantage over their neighbors. Equally if they do not have a size advantage, so are not favored to grow, the mobility advantage acts to shrink these grains. In systems where one phase is initially in the minority, this leads to a phenomenon called abnormal grain growth where the minority component grows to become the majority component, as in Figure 3-20.

More sophisticated mobility functions can be incorporated into the model to simulate the effect of temperature gradients, since mobility of interfaces are often a function of temperature. This is easily implemented into the model by making the mobility a function of the lattice dimensions, for example, $\mu(s_i, s_j, x, y)$.

The odd thing one notices about Potts model simulations as soon as one becomes familiar with them is that increasing temperature does not increase the mobility of the boundaries. This

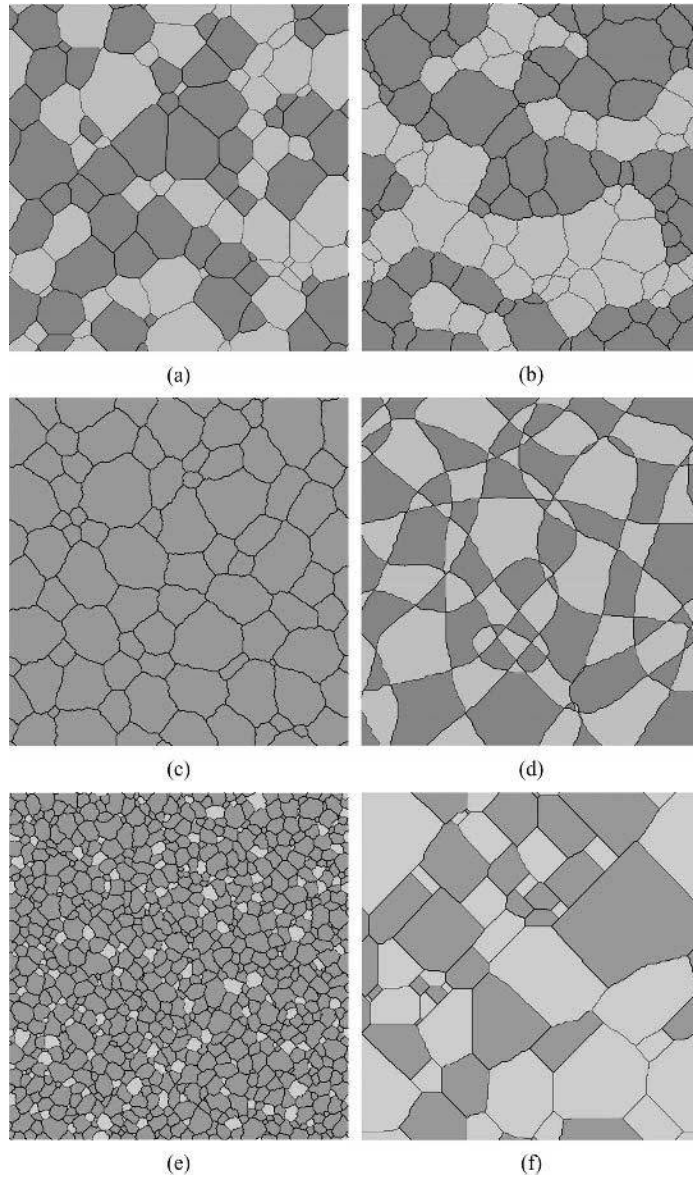


FIGURE 3-19 The effect of anisotropic boundary energy on microstructure during grain growth of a two component system in which the initial distribution of the A and B components was equal, (a) $R_A = R_B = 1$, $kT_s = 0.75$, the isotropic case, (b) $R_A = R_B = 1.5$, $kT_s = 0.75$, (c) $R_A > R_B$, $R_A = 1$, $R_B = 0.67$, $kT_s = 0.75$, (d) $R_A = R_B = 0.5$, $kT_s = 0$, (e) $R_A > R_B$, $R_A = 1.5$, $R_B = 1$, $kT_s = 0.75$, (f) $R_A = R_B = 1.3$, $kT_s = 0$. The simulation was performed using a square (1,2) lattice, Glauber dynamics, and Metropolis transition probability function.

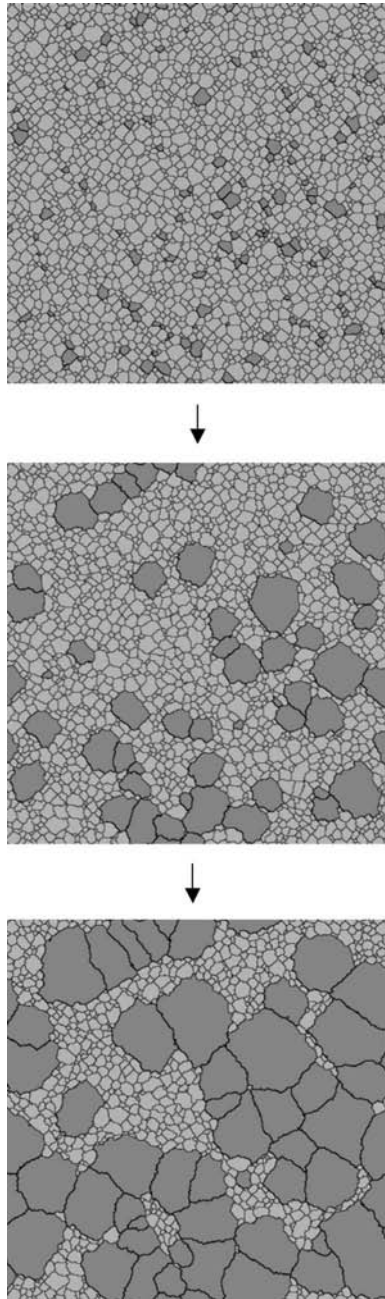


FIGURE 3-20 The evolution of microstructure during a Potts model simulation of a two component system in which the initial distribution of components is unequal and the A-B boundaries have a mobility advantage: $f_B = 0.05$, $M_A = M_B = 1$, $M_{AB} = 100$. The A and B components are differentiated by the gray scale. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.75$.

is obvious from the form of probability transition functions, where the intrinsic mobility of the boundaries is set at $kT_s = 0$. Thus temperature just serves to roughen the boundaries and so prevent lattice effects. How then do we simulate the effect of “real” temperature? It is done by making the intrinsic mobility $\mu(s_i, s_j)$ a function of real temperature. There are many ways to do this which will be covered in Section 3.5. For now it is enough to note that we refer to two temperatures in Potts models, the simulation temperature T_s , which affects the roughness of the boundaries and the real temperature, T_r , which affects the relative mobilities of the boundaries.

PROBLEM 3-18: Anisotropic Mobility

Write a Potts model code to simulate the grain growth of a two component system with anisotropic mobility. Write the code in such a way that the system undergoes an initial period of isotropic growth for a time, t_{init} , to establish an equiaxed grain structure with fractions f_A and f_B of the two components. For $t > t_{\text{init}}$ allocate mobilities M_{AA} , M_{BB} , M_{AB} to the boundaries. Experiment with varying the mobility advantages and the fraction f_A . Show that abnormal grain growth does not occur if the volume fraction of both components is initially equal.

PROBLEM 3-19: Anisotropic Mobility

Write a Potts model code to simulate the effect of moving “real” temperature gradient during normal grain growth of a single component system with anisotropic mobility.

3.3.5 Pinning Systems

So far we have considered the microstructure evolution of two component systems in which the volume fractions of both components are unconserved. What happens when the volume fractions are conserved? The simplest case is where one component is static or inert and the other component is free to evolve its microstructure.

Figure 3-21 shows the evolution of the microstructure for a Potts model simulation using Glauber dynamics for component A and where component B was static. Note how in the initial stages the microstructure of component A evolves but then boundary motion becomes increasingly slow until the microstructure becomes pinned.

The incorporation of an inert component does not require any modification of the Hamiltonian of the system. It requires only the modification of Potts algorithm so that sites of the inert component never undergo spin or component swaps.

PROBLEM 3-20: Pinning Systems

Write a square lattice isotropic Potts model code to simulate 2D microstructural evolution of a two component system, in which component B is static and distributed randomly at the beginning of the simulation as particles of size 3×3 sites. Perform a series of simulations which explore the effect of altering the volume fraction of the B component. What are the issues with using low volume fractions of B?

Where does the pinning behavior originate from if the Hamiltonian used is identical to that for grain growth? There is no explicit implementation of any pinning mechanism or pinning force. The answer is that pinning is a complex emergent phenomenon which arises out of the interaction of many parts of the system. In the case of a single boundary interacting with a single boundary, when a boundary encounters a particle it is initially sucked onto the particle because this lowers the local interfacial energy (specified by the Hamiltonian). If it is a 2D boundary it splits into two boundary segments, each of which can independently satisfy the equilibrium

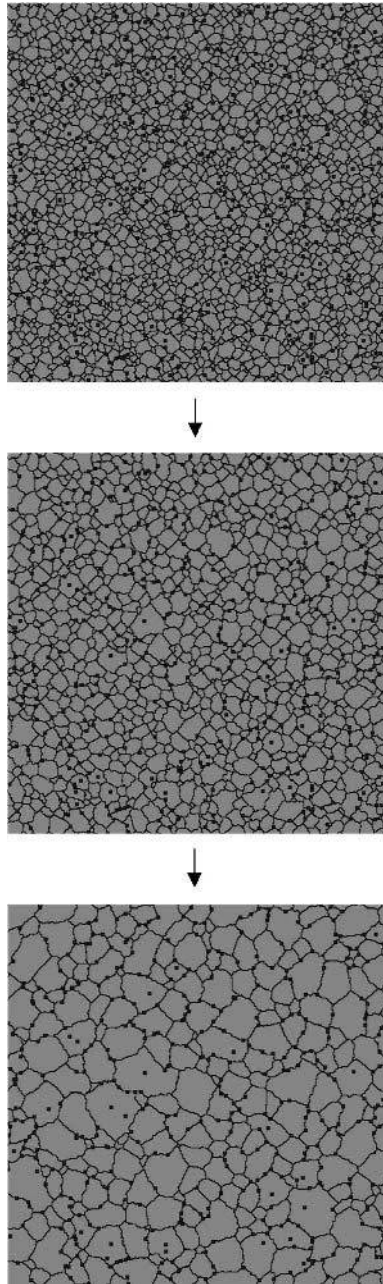


FIGURE 3-21 Potts model simulation carried out on a square lattice, using Glauber dynamics and $kT_s = 0.75$. The second phase has an unchangeable index and so pins the primary phase. The simulations were performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.75$.

surface tension required at the particle interface. The point at which each segment attaches to the particle effectively becomes a node. Detachment from the particle can only occur if the two boundary segments rotate by an angle of π degrees, see Figure 3-22. This results in the stabilization of grains with less than six neighbors, leading to the stagnation of grain growth (which is driven by instability of grains with less than six sides), the strong pinning behavior observed, which is typical of 2D systems.

Performing simulations at non-zero T_s allows boundaries to depin from particles thermally, especially when they are small. It should be noted also that physics of pinning is dimensionally dependent, which means that 2D and 3D pinning are very different phenomena. This is important when using these simulation techniques to understand real microstructural phenomena such as Zener pinning. The theory of pinning and the ability of Potts model simulations to capture the physics of real pinned systems will be dealt with more fully in Section 3.5.8. For the moment it is enough to note that incorporating pinning phases into the model is relatively straightforward, and many different pinning phase morphologies can be modeled. Figure 3-23 shows the wide range of pinning phase morphologies and pinned structures possible in Potts model simulations.

PROBLEM 3-21: Pinning Systems

Write a square lattice anisotropic energy Potts model code to simulate 2D microstructural evolution of a two component system, in which component B is static and distributed randomly at the beginning of the simulation as single particle sites. Perform a series of simulations which explore the effect of changing the ratio J_{AB}/J_{AA} .

3.3.6 Stored Energy

The Q -state Potts model can be used to simulate the effect of volume energy term, such as in the phenomena of recrystallization. The state of each cell is described by a crystallographic orientation and scalar variable that describes the stored energy (in the case of recrystallization in the form of dislocations and point defects). Thus the Hamiltonian for the system is changed to reflect this:

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j) + h_i \quad (3.22)$$

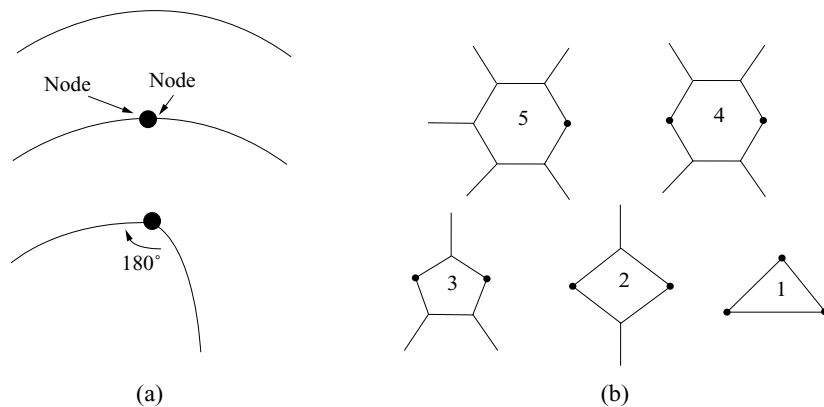


FIGURE 3-22 An illustration of the strong pinning by particles in 2D systems, (a) the case of a single particle becoming a node, (b) stabilization of grains with less than six neighbors by the presence of second-phase particles [Hil65].

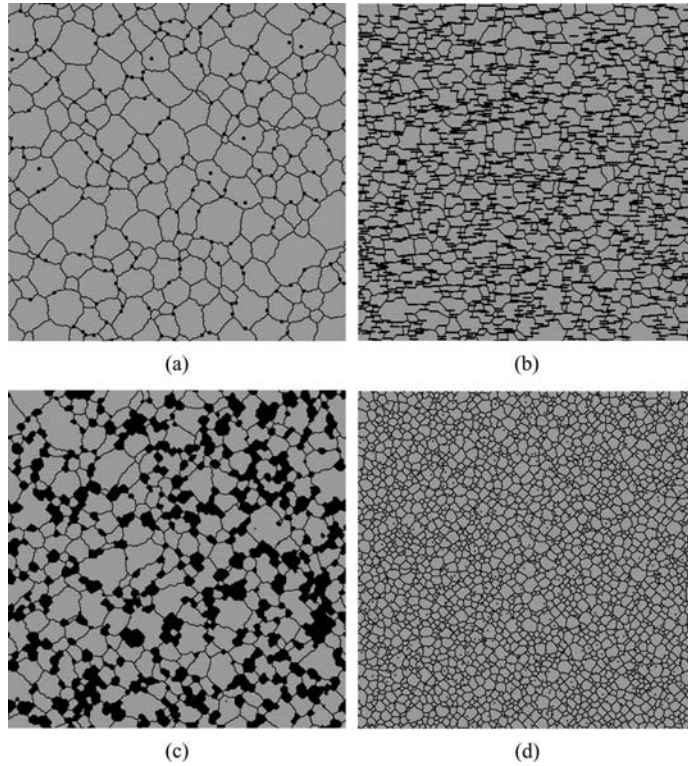


FIGURE 3-23 Illustrating the different types of pinning phase morphology that can be simulated in Potts model simulations, (a) square particles (2×2 sites), (b) aligned rods (10×1 sites), (c) immobile grains (average size 10 sites), (d) dispersoids (1 site). The simulations were performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.75$.

where h_i represents the stored energy term for each site and

$$\gamma(s_i, s_j) = \begin{cases} 0 & \text{in the grain interiors} \\ \frac{J}{2} & \text{at the grain boundaries} \end{cases} \quad (3.23)$$

In the simplest system every site is allocated the same stored energy $h_i = H$, and so the ratio of H/J determines the behavior of the system. For low values of H/J normal grain growth is observed, but as H/J increases the stored energy term dominates. Figure 3-24 shows the microstructural evolution of such a system in which each grain in an initial equiaxed grain structure is assigned a uniform level of stored energy. As the boundaries move, driven both by the stored energy driving force and the curvature driving force, they sweep out areas of zero stored energy. This enhanced boundary motion gives some grains an advantage and they grow to consume the others. Once all the stored energy is consumed the system reverts to normal grain growth behavior. The ratio of the stored energy and the interfacial energy is a crucial variable in the system. If the stored energy of the system is very large compared to the interfacial energy, then boundary growth becomes chaotic.

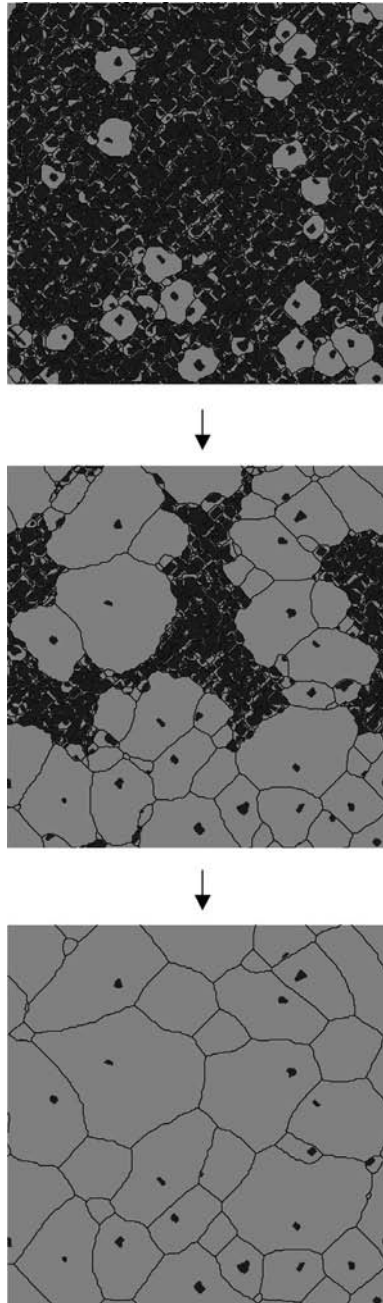


FIGURE 3-24 *The effect of stored energy on an evolving microstructure. A initial equiaxed grain structure is assigned uniform levels of stored energy ($H/J = 0.8$) indicated by a gray scale. Strain-assisted boundary motion occurs, which mimics abnormal grain growth behavior. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0$.*

PROBLEM 3-22: Strain Enhanced Grain Growth

Write a 2D Potts model code to simulate the effects of stored energy. Run a series of simulations in which you grow the grain structure until it is equiaxed and then assign all sites a random value of H . Observe the different growth morphologies.

PROBLEM 3-23: Recrystallization

Write a 2D Potts model code to simulate recrystallization. To initialize the simulation, assign each site a random level of stored energy. Control the nucleation of new grains by varying H_{\max}/J from 0.1–2.0. Observe the different growth morphologies.

3.3.7 Summary

This concludes the initial investigation of the Potts model. The aim of this section has been to give the reader a flavor of what phenomena the Potts model is capable of simulating. By the end of this section the reader should have experience with coding their own Potts models to simulate isotropic grain growth; anisotropic systems; recrystallizing systems; and pinned systems. We have emphasized the importance of gaining practical experience with simulations as a way of understanding the emergent behavior of multicellular systems. Although this is often easier to do in two dimensions, we have seen that the physics can differ significantly in two dimensions and three dimensions, particularly in the case of pinning phenomena. We have also seen how it is important to validate the model by comparing the observed behavior quantitatively with theory. So far we have not dealt with any industrial applications; this is dealt with in Section 3.5. Little attention has been paid to algorithms; this is important as the bigger systems required for the applications mean that it is essential to improve the efficiency of the algorithms. This will be dealt with next.

3.4 Speed-Up Algorithms

For small 2D lattices the vanilla Potts algorithm (Figure 3-6) is sufficiently fast for most purposes. However, in order to use the model to apply to industrially relevant applications, large 3D lattices and long simulation times are required. Such work usually requires speed-up algorithms to be employed. This section presents three such algorithms, each of which uses a different speed-up strategy. All of them have their advantages and disadvantages, and so it is ultimately for the user to decide which is best to employ.

Before embarking on coding a speed-up algorithm, it is important to have a reliable vanilla code running with which the results of the speed-up algorithm can be compared. This requires organization and a methodical approach to programming and simulations. Figure 3-25 shows one possible way of organizing the work. One directory contains each version of the simulation code. A second directory contains the raw data (e.g., snapshots of the lattice at different times) from a single set of simulations, in which the simulation variables, initial conditions, and the version of the code used are all recorded. A third directory contains tools which can take the raw data, and postprocess it to produce microstructural data, for example, tools that can calculate the grain size, or produce a movie from the snapshots. The advantage of this approach is that all tools can be used with all the raw data regardless of which simulation code is used. Second, if the same experiment is carried out using two different simulation codes, then the same tool can be used to postprocess both data sets, and it is very easy to quantitatively compare the speed-up algorithms with the vanilla code.

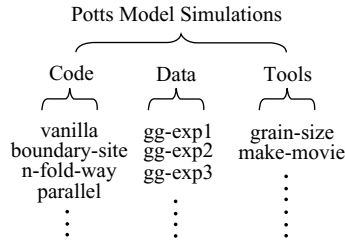


FIGURE 3-25 Suggested data structure for Potts model simulations.

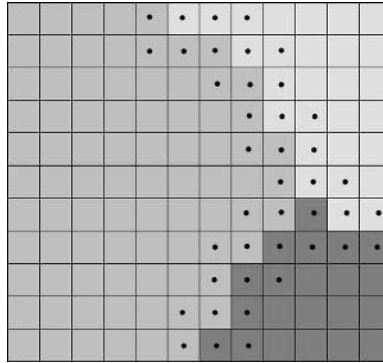


FIGURE 3-26 Identification of boundary sites in the Potts model.

3.4.1 The Boundary-Site Algorithm

The boundary-site algorithm is the simplest of the speed-up algorithms. It relies on the fact that when simulating interface migration it is only the boundary sites that are important (Figure 3-26). The swaps that occur within the domains are in fact problematic, since they represent a nucleation event, which for most applications is unphysical. Excluding these events by using $kT_s = 0$ has the problem that it causes lattice effects. An algorithm that excludes these events *a priori* even at finite temperatures requires the creation of a new algorithm which is not a vanilla Potts algorithm, but a variant, which we will call a boundary-site algorithm.

These boundary-site algorithms are much more efficient for simulating boundary migration than vanilla Potts algorithms because in most systems of interest, the boundary sites are not only the most important sites but also in the minority. Thus by excluding non-boundary sites from potential spin swaps, more CPU time is spent on simulating boundary migration.

There are many ways of ensuring that only boundary sites are sampled for potential spin swaps, each of which has an associated computational overhead and memory allocation implications. The easiest way is to modify the vanilla Potts model so that after a site is selected at random, a neighbor check is performed, and if the site is found not to be a boundary site, time is incremented and a new site selected. Further efficiency gains can be obtained by maintaining a list of the boundary sites and only picking sites from that list. The maintaining of this list has a computational overhead associated with it, and so the overall efficiency of the algorithm depends of the boundary-site/interior-site ratio of the system. In this case incrementing the time is also more complicated, since the algorithm must accurately calculate the effective time, τ_b , associated with each attempted boundary-site swap. In such a simple algorithm τ_b is equal

to the number of boundary sites in the list divided by N . A schematic representation of a boundary-site algorithm is shown in Figure 3-27.

PROBLEM 3-24: Boundary-Site Algorithm

Write a Boundary-site algorithm to simulate normal grain growth on a 200×200 lattice, using $Q = 50$, Glauber dynamics, and a Metropolis transition probability function. Compare the kinetics and grain size distributions with those from a vanilla Potts simulation and show they are identical when $kT_s = 0$. Identify the deviations between the models for non-zero kT_s . Calculate the speed-up achieved as a function of lattice size.

Further speed-up can be obtained by only allowing boundary sites to swap to a spin of one of their neighbors. Once again this is more physically realistic in that it rules out the boundary nucleation events that can occur at finite temperatures. Choosing one spin randomly from the set of available neighbor spins again has implications for τ_b . The effective speed-up is different for each site and depends on its neighborhood. Because this algorithm is actually doing something slightly different from the vanilla Potts, an exact quantitative match is not relevant. However, a check should always be performed with every new code to make sure that grain growth obeys von Neumann laws, with an invariant normalized grain size distribution, and an average grain size which obeys a parabolic growth law.

PROBLEM 3-25: Local Spin Swaps

Write a boundary-site algorithm, which includes local spin swaps, to simulation normal grain growth on a 200×200 lattice, using Glauber dynamics, and a Metropolis transition probability function. Compare the kinetics and grain size distributions with those from a vanilla Potts algorithm and show they are identical when $kT_s = 0$. Calculate the speed-up achieved as a function of Q , for $Q = 2, 25, 50, 100, N$.

3.4.2 The N -Fold Way Algorithm

A speed-up algorithm that *exactly* corresponds to the vanilla Potts model is known as the *n-fold way* algorithm and was first developed by Bortz *et al.* [BKL75]. The algorithm eliminates all unsuccessful spin flips attempts *a priori*, so that all spin flips result in domain evolution. It requires a list of all possible successful spin flips to be compiled along with their transition energies ΔE . An activity $\pi_i = P(\Delta E_i)$ of the N possible spin flips is defined [BKL75] with the total system activity A is given by:

$$A_n = \sum_{i=1}^N \pi_i \tag{3.24}$$

Each site is visited with a probability weighted by its activity, an effective spin flip is performed, and the site and its neighbors are reevaluated and π_i updated.

In the conventional Potts model the simulation is incremented after each attempted spin flip. In the *n-fold way* each spin flip is successful, so the time increment must be scaled by the average time between successful flips in the vanilla algorithm. This time increment is

$$\Delta t = -(\tau/A_n)\ln R \tag{3.25}$$

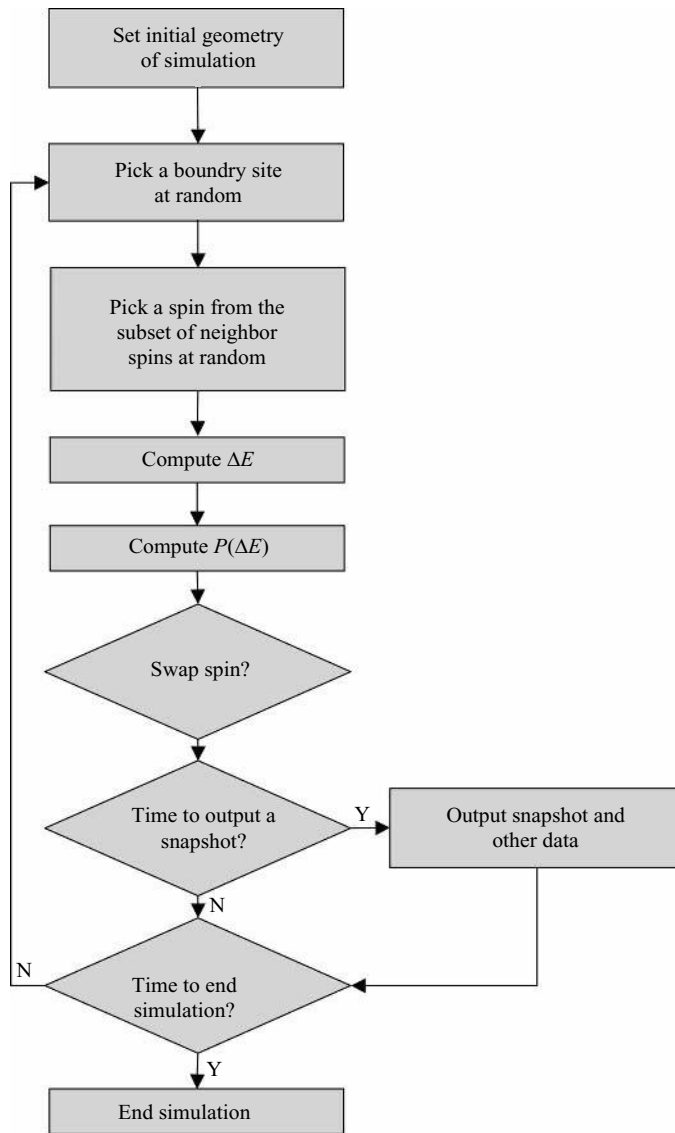


FIGURE 3-27 *The boundary site algorithm for the Potts model.*

where R is a random number uniformly distributed on the interval $(0,1)$. This time increment decreases as the total system activity decreases, reflecting an increased efficiency of the algorithm. Hassold and Holm [HH93] have shown that using this type of n -fold way algorithm results in identical simulation dynamics in substantially less computation time. The algorithm is shown schematically in Figure 3-28.

The algorithm is efficient in the case where either $kT_s = 0$ or when $kT_s \neq 0$ when all spin flips are confined to the boundary sites. For other cases the increase in computational effort involved in compiling and updating lists of the active sites usually outweighs the advantages of the algorithm.

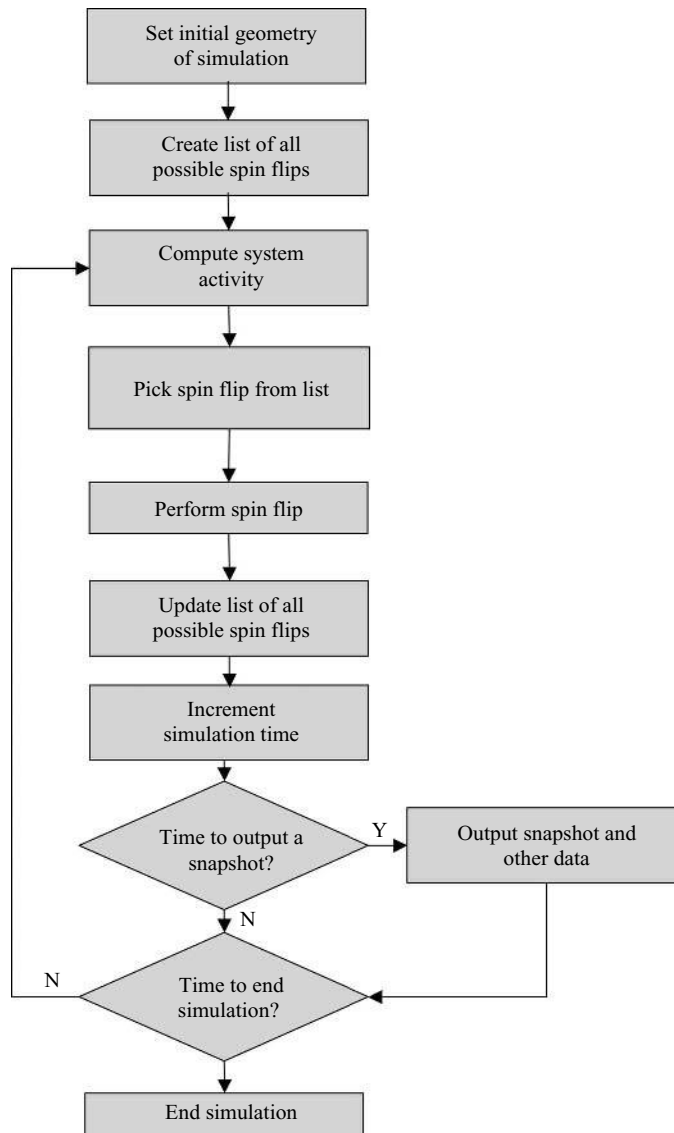


FIGURE 3-28 The n -fold way algorithm for the Potts model.

PROBLEM 3-26: N-fold Way

Code a 2D triangular Ising model using a vanilla algorithm and an n -fold way algorithm. Quantify the efficiency gain for grain growth for various lattice sizes, 10×10 and 100×100 and 1000×1000 .

3.4.3 Parallel Algorithm

The Potts algorithm does not readily parallelize in the obvious way of assigning each of P processors a subset of the lattice sites. This is because two or more processors may pick adjacent sites. If this occurs then when the two processors attempt to calculate ΔE for a spin flip, they

will each do so using incorrect information about neighboring spins (each other). This violates the “detailed balance” which demands that two or more sites may not be flipped simultaneously if their interaction affects the Hamiltonian. A valid parallelization method is to use a checkerboarding approach in which each processor simultaneously attempts flip spins in a subdomain in which all sites share no neighbors with other subdomains.

Checkerboarding involves assigning each lattice site one of a finite set of flavors. For example Figure 3-29(a) shows a 2D square lattice checkerboarded with two flavors, those denoted with circular symbols and those with triangular symbols. This can be used to perform Potts model simulations using first nearest neighbors, because each of the circle sites has no nearest neighbors that are circles. Similarly each of the triangular sites has no nearest neighbors that are triangles. Thus instead of picking sites randomly all the circle sites can be updated simultaneously without the result of any flip attempt affecting the other circle sites. This is also true for the triangular sites. Algorithmically, a loop over the flavors replaces the need to randomly pick sites and thus removes the possibility that two processors might try to flip to adjacent sites.

The checkerboarding method can be applied to 2D square lattices using first and second nearest neighbors but in this case four flavors of lattice sites are needed to ensure that simultaneous updating does not invalidate detailed balance, see Figure 3-29(b). In 2D triangular lattices using first nearest neighbors, three unique flavors are needed. In three dimensions using the 26 first, second, and third nearest neighbors, eight flavors are needed.

By using checkerboarding P processors can be most efficiently employed if the lattice is subdivided into P contiguous domains. Each processor also must store a copy of the narrow strips

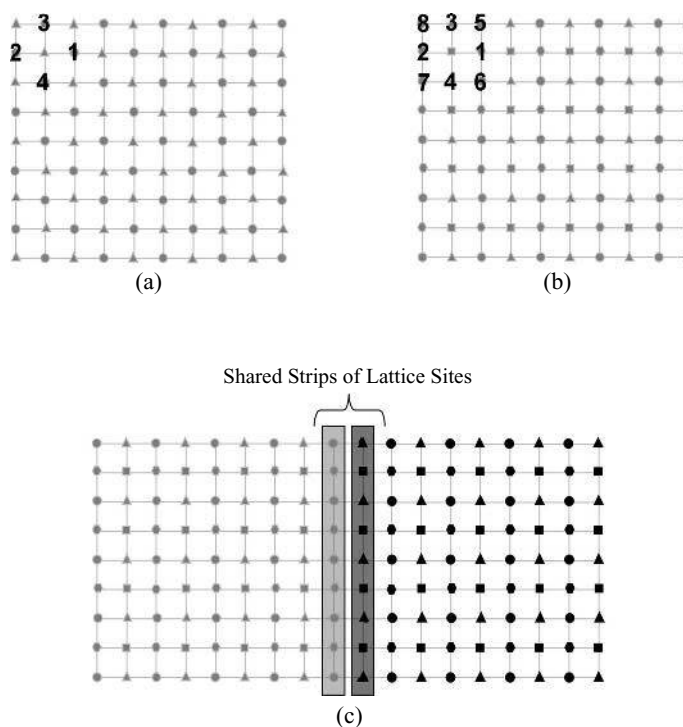


FIGURE 3-29 Checkerboarding of lattices, (a) 2D square lattice using only first nearest neighbors, (b) 2D square lattice using first and second nearest neighbors, (c) showing how sharing of an interface strip allows two processors to simultaneously update their sites using checkerboarding.

(two dimensions) or planes (three dimensions) of the lattice sites that immediately adjoin its subdomain and which are actually owned by neighboring processors as shown in Figure 3-29(c). The important point is that the processors are all synchronized to update the same flavor of lattice site.

The parallel Potts algorithm is shown in Figure 3-30. This algorithm is highly parallel, with the only communication cost being the local exchanges of boundary spins between neighboring

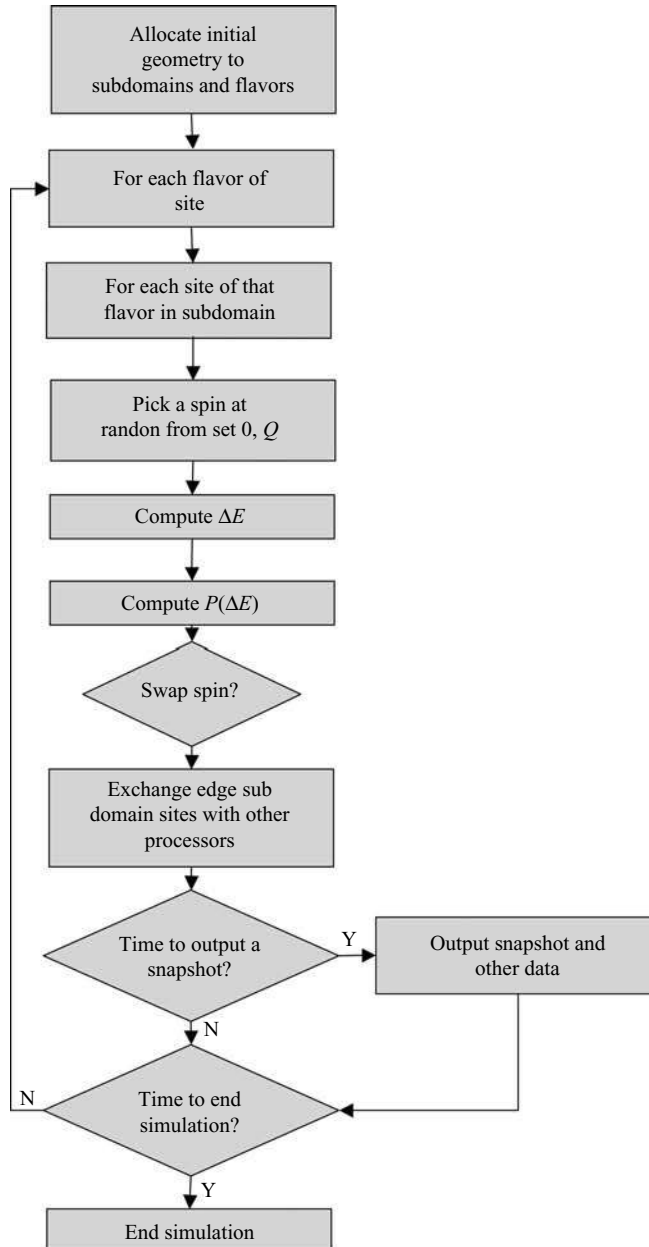


FIGURE 3-30 The parallel algorithm for the Potts model.

processors. These exchanges also serve as synchronization points in the loop over flavors to ensure that all processors work on the same flavor at the same time. In practice, as long as the processor subdomains are of reasonable size (50×50 or larger in two dimensions), the communication costs are only a few percent of the total run time and thus the algorithm can simulate large lattices with parallel efficiencies of over 90% [WPS⁺97].

PROBLEM 3-27: Parallel Algorithm

Code a 3D square Potts model using a parallel algorithm. Perform normal grain growth simulations for lattice size $200 \times 200 \times 200$ and 10^7 MCS, using 1, 2, and 4 processors. Check that identical parabolic kinetics are observed in all cases. Calculate the speed-up per processor of the algorithm.

3.4.4 Summary

This concludes the investigation of the Potts model speed-up algorithms. The aim of this section has been to give the reader a wide selection of Potts algorithms with which to simulate industrial applications. By the end of this section the reader should have experience with coding boundary-site models, n -fold way models, and parallel models. We have stressed throughout that it is important to be organized, systematic, and above all to always verify a new model before applying it to a new phenomenon.

3.5 Applications of the Potts Model

So far we have modeled the general features of grain growth common to all microstructures. However, if we are interested in the microstructural evolution of a particular material then it is important to accurately simulate the microstructural features of the material such as the grain structure, texture, and misorientation distribution function. We will also want to model the kinetics and to compare them with experiment.

3.5.1 Grain Growth

Although it is obvious that the self-organizing behavior of the Q -state Potts model resembles the phenomenon of grain growth, the question arises of how closely do the simulations compare to the experimental measurements of grain growth. To do this we need to measure statistical aspects of the experimental phenomenon and compare them with those measured from the model.

In Section 3.3.1 we noted that the domain structure in the Potts model coarsens in a self-similar manner so that the average domain size increases in time. Experimentally, it is observed that the grain size distribution when normalized by the average grain size remains constant during grain growth. This means that even though some grains grow, while others shrink, the grain ensemble remains self-similar. This type of phenomenon is called normal grain growth. The grain size distribution and the topological distribution derived from 3D Potts model simulations of isotropic grain growth are also observed to be time invariant and in agreement with experimental data, as in Figure 3-31.

PROBLEM 3-28: Normal Grain Growth

Simulate grain growth in three dimensions using the Potts model for $Q = 5, 10, 50,$ and 100 . Plot the grain size distribution as a function of Q and comment on the result.

Measuring grain size in simulations is notoriously laborious for big systems. The simplest way of dealing with this is to issue each site with a unique grain identifier as well as a spin when

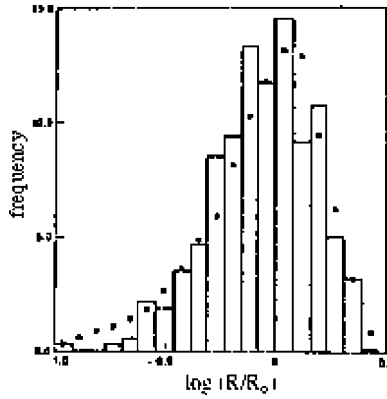


FIGURE 3-31 Grain size distribution during grain growth, a comparison between an Fe sample and the 3D Potts model [AGS85].

setting up the initial simulation microstructure. This unique identifier is swapped along with the spin during any successful spin flip, but is not used except when analyzing the snapshots of the simulations. It provides an easy way to identify all the sites of a particular grain and thus calculate the grain area/volume.

PROBLEM 3-29: Grain Size Perturbations

Simulate 2D grain growth using the Potts model for $Q = 100$. After 100 MCS insert a circular grain into simulation in a random location with a size five times that of the average grain size of the system. Plot the size of this grain normalized by the average grain size of the system against time. Why does this grain not grow abnormally?

The rate at which the average size increases is another parameter by which experimentalists measure normal grain growth. The kinetics of grain growth is characterized by the parabolic equation

$$R_{av}^n - R_0^n = A_{gg}t \tag{3.26}$$

where R_0 is the initial grain size and A_{gg} is a constant. The grain growth exponent, n , has been the focus of much of the debate in the grain growth community. Hillert’s theoretical derivation [Hil65] gives $n = 2$ but most experiments show grain growth exponents much greater than this; typically the values lie between $n = 2.5$ and $n = 4$. It has been argued that impurity effects may be responsible for the deviation from the ideal value. However, even data from a wide range of ultrapure metals show considerable deviation from $n = 2$. 3D Potts model simulations of isotropic grain growth show grain growth exponents in the range $2 < n < 2.5$. Why the range you might ask? The measured exponent depends on many variables of the system, but importantly on the size of the system, kT_s , Q , and on initial distribution of grain size. Issues about why the grain growth exponent is so sensitive to these variables have yet to be definitely resolved.

PROBLEM 3-30: Effect of Temperature on Grain Growth Exponent

Simulate grain growth in two dimensions using the Potts model for $kT_s = 0$, $kT_s = 0.5$, and $kT_s = 1.0$. Plot average grain area as a function of time for each temperature and calculate the grain growth exponent. Note the early nonlinear transient at the beginning and end of the simulations. Why do these occur?

3.5.2 Incorporating Realistic Textures and Misorientation Distributions

Figure 3-32 shows a 2D map of a typical microstructure obtained using an electron back-scattered diffraction (EBSD) method. It illustrates clearly that each grain has a unique crystallographic orientation and that each grain boundary will have a unique misorientation and rotation axis. It is essential to capture this level of complexity in the Potts model if we are to simulate the behavior of real experimental systems.

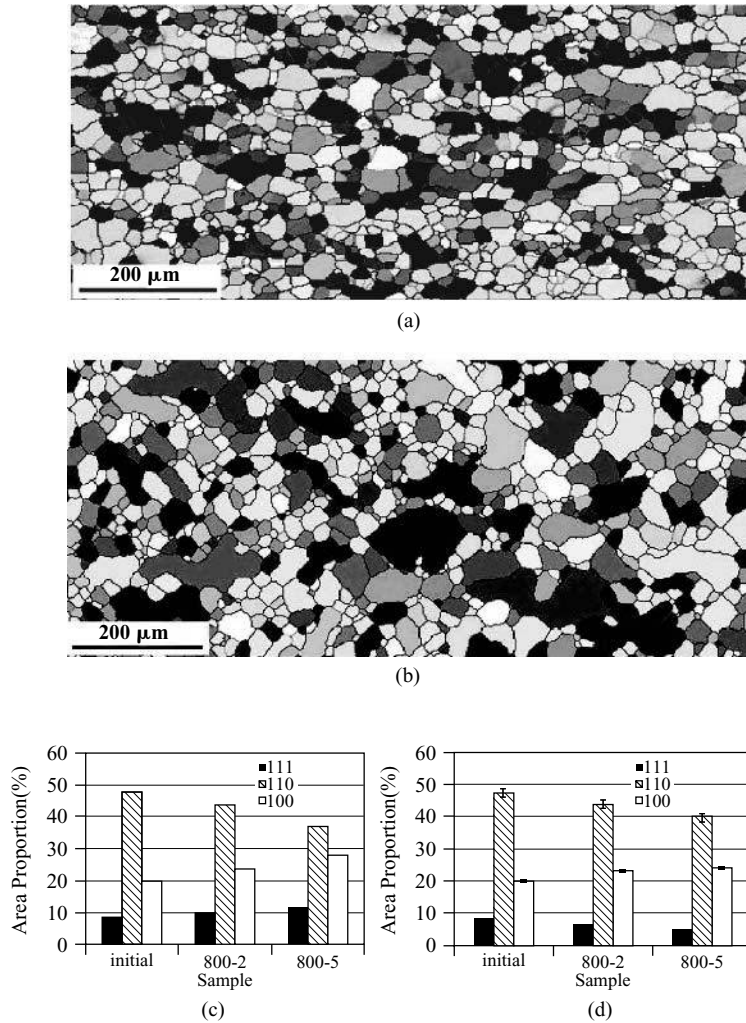


FIGURE 3-32 (a) EBSD map of annealed microstructure of a transformer silicon steel specimen, (b) Potts model simulated microstructure after grain growth of a transformer silicon steel specimen using as recieved starting microstructure from EBSD, (c) shows the development of the $\langle 111 \rangle$, $\langle 110 \rangle$, and $\langle 100 \rangle$ textures fibers as measured from experiment, (d) shows the development of the $\langle 111 \rangle$, $\langle 110 \rangle$, and $\langle 100 \rangle$ textures fibers as measured from Potts model simulations [HMR07].

For 2D simulations the most straightforward way of doing this is to incorporate the microstructural information and the crystallographic information directly from the EBSD data set. Since each grain in the experimental data set has a unique crystallographic orientation, it is important to use a unique spin Potts algorithm (as described in Section 3.4). Typically this means that each lattice site in the simulation is allocated a unique spin number and a table is created which correlates the spin number with the Euler angles corresponding to the crystallographic orientation of the grain. A normal Potts model simulation can then be performed with the crystallographic information of each lattice site being used to plot the evolution of microstructure in the development of textures as in Figure 3-32.

Although this process seems straightforward enough, there are some important issues that we have omitted to discuss. First, we did not incorporate the experimental microstructure with its associated crystallographic details directly from the microscope into the model. In such EBSD maps there is a good deal of noise that corresponds to some pixels being identified as single site grains, when in fact they are more likely to be a site whose orientation was incorrectly measured. This kind of noise may not be just due to incorrect measurement; in the case of deformed microstructures, the dislocation and other defects may be associated with low angle subboundaries which are topologically distinct from grain boundaries. Also since the map is a 2D section of a 3D microstructure, some topological features may appear to be noise when in fact they are the tip of a grain protruding into the 2D section. For these and many other reasons, the importing of a microstructure into a Potts model often requires a clean-up filter to be applied so that these effects can be mitigated and features which are not going to be included in the model can be removed. However, it is obvious that using these filters can also distort the data set in some cases changing the fundamental of the microstructure to be studied. For a modeler the lesson here is to ask for both the filtered and unfiltered data from the microscope, to ensure that radical massaging of the data is not occurring which would then render the simulations meaningless.

These 2D microstructure maps are extracted from a small volume of the material. This volume is represented in the model only through the boundary conditions. Thus choice of the boundary conditions is important when performing simulations and also when interpreting the results. Choosing periodic boundary conditions is not an option since there will not be continuity across the simulation boundaries. The choice of mirror or free surface boundaries is available, and both have implications. Furthermore the fact that a 2D simulation is being performed of a 3D phenomenon needs also to be taken into account. Upshot of these factors is that extreme care should be taken when carrying out and interpreting such simulations. The best practice is to carry out a large number of simulations using a large number of different input microstructures and to measure the evolution of average characteristics, for example, the average texture, misorientation distribution function (MDF), and grain size. It is when these averaged quantities are compared with experimental results that meaningful conclusions and predictions may be drawn, see Figure 3-32.

PROBLEM 3-31: Incorporating Realistic Textures and Misorientation Distributions

Write a code to import the output from a experimental EBSD orientation map and import it into the Potts model. Take a snapshot of the imported microstructure and compare it with the EBSD map. Measure the grain size, MDF, and texture of the imported microstructure and compare your results with those calculated by the EBSD software. Use your imported microstructure as the starting configuration for an isotropic grain growth simulation using the Potts model.

PROBLEM 3-32: Comparing the Effect of Boundary Conditions

Use your imported microstructure as the starting configuration for an isotropic grain growth simulation using the Potts model. Compare the grain growth kinetics and grain size distributions obtained using mirror boundary conditions with those obtained using free-boundary conditions.

There are no routine methods for extracting the necessary 3D information from experiment. It is possible to combine EBSD with serial sectioning, but this a very labor intensive task and still leaves the problem of how to extrapolate between the sections. 3D X-ray tomography methods have more recently become possible using high energy focused synchrotron X-ray sources, but at the moment the resolution is low and again the method is not widely available. Another approach to this problem is to use computation methods to reconstruct an equivalent 3D microstructure with the grain size, grain size distribution, texture, and MDF, since obtaining these characteristics of the 3D microstructures from experiment is straightforward.

The first step is to obtain a 3D microstructure with the right grain size and grain size distribution. This is done by using a 3D Potts model and using anisotropic mobility to grow an appropriate microstructure using trial and error, see Figure 3-33(a). This is easy for equiaxed microstructures and less easy for more complicated microstructures. Next the experimental texture is discretized into Q orientations and allocated randomly to the spins of the grains of the 3D microstructure. This produces a 3D microstructure with the correct texture but random MDF. This MDF is calculated and quantized into n_b bins, such that S_k is the number of boundaries with misorientations between $k\Delta\theta$ and $(k+1)\Delta\theta$, $k = 0, 1, \dots, n_b$. A system Hamiltonian is defined as the sum of the squared differences between S_k^m and S_k^{exp} :

$$H_{\text{mdf}} = \sum_{k=0}^{k=n_b} (S_k^m - S_k^{\text{exp}})^2 \quad (3.27)$$

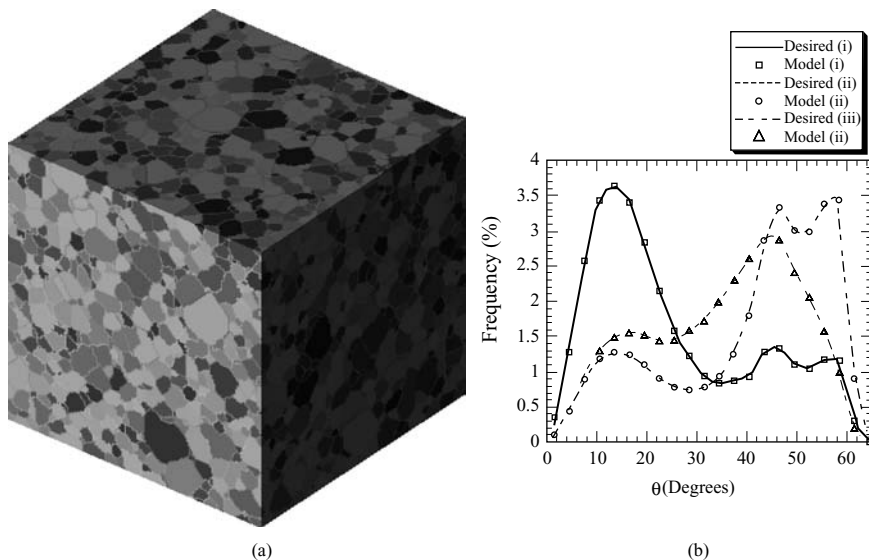


FIGURE 3-33 (a) Three-dimensional equiaxed microstructure grown using the Potts model, (b) Showing the desired and the achieved MDFs generated by discretizing a texture, allocating orientations to the grains, and then using the swapmethod to achieve the desired MDF [MGHH99].

where S_k^m defines the MDF of the model and S_k^{exp} defines the experimental MDF. H_{mdf} is a state variable providing a measure of the difference between the model MDF and the experimental MDF. It is equal to zero when the model MDF and the experimental MDF are identical. We use a Monte Carlo algorithm in order to minimize H_{mdf} and in doing so construct the desired MDF. The method is as follows: two grains are chosen at random, and the H_{mdf} due to swapping their orientations is calculated. The probability $p(H_{\text{mdf}})$ that the swap is accepted is a Metropolis function. Figure 3-33(b) shows the wide range of MDFs that can be achieved using this algorithm. (Read ref. [MGHH99] for more information.)

This swap method is effective and produces a starting 3D microstructure with a texture and MDF that are identical to the experiment. It is not elegant. More ambitious ways of reconstructing 3D microstructures from 2D metrics, which integrate the microstructure generation, texture generation, and MDF optimization steps into one step have been proposed. Unfortunately none yet have been shown to work. Progress on 3D X-ray methods may make the swap method redundant in the future. It will be interesting to see.

3.5.3 Incorporating Realistic Energies and Mobilities

Read and Shockley [RS50] derived an analytical expression for the energy (per unit area) of a low angle grain boundary. The boundary is assumed to comprise of a regular array of dislocations. The boundary energy can be expressed as a function of the misorientation:

$$\gamma = \gamma_0 \theta (A - \ln \theta) \quad (3.28)$$

The parameters γ_0 and A are related to elastic constants and properties of the dislocation cores: γ_0 sets the overall energy scale, and A adjusts the angle of the maximum grain boundary energy. For large angle grain boundaries, this model would not be expected to be valid, as the dislocation cores would overlap substantially, and their interaction could not be neglected. Nevertheless, this formula has been successfully fit to experimental grain boundary energies for wide misorientation angles. Thus a normalized version of equation (3.28) can be used to model the functional form of a general grain boundary in the Potts model:

$$J_{RS} = J_0 \left(\frac{\theta}{\theta_m} \right) \left\{ 1 - \ln \left(\frac{\theta}{\theta_m} \right) \right\} \quad (3.29)$$

where θ_m is the misorientation angle that results in the maximum boundary energy of the system. Experimentally it is observed to lie between 10° and 30° , depending on the system [SB95].

As we have seen in the last section, in the Potts model a continuum microstructure from experiment can be bit mapped onto a discrete lattice where each lattice site is allocated an index s_i and a discrete crystallographic orientation O_i so that all sites within a grain have the same index and orientation. In such a system the Hamiltonian becomes:

$$E = \sum_{i=1}^N \sum_{j=1}^z \gamma(s_i, s_j, O_i, O_j) \quad (3.30)$$

Thus boundaries are represented by interfaces between neighboring sites of unlike index and possess an excess energy given by equation (3.29), thus:

$$\gamma(s_i, s_j, O_i, O_j) = \begin{cases} 0 & \text{in the grain interiors } (s_i = s_j, O_i = O_j) \\ \frac{J_{RS}}{2} & \text{for boundaries } (s_i \neq s_j, O_i \neq O_j) \end{cases} \quad (3.31)$$

Clearly in most real systems mobility is also a function of the boundary character: $\mu(s_i, s_j, O_i, O_j)$. Thus we must modify the probability transition function so that probability of a spin flip is proportional to the mobility of that boundary. The Metropolis probability transition function then becomes:

$$P(\Delta E) = \begin{cases} p_0 & \text{if } \Delta E \leq 0 \\ p_0 \exp \frac{-\Delta E}{kT_s} & \text{if } \Delta E > 0 \end{cases} \quad (3.32)$$

where $p_0 = \frac{\mu(s_i, s_j, O_i, O_j)}{\mu_m}$ and μ_m is the maximum mobility in the system. Note that these are reduced mobilities measured from experiment and have a wide range of functional forms. Getting this data from experiment is often nontrivial and, like boundary energies, these mobilities may also be a function of boundary plane (especially in the case of twins) and also composition. By including or not including such factors in a model we are making assumptions about which are the important factors in a system. We are also making the Potts model more complex. Thus it is best practice in such situations to carry out simulations on simple geometry to validate the model before going on to tackle the full 3D polycrystalline system. The simplest of such systems, but which nevertheless still contains boundaries and triple points, is discussed in the next section.

3.5.4 Validating the Energy and Mobility Implementations

Although the implementation of the Read–Shockley energy function seems a straightforward extension of the model to change the boundary energy, it has another implicit effect, which is to change the node angles of the boundaries. As discussed in Section 3.3.3, this changes the boundary curvature acting on a boundary and so the driving force on that boundary. If we are to simulate systems with a continuous range of boundary energies and so a continuous range of node angles, we need to make sure that the discrete nature of the simulation lattice does not affect these angles.

One way to do this is to consider a model geometry such as that shown in Figure 3-34. We consider a system with a constant driving force for motion and in which the triple points have invariant geometry. A similar approach is taken by experimentalists studying boundary and triple point mobility. The grain structure is columnar, with two grains, B and C, capped by a third grain, A. Boundary conditions are periodic in the x -direction and fixed in the y -direction. There are two boundary misorientations in the system: θ_1 is the misorientation angle of the A–B and A–C boundaries, and θ_2 is the misorientation angle of the B–C boundaries. There are two triple junctions in the system, and the geometry is arranged such that these two are identical and symmetric. From equation (3.17) the equilibrium junction angle where θ_1 is the energy of the A–B and A–C boundaries, and θ_2 is the energy of the B–C boundaries.

The driving force acting on the boundary is γ_2/D . Assuming that the driving force is proportional to the velocity of the boundary, the boundary velocity in the y -direction

$$\frac{dy}{dt} = \frac{\mu_1 \gamma_2}{D} \quad (3.33)$$

where μ_1 is the intrinsic mobility of the A–B and A–C boundaries.

To examine the validity of the Q -state Potts method, a nominal γ_2 is set and dy/dt is measured with time. By finding the regime in which dy/dt is constant, and using equation (3.33), the effective γ_2 can be extracted. Figure 3-35 compares the measured γ_2 to the nominal γ_2 . It can be seen that for large γ_2 (i.e., high misorientations) there is good agreement between the

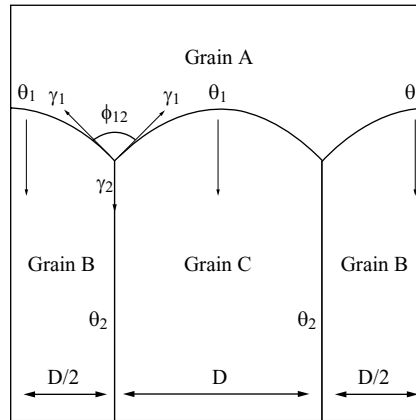


FIGURE 3-34 The boundary geometry used to validate the Q -state Potts model for anisotropic grain growth. Boundary conditions are continuous in the X -direction and fixed in the Y -direction. The boundary between grain A and grains B and C is the only boundary that moves. θ_1 is the misorientation between grain A and grain B and also between grain A and grain C. θ_2 is the misorientation between grain B and grain C. The equilibrium angle of each triple point, ϕ_{12} , is defined by the ratio of the boundary energies of the boundaries that intersect at the triple point, $\gamma(\theta_1)$ and $\gamma(\theta_2)$.

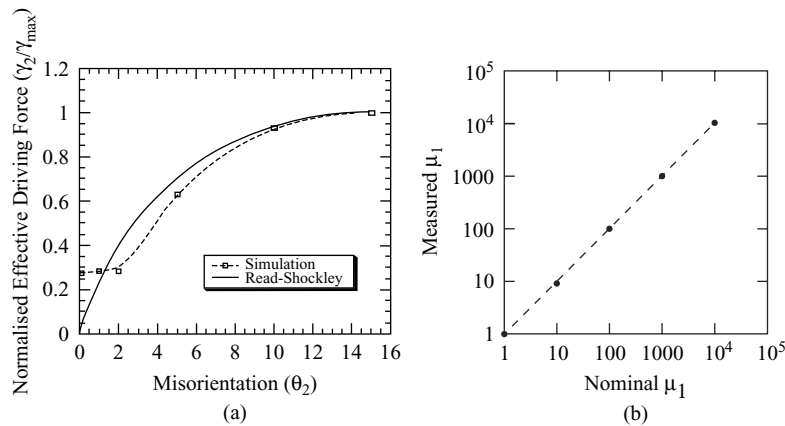


FIGURE 3-35 (a) Measured γ_2 versus nominal γ_2 for Potts model simulations of boundary motion in the system illustrated in Figure 3-34, $kT_s = 0.5$. (b) Measured μ_1 versus nominal μ_1 for Potts model simulations of boundary motion in the system illustrated in Figure 3-34 with $\mu_2 = 1$.

simulation and the theory. But as γ_2 decreases, a deviation from theory is observed; the effective γ_2 becomes constant as the nominal γ_2 continues to decrease. This deviation occurs around $\gamma_2 = 0.4\gamma_1$, corresponding to a misorientation angle $\theta_2 = 2^\circ$ when $\theta_1 = 15^\circ$.

This behavior has its origin in the discrete nature of the lattice. As θ_2 gets smaller relative to θ_1 , the equilibrium triple junction angle, ϕ_{12} , gets larger until it approaches 180° and the A–B and A–C boundaries become nearly flat. Because junction angles must be changed by the addition or removal of an entire step in a discrete lattice, small differences in the junction angle cannot be resolved. That is, at some point, the last step is removed, the boundary

becomes flat, and the triple junction angle cannot change with further decreases in γ_2 . Because the triple junction angle defines boundary curvature, it also defines the driving force. Thus if this angle becomes invariant at some γ_2 , so does the driving force acting on the boundary. This effect is unavoidable in these discrete lattice simulations and hence there is a limit to the range of anisotropies that the model can simulate. For simulations on the square lattice, the limit is reached around $\gamma_2 = 0.4\gamma_1$, when $\phi_{12} = 157^\circ$; larger triple junction angles cannot be resolved.

Note that this effect limits only the maximum triple junction angle and thus the range of boundary energies (anisotropy) that may be resolved. It does not limit the absolute value of the boundary energy. For example, a system of $\theta = 1^\circ$ boundaries, each with energy $\gamma = 0.25$, has 120° triple junctions and can be successfully simulated by the Q -state Potts model. The triple junction limitation need be considered only if a higher angle boundary (in this case, $\theta \geq 4^\circ$) must be included in the system.

The limitation on energetic anisotropy does not affect the model's ability to simulate nonuniform boundary mobility. Since mobility is independent of curvature, it is unaffected by triple junction angles. Figure 3-35 shows the linear relationship between mobility and velocity in the Q -state Potts model over four orders of magnitude. (Read ref. [HMR03] for further information.)

PROBLEM 3-33: Validating a 3D Potts Model

Validate the energy and mobility implementation of a 3D Potts model using a 3D version of the geometry shown in Figure 3-36.

3.5.5 Anisotropic Grain Growth

Having validated the model we are now free to simulate anisotropic grain growth using realistic textures, misorientation distributions using Read–Shockley energies, and anisotropic mobilities. Figure 3-37 shows the evolution of such a system in which the initial microstructure has a strong texture $\langle 100 \rangle$ cube texture. The system undergoes normal grain growth, which causes a tightening of the texture. The boundaries are colored to show their misorientation, black being high misorientation and white being low misorientation. Note how all the high misorientation boundaries (dark colored) are removed from the system during grain growth with all the boundaries becoming white. This causes a reduction in the average misorientation and a narrowing misorientation distribution. This effect is observed experimentally and is due to the high energy boundaries being replaced by low misorientation boundaries.

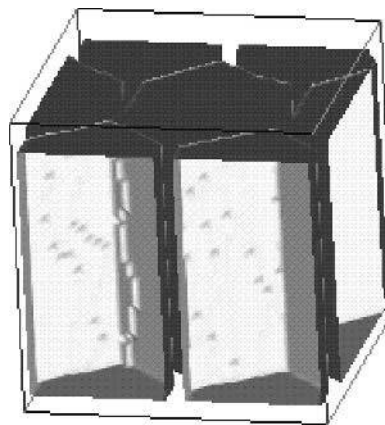


FIGURE 3-36 The 3D hexagonal geometry used to validate the Potts model for anisotropic energies and mobilities [HMR03].

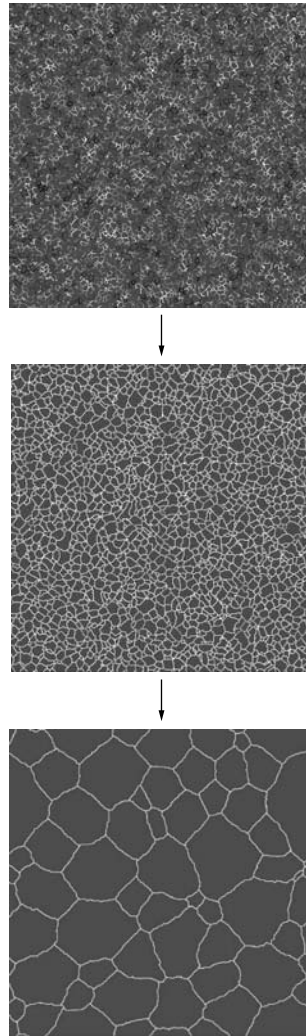


FIGURE 3-37 *The evolution of microstructure during a Potts model simulation of anisotropic grain growth of a single texture component, using Read–Shockley energies and uniform mobilities. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.5$.*

Being able to understand what factors influence such changes in texture and MDF is one of the main advantages of simulation. Turning off anisotropic mobility is an impossible experiment to perform, but it is trivial to perform the same set of simulations with Read–Shockley energies and uniform mobilities, or anisotropic energies and uniform energies. Comparing the results in this case reveals that mobility has little effect on the texture changes, and the energy function is the dominant factor (for more information read ref. [HHM01]).

What if we were to explore the effect of different energy functions, in particular the effect of energy cusps, such as those due to coincidence site lattice (CSL) boundaries? This requires changing the energy function and thus the Hamiltonian of the system.

PROBLEM 3-34: Anisotropic Grain Growth

Write an anisotropic Potts model simulation code which incorporates Read–Shockley energies, isotropic mobilities, and Euler angle description of orientations. Investigate the effect of grain growth on the MDF of an initial random texture.

The CSL description of grain boundaries is a geometric model based on the fact that for certain misorientation rotations, a fraction $1/\Sigma$ of atomic lattice sites will be coincident. The resulting boundary is termed a CSL boundary and is characterized by Σ . Exact CSL boundaries are seldom observed in general materials because the CSL relationship requires three independent boundary degrees of freedom to assume particular values. Brandon [19] introduced the concept of an acceptance criterion, which admits a wider range of CSL boundaries. Misorientations with angular deviations of less than $\Delta\theta\Sigma = 15^\circ/\Sigma^{1/2}$ from the true CSL misorientation are assumed to be within the Σ CSL.

The Read–Shockley derivation requires that the array of boundary dislocations be spaced uniformly by some multiple of the Burgers vector \mathbf{b} . A CSL boundary can be viewed as a secondary array of dislocations with spacing \mathbf{b}/Σ imposed on this primary array. As such, the contribution to the grain boundary energy from the CSL can be modeled:

$$\delta J_\Sigma = \begin{cases} -\frac{J_0}{\Sigma} \left\{ 1 - \frac{\delta\theta}{\delta\theta_\Sigma} [1 - \ln(\frac{\delta\theta}{\delta\theta_\Sigma})] \right\} & \text{for } \delta\theta < \delta\theta_\Sigma \\ 0 & \text{for } \delta\theta \geq \delta\theta_\Sigma \end{cases} \quad (3.34)$$

where $\delta\theta$ is the misorientation angle between the true CSL rotation and the actual grain boundary rotation, $\delta\theta_\Sigma$ parameterizes the width of the energy well and is given by the Brandon criterion, and J_0 determines its depth which maintains consistency with non-CSL boundary model in equation (3.29). The total energy of a boundary in our system is thus given by the sum of equations (3.29) and (3.34). Note that for non-CSL boundaries, the contribution from equation (3.34) is zero. Figure 3-38(a) shows the form of such an energy function.

This type of simulation shows some interesting differences between modeling the full 3D crystallographic orientations of a crystal and the 2D crystallographic orientation. In the latter case each grain requires only a scalar index to denote its orientation, and the misorientation θ is then easily calculated as a sum. In such a system, energy cusps of CSL as shown in Figure 3-38(a) have a profound effect on grain growth with the MDF produced mirroring the energy function and a large fraction of the boundaries to forming multijunctions as shown in Figure 3-38(b). However, if the same simulations are carried out in which each grain requires three Euler angles to denote its orientation, then the evolution is very different. The extra degrees of freedom that exist in Euler space mean that the chances of forming a boundary within the CSL limit become much smaller. The vast majority of boundaries that are classified as CSLs do not have significantly reduced energy; in fact, 95% of nominal $\Sigma 5$ boundaries have energy within 5% of the random, high angle boundary energy. Even if the misorientation angle of the generic CSL is close to that of the exact CSL, the axis need not be close to the true axis. Therefore, most nominal CSL boundaries have energy near that of non-CSL boundaries and should not be morphologically enhanced during grain growth (for more information read ref. [HHM03]).

PROBLEM 3-35: The Effect of CSL Boundaries on Grain Growth

Write an anisotropic Potts model simulation code which incorporates Read–Shockley energies, CSL boundaries, and isotropic mobilities. Show that when the orientations of the grain are denoted by a scalar quantity, grain growth increases the number of CSLs in the system. Also show that this does not happen when the grain orientations are represented by Euler angles.

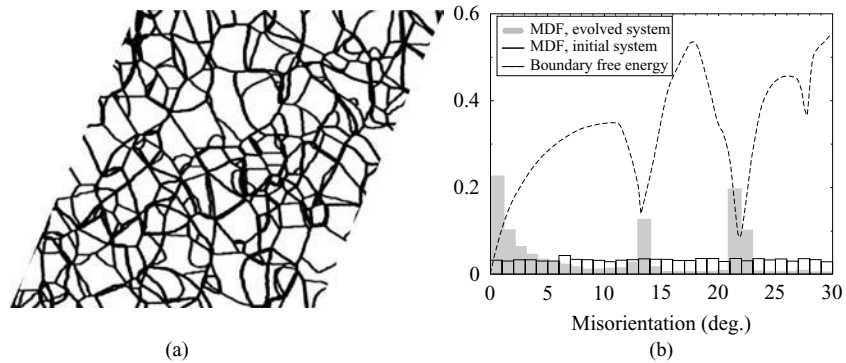


FIGURE 3-38 Potts model simulation of anisotropic grain growth, (a) 2D microstructure growth showing the multijunctions that form with highly anisotropic energy functions, (b) showing the relationship between MDF of the evolved system and the energy function.

3.5.6 Abnormal Grain Growth

Abnormal grain growth might be viewed as anisotropic grain growth gone wrong. Systems that exhibit abnormal grain growth can also exhibit the self-organizing scaling characteristics of grain growth as illustrated in the previous section. But under certain conditions the scaling breaks down and a small percentage of grains grows abnormally to consume the microstructure. Where these abnormal grains come from, or what causes them to form, is still an open research question, but what is clear is that by its very nature, abnormal grain growth is a statistical process. One cannot point to a particular grain and determine whether it will grow abnormally; rather one can predict a probability that it will do so. The reason for this unpredictability is that the growth of a grain depends also on the growth of its neighbors, which in turn depends on their neighbors and so on. This means that simulations of abnormal grain growth need to be large enough to capture these rare events. The rarer the event the larger the simulation must be. The alternative to performing very large simulations is to carry out many smaller simulations. Either way we need to be careful to statistically sample the system.

Figure 3-39 shows the time evolution of anisotropic system with Read–Shockley energies and anisotropic mobilities. The gray scale of the boundaries is proportional to their misorientation, thus the darker boundaries have higher misorientation. Initially, the grains undergo normal growth in which they remain relatively uniform in size, but the average boundary misorientation decreases. However, in this particular simulation one subgrain begins to grow discontinuously. This subgrain has an orientation far from the mean, so that the mean misorientation of its boundaries is larger than average, as indicated by its light boundaries in Figure 3-39. As it grows, its high misorientation boundaries come to dominate the system, and the average boundary misorientation increases. In texture space, we observe an initial tightening of texture during normal growth, followed by a shift to a new texture as the abnormal grain envelopes the system.

Note that only 10% of simulations exhibited such abnormal behavior; the remainder of the simulations displayed only normal growth. Should we expect such abnormal behavior from this system? How do we know whether the model is predicting the probability of such events correctly?

Our starting point to answer this question is the mean field theory of Rollett and Mullins [RM97], from which we can predict the growth rate of the abnormal subgrain from and compare it with its actual growth rate, as measured in the simulation. Figure 3-40 shows the results.

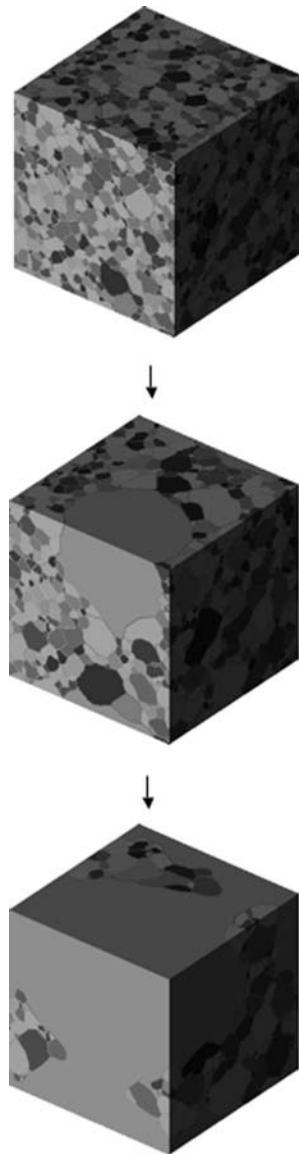


FIGURE 3-39 *The evolution of microstructure during a Potts model simulation of anisotropic grain growth of a single texture component, using Read–Shockley energies and anisotropic mobilities. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 1.0$.*

At early times, stochastic effects cause the data to be quite noisy, and at late times when the abnormal subgrain has consumed much of the system, finite size effects become important. However, the agreement is good at intermediate times, indicating that the abnormal subgrain is growing as predicted by theory. This comparison with theory tells us that when abnormal subgrains appear they behave as they should, but theory does not tell us how many such events

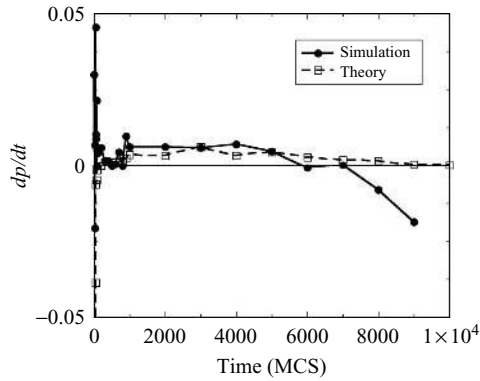


FIGURE 3-40 Time rate of change in the size of an abnormally growing subgrain normalized by the mean size of the normal subgrains. The solid line is for the abnormal grain shown in Figure 3-39, and the dashed line gives the results of the analytical theory of Rollett and Mullins [RM97] for the same subgrain. At intermediate times, the agreement between theory and simulation is excellent, indicating that the abnormal growth is driven by an energy and/or mobility advantage, as predicted by the theory.

we can expect. Is the observed occurrence of abnormal events in 10% of the simulated structures reasonable?

One of the virtues of computer simulation is the ability to decouple physical parameters to help determine underlying mechanisms. In the simulations of abnormal subgrain growth discussed previously, both boundary energy and mobility varied with misorientation, and no stress or surface effects were included. Thus, boundary energy, mobility, or both are the motive forces for abnormal growth. To determine which, we can perform simulations with uniform boundary mobility and Read–Shockley energies, in which case abnormal growth is not observed. If we run simulations with a gently varying mobility function and Read–Shockley energies, again, no abnormal growth occurs. But when we restore the anisotropic exponential mobility function and made the boundary energy constant and uniform; abnormal growth occurs as before. Therefore, we conclude that to observe abnormal growth in these highly textured systems, it is necessary and sufficient to have some very high mobility boundaries available to the system. With this knowledge, we can develop a model for the abnormal growth process and predict nucleation rates which can be compared with experiment. For more information read ref. [HMR03].

Something else that a simulation is able to offer is to compare the evolution of the same system under different conditions. For instance we can investigate the effect of texture gradient on grain growth using the same MDF and energy and mobility functions. Figure 3-41 shows an example of such a simulation which shows an abnormal grain growing rapidly through a texture gradient until it meets grains of a similar orientation, at which point it slows down and normal grain growth occurs.

PROBLEM 3-36: Abnormal Grain Growth

Write a 3D anisotropic Potts model simulation code which incorporates Read–Shockley energies and binary mobilities in which $M = 1$ for $\theta < \theta^*$ and $M = 1000$ for $\theta > \theta^*$. Simulate grain growth of equiaxed structures with strong single component textures exploring the effect of θ^* on the occurrence of abnormal grain growth.

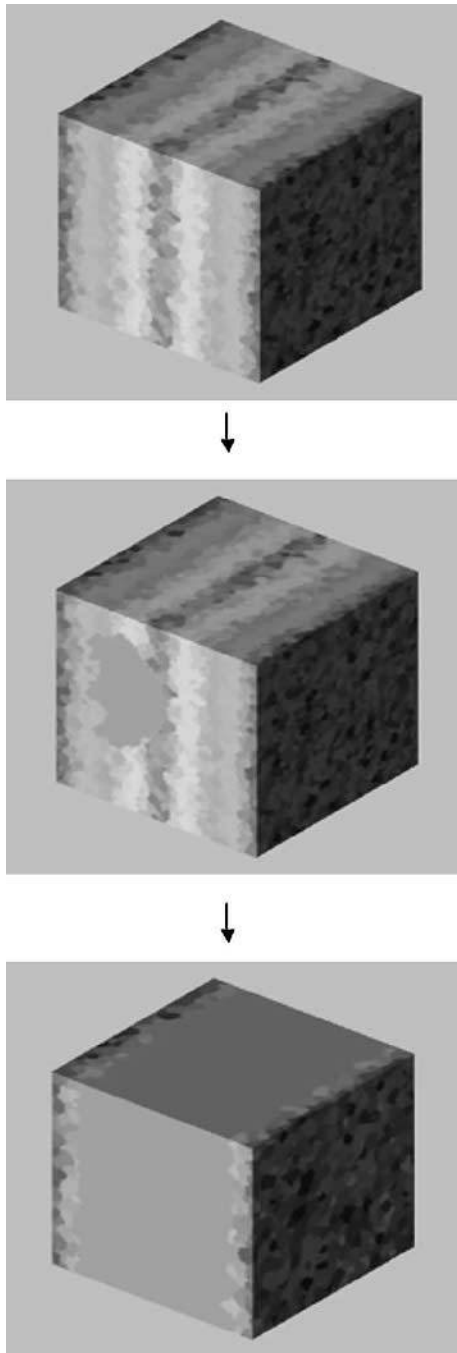


FIGURE 3-41 The evolution of microstructure during a Potts model simulation of anisotropic grain growth in a texture gradient, using Read–Shockley energies and anisotropic mobilities. The simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.5$.

PROBLEM 3-37: The Effect of System Size on Abnormal Grain Growth

Rerun simulations carried out in the previous exercise for different simulation volumes, for example, 50^3 , 100^3 , and 200^3 . Does the nucleation rate change with simulation size?

3.5.7 Recrystallization

Figure 3-42 shows an example of a 3D Potts model simulation of recrystallization in an extruded aluminium rod. There are many important issues that need to be dealt with to model recrystallization correctly. Clearly the microstructure, texture, and misorientation distribution are important, as is the overall macroscopic shape of the sample as shown in Figure 3-42. But perhaps the most important variable is the stored energy and its correlation with the microstructure. This is not only because the stored energy is the main driving force of recrystallization but also because the nature and density of the stored energy determines the nucleation rate.

Stored energy is the main driving force in recrystallization, but it is generally not easy to measure. More recently the pattern quality of EBSD Kikuchi images has been used as an index of work hardening state and hence dislocation density on a local scale. At best this is an accurate qualitative indication of the distribution of stored energy in the system. It is also of necessity a 2D measure. Figure 3-43 shows an example of an EBSD map in which the pattern quality measure is superimposed onto the grain structure. When importing these structures, the pattern quality is turned into the scalar variable, h_i , describing the local stored energy. An alternative way of modeling the distribution of stored energy is to perform simulations of plastic deformation and take the output of these simulations as the input for the Potts model simulations. These polycrystal plasticity simulations are notoriously difficult to perform accurately but they do have the advantage that variables that describe the deformed state are known. Once again these can be converted into the scalar quantities, h_i , needed by the Potts model.

Once the structure has been imported into the model, the next problem concerns the nucleation criteria to produce strain free nuclei in the structure. The simulations shown in Figure 3-42 were performed with site saturated surface nucleation, which means that all the nuclei were allocated to sites on the surface at the beginning of the simulation, and recrystallization occurred as a result of growth and competition between these nuclei. The precise nucleation model is extremely important in these simulations since it dictates the final grain size and the texture of the simulations. Thus great care must be taken when implementing and interpreting the nucleation model.

The growth part of the model, which really is the only part of such simulations that concerns the Potts model, really affects only the kinetics of grain growth. We have mentioned before that there are no absolute length and timescales in Potts model simulations. Determining critical length and timescales is not usually a problem, although this cannot be done *a priori*, but must be performed by some calibration procedure, which involves measuring the starting grain size in sites and equating this to the imported grain size measured in microns. The annealing time can be calibrated in a similar manner. However, users of the Potts model should beware that there is a difficulty with kinetics of the simulations. As mentioned in Section 3.3.6, the driving force for growth is not proportional to the stored energy; the boundaries are all driven forward with equal force. Thus the kinetics are unlikely to be accurate even when the calibrations have been carried out unless boundary velocity is independent of driving force. Rollett and Raabe have proposed a hybrid model of recrystallization which involves the coupling of the Potts model with a cellular automaton (CA) to get around this problem [RR01]. This model is ideal if the aim is to model recrystallization and the subsequent grain growth in a single simulation. But in general CAs are generally more appropriate for modeling recrystallization than MC models. For more information read reference [RR01].

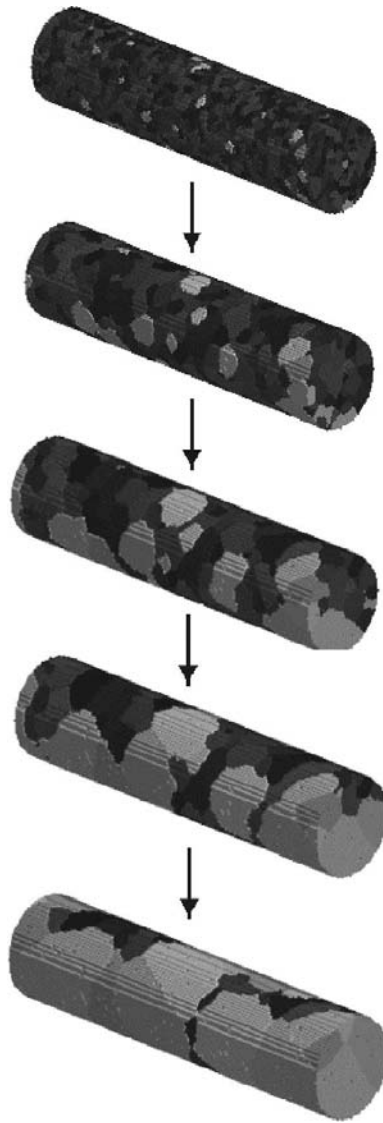


FIGURE 3-42 A series of snapshots from a simulation of Potts model simulation of an extruded aluminium rod recrystallizing with site saturated surface nucleation. The light gray indicates a recrystallized grain, and the dark gray grains are unrecrystallized. The system is a $50 \times 50 \times 200$ cylinder, with periodic boundary conditions in the axial (z) direction [HB01].

3.5.8 Zener Pinning

Zener pinning is a specific case of the general phenomenon of boundaries sticking to surfaces which you see work in the meniscus of a glass of water. When a boundary comes in contact with a surface, it is removed, and since recreation of this boundary requires energy, the phenomenon is often associated with pinning forces. This is why a drop of water does not bounce off a windscreen, but rather sticks to it. In metals the same thing happens between grain boundaries

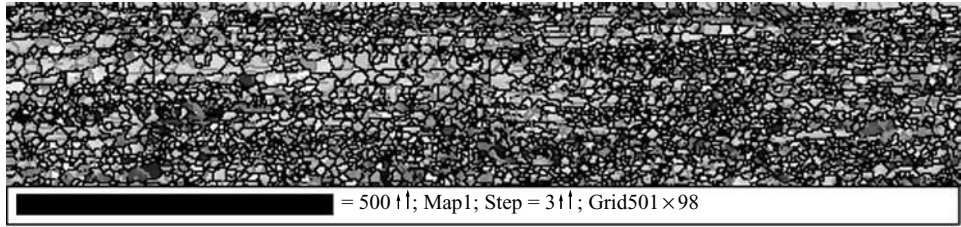


FIGURE 3-43 Input microstructure from EPSD for a recrystallization simulations. The gray scale is indicative of the stored energy in the sample [HMR07].

and particles. On contact part of the boundary is removed and thus to pull away from the particle requires energy to recreate this boundary. Figure 3-44 shows the shape of a boundary being pulled away from a particle in three different systems: a soap bubble, a TiN particle in steel, and the Potts model. As mentioned before this is a 3D phenomenon entirely, and different physics applies in the 2D case.

The catenoid shape that is created when trying to pull a boundary off a particle is characteristic of Zener pinning, and as we can see from Figure 3-44 the Potts model appears to work well. However, it is not enough for the model to appear to give the right shape; we need to quantify the correspondence.

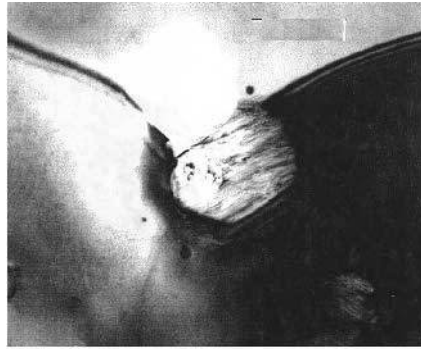
If the particle is incoherent with the matrix and the interface tension γ at the boundary–particle interface is isotropic, this results in an angle of $\pi/2$ at the boundary–particle interface. Under these circumstances the boundary exhibits a characteristic dimple shape, which is a minimal surface described as a catenoid of revolution. Hellman and Hillert derived the catenoid boundary shape as a function of z_0 , the distance between the particle center and the spherical boundary cap [see Figure 3-45(a)]:

$$z_0 = a \left[b + \cosh^{-1} \left(\frac{\rho}{a} \right)^{1/2} - \frac{\rho}{a} \left(1 - \left(1 - \frac{a}{\rho} \right)^{1/2} \right) \right] \quad (3.35)$$

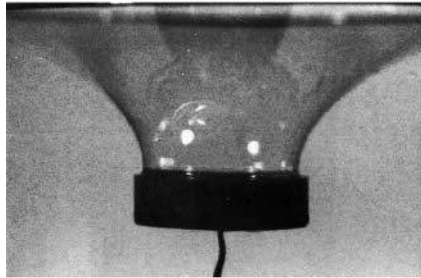
where θ is the bypass angle, r is the particle radius, and ρ the radius of the curvature of the boundary, $a = r \sin \theta \cos \theta$ and $b = \sec \theta \cosh^{-1}(\operatorname{cosec} \theta)$.

Figure 3-45(b) shows the comparison between the Potts model and the theoretical predictions. As can be seen they are very close, but we can go further than this and show the correspondence over the whole range of bypass angles by plotting the pinning force. The pinning force can be calculated by measuring the derivative of the energy with distance traveled by the boundary. The only complication with this is knowing the distance of the boundary, since locally it changes shape dramatically. Thus these calculations require large simulations to be carried out in which the local dimple is a small perturbation on the overall boundary shape. Figure 3-45(c) shows the normalized force (F/F_p) versus normalized displacement (z_0/r) for the model and theory. Note that boundary detachment occurs at a bypass angle of 70° , and this means that although the maximum pinning force is independent of ρ/r , the point of boundary detachment, the “stickiness” of a particle–boundary interaction, is not.

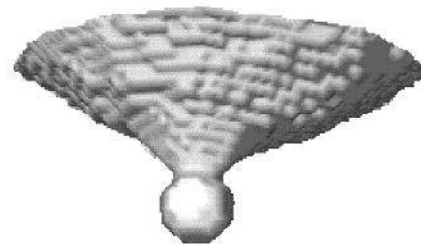
What these simulations show is that the Potts model gives good agreement with the theoretical curve although there is disagreement with the details of boundary detachment. This gives us confidence to scale up the model and look at the effect of arrays of particles on grain growth. Figure 3-46(a) shows a snapshot of a simulation carried out to investigate the effect of volume



(a)



(b)



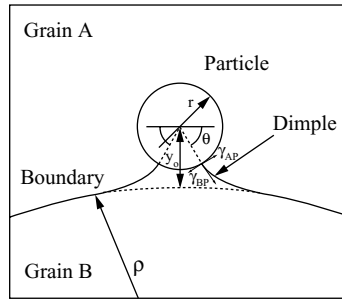
(c)

FIGURE 3-44 The boundary–particle interaction, (a) TiN particle interacting with carbon steel grain boundary, (b) soap film interacting with a particle [AHL69], (c) the simulation was performed using a square (1,2) lattice, Glauber dynamics, Metropolis transition probability function, and $kT_s = 0.5$.

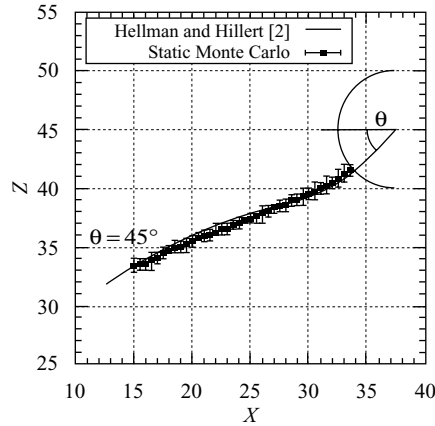
fraction of particles on grain growth. By carrying out a systematic study of the effect of volume fraction on pinned grain size, it is possible to plot the relationship and compare with experiment. Figure 3-46(b) shows that the Potts model does a good job of describing the phenomenon. For more information read ref. [HHCM06].

PROBLEM 3-38: Zener Pinning

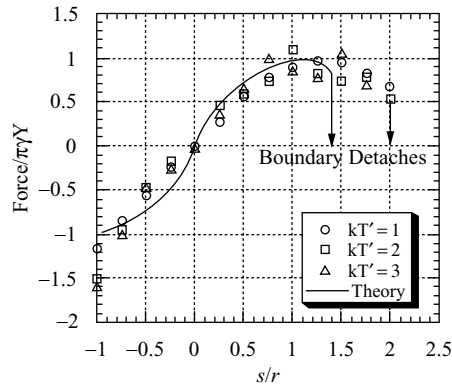
Write a 3D Zener pinning Potts model simulation code which simulates grain growth in the presence of cube-shaped particles (3^3 sites) for volume fraction = 0.1. Investigate what happens when the particles are dissolved from the simulation.



(a)



(b)



(c)

FIGURE 3-45 (a) A schematic of the formation of a dimple during grain boundary bypass of a particle. r is radius of the particle, ρ is the radius of boundary curvature, θ is the boundary bypass angle, γ is the boundary surface tension, and γ_{AP} and γ_{BP} are the two particle–boundary surface tensions, and y_0 is the distance of the boundary from the particle center, (b) comparison of the dimple shape produced by a Potts model and theory, (c) comparison of the pinning force produced by a Potts model and theory.

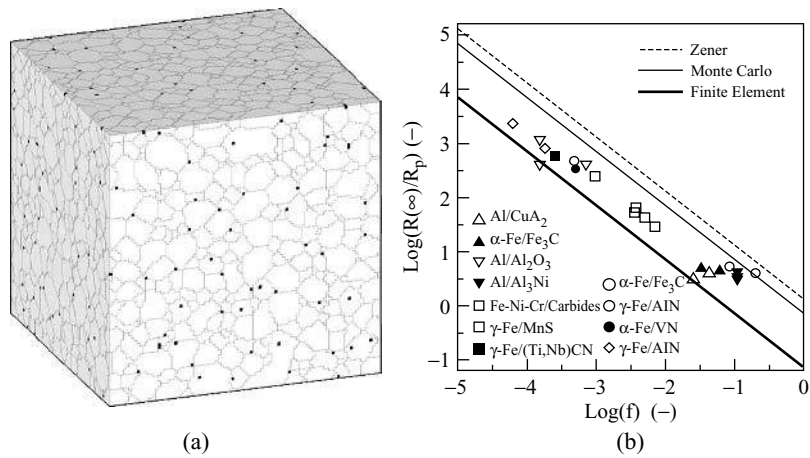


FIGURE 3-46 (a) Snapshot of a pinned microstructure in a Potts model simulation of Zener pinning on a $400 \times 400 \times 400$ lattice, using particles with sizes $3 \times 3 \times 3$; (b) Comparison of pinned grain size with experimental data [CDMF05].

3.6 Summary

In this section we have explored several applications of the Potts model. By the end of this section the reader should have experience with coding anisotropic grain growth models including realistic textures and MDFs, recrystallization models, and Zener pinned systems. We have left out many other applications and the reader is invited to explore others, such as the coupling of a Potts model with finite element models or its application to biological systems. Although the model has its drawbacks, we have seen through careful analysis that it captures the physics of grain growth, and related phenomena such as Zener pinning, very well. At the moment for large 3D systems, with complicated textures and pinning phases, the model has no equal.

3.7 Final Remarks

The aim of this chapter has been to give an introduction to the Potts model and a practical guide to programming and performing simulations. The beauty of the Potts model is that it is a simple way to model complex systems by modeling local physics (in particular, the effect of surface tension phenomena on the development and evolution of microstructure.) We have investigated at some length the reason why the model captures the physics of curvature driven growth accurately through a mechanistic exploration on the movement of kinks and ledges. At all times we have been concerned with comparing the model with theory and experimental data; this instinct is essential to any modeler. It is easy to make a model yield pretty pictures that appear to have a correspondence with a “real” phenomenon, but quantification of the simulation is the only way to use the model as a method to gain physical insights and understanding. Finally it is important to note that a model is just that, a model, and the benefit of it is as much to guide experiments and to hone the intuition about physical phenomena as it is to make predictions. The major role of computer models is to reduce the number of experiments that need to be carried out and to highlight what variables are key to understanding the results. The Potts model should be seen in this light; it is a guide to the intuition, and above all it is a medium of communication between experimentalists and theoreticians.

3.8 Acknowledgments

The author of this chapter would like to thank Elizabeth Holm, Tony Rollett, and other co-workers, past and present, for their fellowship, insight, and discussions.

Bibliography

- [AGS85] M. P. Anderson, G. S. Grest, and D. J. Srolovitz. *Scripta Met.*, 19:225–230, 1985.
- [AHL69] M. F. Ashby, J. Harper, and J. Lewis. *Trans Met. AIME*, 245:413, 1969.
- [BKL75] A. B. Bortz, M. H. Kalos, and J. L. Liebowitz. A new algorithm for Monte Carlo simulation of Ising spin systems. *J. Comp. Phys.*, 17:10–18, 1975.
- [BT52] J. E. Burke and D. Turnbull. Recrystallization and grain growth. *Prog. Metal Phys.*, 3:220–292, 1952.
- [CDMF05] G. Couturier, R. Doherty, C. Maurice, and R. Fortunier. *Acta Mater.*, 53(4):977–989, 2005.
- [HB01] E. A. Holm and C. C. Battaile. *JOM*, 53:203, 2001.
- [Her49] C. Herring. *Surface Tension as a Motivation for Sintering*. McGraw-Hill, New York, 1949.
- [HH93] G. N. Hassold and E. A. Holm. A fast serial algorithm for the finite temperature quenched Potts model. *Computers in Physics*, 7:97–107, 1993.
- [HHCM06] A. Harun, E. A. Holm, M. P. Clode, and M. A. Miodownik. On computer simulation methods to model Zener pinning. *Acta Mater.*, 54:3261–3273, 2006.
- [HHM01] G. N. Hassold, E. A. Holm, and M. A. Miodownik. The influence of anisotropic boundary properties on the evolution of misorientation distribution during grain growth. *Acta Mater.*, 49:2981–2991, 2001.
- [HHM03] G. N. Hassold, E. A. Holm, and M. A. Miodownik. On the accumulation of CSL boundaries during grain growth. *Materials Science and Technology*, 19:683–687, 2003.
- [Hil65] M. Hillert. On the theory of normal and abnormal grain growth. *Acta Metall.*, 13:227, 1965.
- [HMR03] E. A. Holm, M. A. Miodownik, and A. D. Rollett. On abnormal subgrain growth and the origin of recrystallization nuclei. *Acta Mater.*, 51:2701–2716, 2003.
- [HMR07] Y. Hu, M. A. Miodownik, and V. Randle. *Materials Science and Technology*, 2007. In Press.
- [HSC93] E. A. Holm, D. J. Srolovitz, and J. W. Cahn. Microstructural evolution in two-dimensional two-phase polycrystals. *Acta Metall. Mater.*, 41:1119–1136, 1993.
- [Jan03] K. G. F. Janssens. Random grid, three dimensional, space-time coupled cellular automata for the simulation of recrystallization and grain growth. *Mod. & Sim. In Materials Sci. Eng.*, 11(2):157–171, 2003.
- [MGHH99] M. A. Miodownik, A. Godfrey, E. A. Holm, and D. A. Hughes. On boundary misorientation distribution functions and how to incorporate them into three-dimensional models of microstructural evolution. *Acta Mater.*, 47:2661–2668, 1999.
- [MSG02] M. I. Mendeleev, D. J. Srolovitz, G. Gottstien, and L. S. J. Shvindlerman. *Mater. Res.*, 17:234, 2002.
- [Mul56] W. W. Mullins. Two-dimensional motion of idealized grain boundaries. *J. Appl. Phys.*, 27:900–904, 1956.
- [RM97] A. D. Rollett and W. W. Mullins. *Scripta Mater.*, 36:975, 1997.
- [RR01] A. D. Rollett and D. Raabe. A hybrid model for mesoscopic simulation recrystallisation. *Computational Materials Science*, 21:69–78, 2001.
- [RS50] W. T. Read and W. Shockley. *Phys. Rev. B*, 78:275, 1950.
- [SB95] A. P. Sutton and R. W. Balluffi. *Interfaces in Crystalline Materials*. Oxford Science Publications, Oxford, 1995.
- [vN52] J. von Neumann. *Metal Interfaces*, p. 108. ASM, Cleveland, 1952. (in discussion to Smith).
- [WGS02] M. Winning, G. Gottstien, and L. S. Shvindlerman. *Acta Mater.*, 50:353, 2002.
- [WPS+97] S. A. Wright, S. J. Plimpton, T. P. Swiler, R. M. Fye, M. F. Young, and E. A. Holm. Potts-model grain growth simulations: Parallel algorithms and applications. Technical Report Sandia Report SAND-97, Sandia National Laboratories, 1997.

4 Cellular Automata

—Koen Janssens

4.1 A Definition

In a generalized form cellular automata are defined by the following minimal description:

1. n -dimensional space is *partitioned* into a discrete subset of finite n -dimensional volumes, which are named *cells*.
2. A *state* is assigned to each cell.
3. At every time step, a local *neighborhood* is defined for each cell.
4. A *state change rule* is defined, which computes the new state of a cell as a function of the state(s) of all cell(s) in the local neighborhood of that cell.
5. A cellular automata simulation proceeds by discrete simulation steps—hence discrete time steps—consisting of the computation of the new states of all cells based on their current states.

Note that this definition is less restrictive than the one usually implemented. First, the partitioning is not required to be regular grid based; in other words not all cells in a simulation need to have the same shape and/or size. This allows for specialized cellular automata and adaptive cell shape and size distributions. Second, the number of possible states does not need to be finite. This allows for *continuous* states or states with continuous attributes. Third, the neighborhood definition may vary from cell to cell and from step to step, allowing for influence of local conditions on neighborhood definition and also for network-type cellular automata. Finally, a state change rule does not necessarily need be deterministic, which allows for probabilistic cellular automata. We will return to these more advanced types of cellular automata, but for now we return to the basics.

4.2 A One-Dimensional Introduction

As an example in one-dimensional space let us go through the description given previously:

1. Consider a straight line and partition it by defining a subset of line segments of finite length. To simplify things, define these segments with constant length.

2. Define the state of a cell as discrete binary values of either 0 or 1.
3. Define the neighborhood of a cell as itself and the two neighboring segments.
4. Define a state change rule as

$$\text{New State} \begin{cases} 0 & \text{if all cells in the neighborhood have the current state 0} \\ 1 & \text{if any cell in the neighborhood has the current state 1} \end{cases} \quad (4.1)$$

which can be depicted as in Figure 4-1.

To run this simple cellular automaton we only need to define how many time steps we want to compute, and what the initial distribution of cell states is. An algorithm could look like Algorithm 4-1. This kind of cellular automata can easily be computed using Mathematica [Mat], and the output is as presented in Figure 4-2.

ALGORITHM 4-1: An Algorithm for CA Rule 254

- 1: nsteps = 100
 - 2: $s[0 : 100][0 : 100]$ is a Boolean array {width \times nsteps + 1}
 - 3: $t[0 : 100]$ is a Boolean array {to temporarily store the new states}
 - 4: Set all cells to state 0
 - 5: Initialize $s[50][0] = 1$
 - 6: **for** $i = 1 \rightarrow 100$ **do**
 - 7: Compute new states using equation (4.1) and the cell states in row $i - 1$ and store in t
 - 8: Copy states from t to row i in s
 - 9: **end for**
-



FIGURE 4-1 Graphical depiction of a one-dimensional cellular automaton state transformation function. A shaded square has state 1, an unshaded state 0.

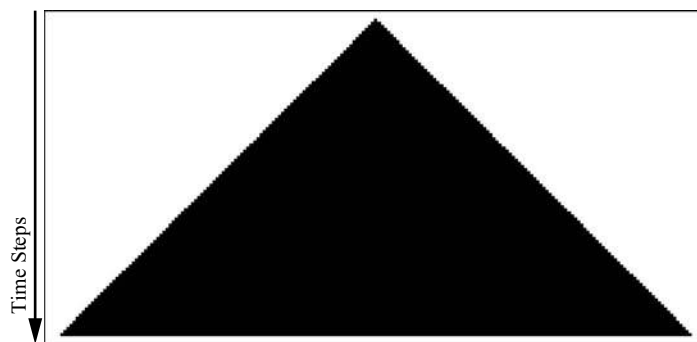


FIGURE 4-2 This example simulates the growth of a single, one-dimensional grain into a deformed microstructure using very basic cellular automata.

4.2.1 One-Dimensional Recrystallization

The preceding introductory example is really as basic as a cellular automaton can get. Surprisingly enough, we can already build a recrystallization model—admittedly it will be very much simplified and only one-dimensional, but nevertheless a model that demonstrates how cellular automata can be straightforwardly used to model microstructure evolution. The fact that it is one-dimensional also allows us to show you the timeline without turning to animated figures, which is rather difficult on regular paper.

Let us say state 0 represents plastically deformed material and state 1 represents recrystallized material. Algorithm 4-1 thus simulates the nucleation (line 5 in the algorithm) and growth of a single recrystallized grain in a deformed material, as shown in Figure 4-2—it is as simple as that.

Assume a one dimensional material with random position, site saturated nucleation, with a nucleation probability of 0.01 per cell volume. To simulate site saturated recrystallization we only need to replace the initialization in Algorithm 4-1 to setting the state at $s[i][0]$ for all i to state 1 with a probability of 0.01, resulting in a simulation result similar to the one shown in Figure 4-3. Note that we could keep track of the recrystallized grains simply by assigning different states to every recrystallized grain; that is, the state is equal to 0 for the deformed material, and equal to a grain identification number for the new grains.

PROBLEM 4-1: Multiple Grains in One Dimension

Actually, in the preceding one issue remains unsolved before we can keep track of grains separately. What is it? And how would you solve it?

4.2.2 Before Moving to Higher Dimensions

The example of one-dimensional recrystallization in Figure 4-3 is very limited in that all the grain boundaries move at the same velocity all of the simulation time. Knowing that cellular automata are discretized in space *and* time, how can we add velocity variations to the algorithm?

Velocity Variations with Time

Assume the deformed microstructure is recrystallized at an elevated temperature T . To do so one must bring the material to that temperature from the ambient temperature, say T_0 , and at the end of the process allow the material to cool down to that same temperature T_0 , as illustrated in Figure 4-4. Using the classic model for grain boundary motion we know that the velocity of the grain boundary is given by the following equation:

$$v = mp \tag{4.2}$$

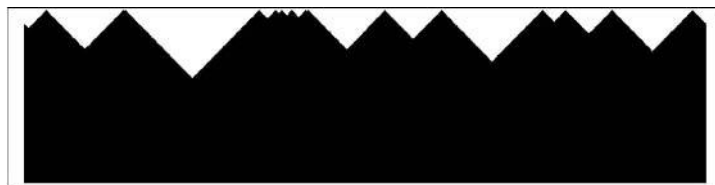


FIGURE 4-3 *One-dimensional recrystallization with site saturated nucleation using cellular automaton 254.*

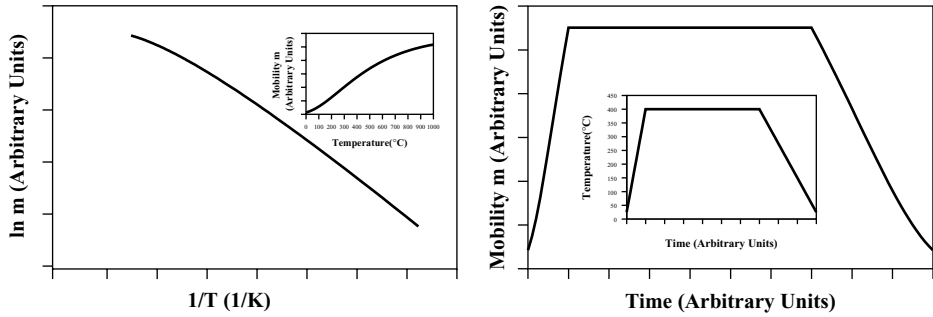


FIGURE 4-4 Mobility variation during an annealing cycle. The diagram on the left shows the classic Avrami model for the dependency of the grain boundary mobility on temperature. The diagrams on the right plot the typical temperature cycle and the mobility as a function of time.

Let us assume the plastic deformation to be constant over the whole specimen and that the driving pressure p is constant. We know that the grain boundary mobility depends on temperature. This dependence is usually assumed to follow an Avrami-type behavior:

$$m = \frac{m_0}{T} e^{-Q/kT} \quad (4.3)$$

Note that in literature one commonly neglects the $1/T$ dependence of the prefactor and includes it as a “minor temperature dependence” in m_0 . The mobility is typically plotted as $\ln m$ versus $1/T$ as shown in Figure 4-4. All this means is that we have to be able to vary the grain boundary velocity as a function of time in our cellular automata—or not?

Well, let us analyze this simplified case. If indeed the driving force p is constant and *only* mobility depends on temperature, then it must follow that the outcome of the simulation is *not* dependent on the exact temperature–time path. Instead of modifying the cellular automaton, we can also solve our problem by merely adapting the time step to cellular automaton-step scaling. If our one-dimensional cellular automaton is defined with a constant cell length equal to l_{CA} , we know that at each CA step a grain boundary moves over a distance l_{CA} :

$$l_{CA} = v\Delta t \quad (4.4)$$

and with equations (4.2) and (4.3) one can straightforwardly derive that

$$\Delta t = \frac{T l_{CA}}{m_0 p e^{-Q/kT}} \text{ with } T = f(t) \quad (4.5)$$

Rephrased: a single simulation tells us *where* grain boundaries go, equation (4.5) tells us when they will get there.

It should be no surprise that the example in the previous section is overly simplified and far from real microstructure processing. For the larger part of cases equation (4.5) will not work, the reason being that the driving pressure p and/or the mobility m show local variation, that is, variation from position to position in the microstructure.

One-Dimensional Recrystallization of a Heterogeneously Deformed Microstructure

In the example of a one-dimensional microstructure that has been deformed inhomogeneously, the driving pressure p for recrystallization will vary from position to position. It is clear from equation (4.2) that this means that grain boundaries will move at speeds varying throughout the microstructure. In the fully deterministic cellular automata model we presented so far such is impossible: the grain boundary velocity is constant as fixed by the cellular automaton grid: $v = l_{CA}/\Delta t$.

The most straightforward way to introduce the capability to model locally varying grain boundary velocities is to implement a cellular automaton with *probabilistic* state change rules, in short a “probabilistic cellular automaton.” As an example, we can change the state change rule given in equation (4.1) to a probabilistic version:

$$\begin{aligned} \text{current state} = 0 &\Rightarrow \text{new state} \begin{cases} 0 \text{ with probability } 1 - P \\ 1 \text{ with probability } P \end{cases} \\ \text{current state} = 1 &\Rightarrow \text{new state} = 1 \end{aligned} \quad (4.6)$$

The probability function P is computed locally for each cell based on its neighborhood. In our one-dimensional example P could be computed as follows:

1. Determine the maximum driving pressure p_{\max} occurring in the current state of the simulation.
2. Use p_{\max} and equation (4.5) to compute Δt for the current CA step.
3. Using the local value for p for each cell, compute the local value of P using

$$P = \frac{p_{\text{local}}}{p_{\max}} \quad (4.7)$$

4. Finally, use equation (4.6) in Algorithm 4-1 to simulate the recrystallization process.

You should now be able to program your first probabilistic cellular automaton. The results should look something like Figure 4-5.

Continuous Nucleation versus Site Saturation

The preceding examples all stick to site saturation for the nucleation of new grains in the deformed microstructure. Now let us consider the case in which nucleation is a continuous process in which the emergence of new nuclei per units of time and volume is dictated by a temperature dependent nucleation rate $\dot{n}(T)$. As this function is different from the temperature dependence of the grain boundary mobility m [equation (4.3)], equation (4.5) will not work for all cases. Before we can explain why, we must know how to introduce the nucleation process into the cellular automaton.

In the context of a discrete cellular automata model continuous nucleation means that per CA step a certain number of nuclei n_{CA} is generated. Assuming the CA model consists of m cells, and at a certain simulation step the remaining number of cells in the deformed state is m_d , then the real, immediate nucleation rate is

$$\dot{n}(T) = \frac{n_{CA}/m_d}{\Delta t l_{CA}} = \frac{\dot{n}_{CA}}{\Delta t l_{CA}} \quad (4.8)$$

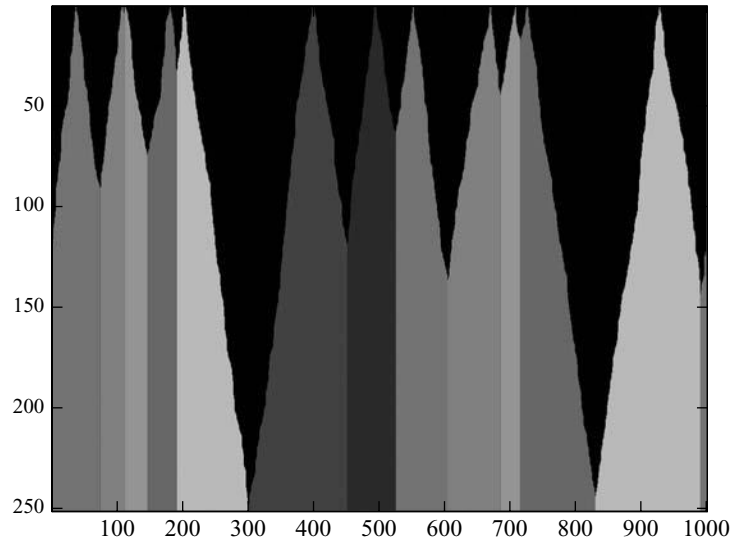


FIGURE 4-5 One-dimensional recrystallization with site saturated nucleation using a probabilistic cellular automaton, $p_{\text{local}}/p_{\text{max}} = 0.4$ and randomly located nuclei (2 in 100 cells are nuclei).

Important to note is that n_{CA} must be chosen between 1 and a number substantially smaller than m . Otherwise our CA grid remains empty or is filled with nuclei within few CA steps and we are no longer able to simulate the growth process. Also note that a nonprobabilistic cellular automaton can only approximate a continuously varying nucleation rate with a discrete and finite subset of nucleation rates. This is usually unacceptable, and it is common to implement a probabilistic nucleation model, in which \dot{n}_{CA}^P is defined as the probability per CA step and per cell that a nucleus is generated. This leads to the following relation:

$$\dot{n}(T) = \frac{\dot{n}_{\text{CA}}^P}{\Delta t l_{\text{CA}}} \quad (4.9)$$

In our one-dimensional CA-model equation (4.5) defines the relation between Δt and l_{CA} . Whether equation (4.8) or (4.9) is used as a nucleation model, in both cases we only have a limited range from which we can choose \dot{n}_{CA} or \dot{n}_{CA}^P to comply. Let us have a look at this problem in detail with a case study.

PROBLEM 4-2: One-Dimensional Recrystallization under Conditions of Continuous Nucleation

Program a one-dimensional cellular automaton, capable of simulating the continuous nucleation and growth of recrystallizing grains. Use a statistical nucleation model, and allow the model to have a table specify the temperature versus time.

The input variables to our problem are

- $T(t)$ Temperature [K] as a function of time. This function is, for example, made available in the form of a tabulated function, and specifies how long our simulation needs to run.
- $\dot{n}(T)$ The nucleation rate [$\text{mm}^{-1}\text{s}^{-1}$] as a function of temperature, also given as a tabulated function.

- $m(T)$ The grain boundary mobility m , specified by the parameters m_0 in [Kmm³/Ns] and Q in [J], and a function of the temperature T [K] as in equation (4.3).
- p Assume homogeneous deformation, specified by a constant driving pressure p in [N/mm²].

To simulate the process our algorithm must perform the top level functions specified in Algorithm 4-2.

ALGORITHM 4-2: Top Level Functionality of a One-Dimensional Recrystallization Simulation

- 1: Initialize the simulation by setting the number of cells nc and the number of CA steps ns to perform.
 - 2: Read input variables
 - 3: Determine Δt corresponding to a CA step.
 - 4: Determine a feasible value for l_{CA} using \dot{n} and total process time tps
 - 5: **for** $i = 1 \rightarrow ns$ **do**
 - 6: Perform nucleation step
 - 7: Perform growth step
 - 8: **end for**
 - 9: Ouput results
-

Now let us have a line by line look at the details of how to perform these functions:

Line 1: Reasonable numbers should be picked. For example, aiming at an average of 20 cells per recrystallized grain and expecting a total of 20 grains, we need at least 400 cells and 20 CA steps, so let us use 500 cells and 50 CA steps. Evidently, a better way to do things would be to test for a simulation to finish and to use a while-loop at line 5 instead of the current for-loop.

Line 3: As our driving pressure is constant we only need to determine the max mobility m_{\max} observed in the given temperature range (at the maximum temperature) to find the following relation:

$$\Delta t = l_{CA} \frac{T_{\max}}{m_0 p e^{-Q/kT_{\max}}}$$

This equation shows that in a cellular automaton space and time are coupled quantities, that is, if one selects a value for l_{CA} , one also selects the resolution of the time step Δt .

Line 4: We need a number for \dot{n}_{CA}^P so that we end up with an acceptable number of grains for the simulation size we set at line 1. Not taking into account growth, 50 CA steps with 500 cells result in a maximum of 25,000 “opportunities” to generate a nucleus, so one could start with $\dot{n}_{CA}^P = 0.001$ to end up with about 20 recrystallized grains. This leads us to a second relation between Δt and l_{CA} :

$$\Delta t = \frac{\dot{n}_{CA}^{P_{\max}}}{\dot{n}_{\max} l_{CA}}$$

in which $\dot{n}_{CA}^{P_{\max}}$ should be a useful number larger than zero but smaller than one. At this point it should be clear to you that, unless the time step Δt and/or the spatial resolution of the CA, as set by l_{CA} , is not constant, this puts limits on the range of parameter values that can be addressed in a cellular automaton:

$$0 < \dot{n}_{CA}^{P_{\max}} = \dot{n}_{\max} l_{CA} \Delta t = \frac{\dot{n}_{\max} T_{\max}}{m_0 p e^{-Q/kT_{\max}}} l_{CA}^2 \ll 1$$

Note: Although the part of the algorithm detailed so far is not really an essential part of a CA algorithm, it is worthwhile as it tells us whether the simulation we perform is correctly sized for the problem we want to analyze *before* we run the actual simulation. Evidently, this is overkill for the small problem we analyze in this example, but having the discipline to do so becomes valuable when more complex problems are attacked.

Line 6: A nucleation step consists of finding all the remaining cells still in state “deformed”, and changing the state of these cells to “recrystallized” with probability \dot{n}_{CA}^P . Note that in doing so one forces a nucleus to have the size of one cell. One can circumvent this problem by introducing a third state “nucleated” next to “deformed” and “recrystallized”, and modifying the growth step to handle this additional state properly.

Line 7: A growth step consists of locating all cells changing state from “deformed” to “recrystallized” using the state change rules specified in equation (4.1).

Now all that is left for you to do is to program this algorithm in your favorite language.

PROBLEM 4-3: Modification to Problem 4-2

Edit algorithm and source code of Problem 4-2 to include the situation in which a microstructure was deformed heterogeneously, resulting in a locally varying driving pressure p .

PROBLEM 4-4: Another Modification to Problem 4-2

Edit algorithm and source code of Problem 4-2 to handle a third state “nucleated”, so that nuclei do not automatically appear as fully recrystallized cells. The transition from state “nucleated” to “recrystallized” should be handled in the growth step. Watch out for cells that become “nucleated” and that do not grow in the next growth step. Do you allow them to be overgrown by neighboring grains or not? Is there a difference between simulation results making one or the other choice?

4.3 +2D CA Modeling of Recrystallization

The step from one-dimensional to two- or three-dimensional cellular automata involves one major issue. In 2D and 3D representations of microstructure, interfaces between neighboring grains are lines and surfaces and need to be approximated within the CA grid. Grain boundaries play an important role in the details of microstructure evolution, and an imperfect representation may lead to serious modeling problems. We will address these problems in the following sections.

Apart from these interface-representation problems the extension to +2D is straightforward and only requires substantial work on the programmer’s part. We will try to give the reader a good idea of the work involved, as the one-dimensional CA used so far is only helpful for educational purposes. Whether you choose 2D or 3D when you start writing your own cellular automaton, be aware that, depending on the problem to be analyzed, either can be the best choice, so it is always a good idea to provide your source code with the capabilities for both types of simulations.

4.3.1 CA-Neighborhood Definitions in Two Dimensions

Before we can turn our attention to neighborhood definitions, we must realize that a cellular automaton operates on a space-filling grid of cells. The most classic form is a periodic grid of squares or cubes. Other types of grids are possible, as illustrated in Figure 4-6.

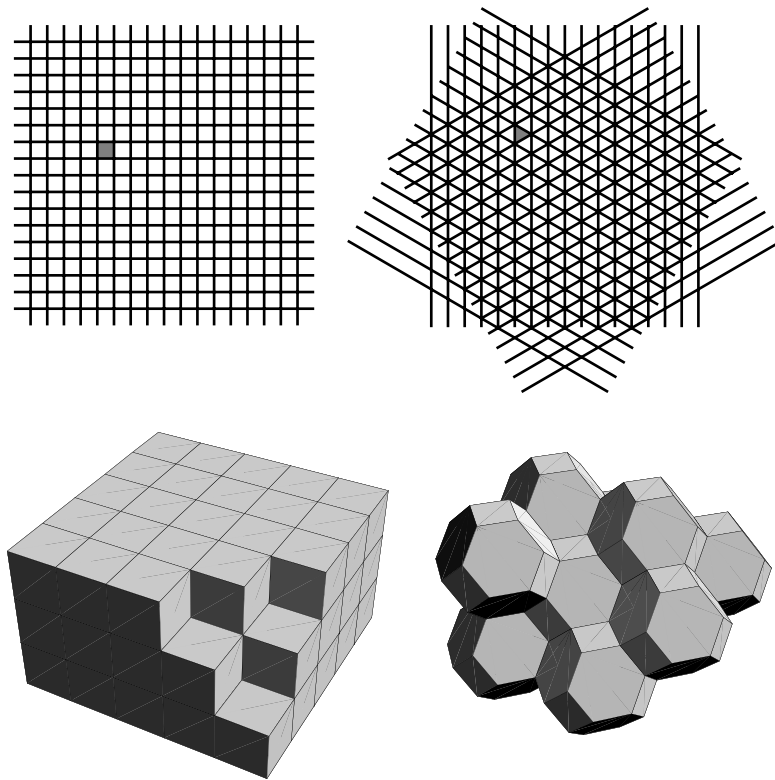


FIGURE 4-6 *Types of periodical cellular automaton grids in two and three dimensions.*

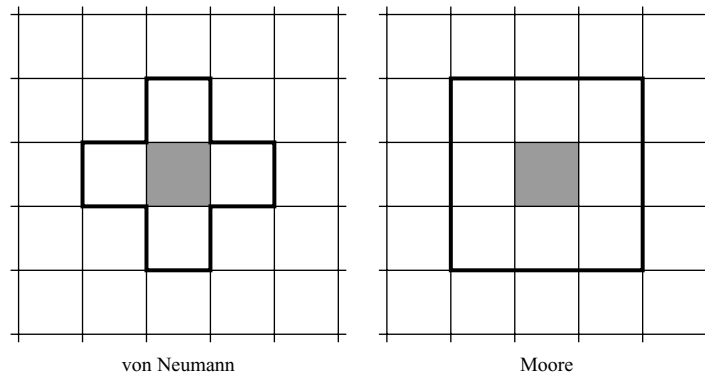


FIGURE 4-7 *The classic neighborhood definitions of von Neumann and Moore.*

von Neumann and Moore

Classic grids of cells come with classic definitions of neighborhood, and the most known are definitely the von Neumann and Moore neighborhood definitions on a 2D square grid of cells as illustrated in Figure 4-7. In these types of neighborhoods, of which a multitude of shapes can be imagined, a cell (shaded in the figure) is part of its own neighborhood. This is the usual case when using cellular automata in microstructure evolution modeling.

Margolus

The central idea of a Margolus-type neighborhood definition is that at different CA steps, different subzones of a total neighborhood are sampled (Figure 4-8). This allows for neighborhoods with a simplified shape, and thus simpler state-change rules. As described by Chopard and Droz [CD98], Margolus neighborhood types to some extent prevent long distance effects.

4.3.2 The Interface Discretization Problem

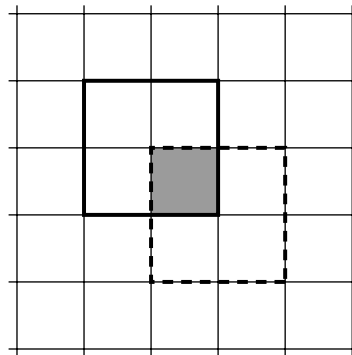
The easiest way to explain why a CA grid has difficulties with modeling lines and surfaces is by having the student program one without correction for this specific problem. Therefore, try Problem 4-5. The algorithm could look like Algorithm 4-3.

PROBLEM 4-5: A Simple 2D Cellular Automaton for Recrystallization

Using Algorithm 4-2 as top level, and forgetting about the relation between real time and CA time to simplify things, design and program a 2D algorithm that allows you to select different types of neighborhoods at run time, so that you can study the effect of the neighborhood definition on the simulation.

ALGORITHM 4-3: Top Level Functionality of a 2D Recrystallization Simulation

- 1: Initialize the simulation by setting the number of cells $nc = nc_x nc_y$.
 - 2: Define the cell state 0 for unrecrystallized material, and state 1 for recrystallized.
 - 3: Define the neighborhood, for example, von Neumann.
 - 4: Define the state change rule so that a cell that has a recrystallized cell in its neighborhood becomes (or remains) recrystallized itself at the next CA step.
 - 5: Assume site saturation, set a number of nucleation sites.
 - 6: **while** there are still cells with status 0 **do**
 - 7: Compute recrystallization step
 - 8: **end while**
 - 9: Output results
-



Margolus

FIGURE 4-8 The Margolus neighborhood definition. The neighborhood definition is different for each increment and switches between the definitions given by the full and the dashed line. The state transformation rule depends on relative location of the neighborhood to the focus cell (shaded in the figure). See ref. [CD98] for a more detailed description.

If one performs a simulation with an algorithm like Algorithm 4-3, the result should be similar to what is shown in Figure 4-9. Clearly, the grains grow with a geometry suspiciously similar to the shape of the neighborhood used to model the process. This is confirmed if one analyzes the distribution of the orientation of the grain boundary surfaces, as depicted in Figures 4-10 and 4-11.

As one can observe that the density of the cellular automaton grid and the reach of the neighborhood is varying with direction, one could naively come up with a probabilistic solution as depicted in Figure 4-12, showing a Moore-type neighborhood corrected for its shape to approximate a circle. The correction is such that a deformed cell still changes state to recrystallized when it finds a recrystallized neighbor, but only with a probability equal to the relative area of intersection with a circle as drawn in the top half of Figure 4-12. The bottom half illustrates what this means for the “shape” of the neighborhood (as it cannot resolve curvature within one cell). Applying this probabilistically corrected state transformation function, the shape of the growing grain is unfortunately not circular, and that independent of the resolution at which the simulation is performed, as illustrated in Figure 4-13.

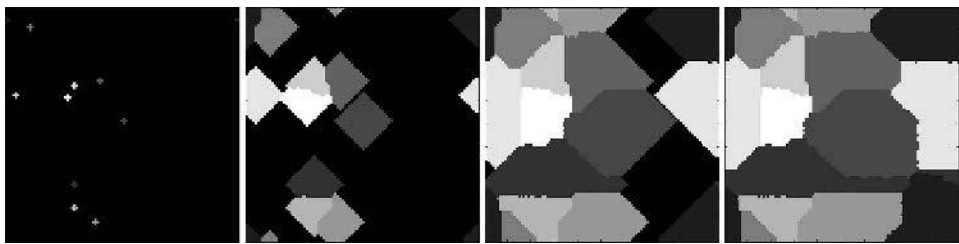


FIGURE 4-9 Illustration of the influence of the von Neumann neighborhood definition on a 2D CA-recrystallization simulation.

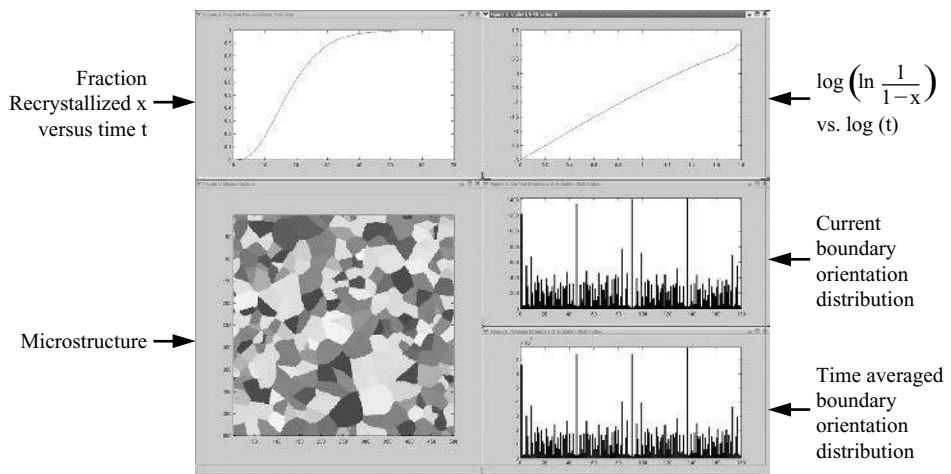


FIGURE 4-10 Evolution of the distribution of the grain boundary planes orientation during a recrystallization simulation performed with a von Neumann neighborhood definition. The orientations at 0° , 45° , 90° , etc are more frequently observed due to the grid on which the cells are placed.

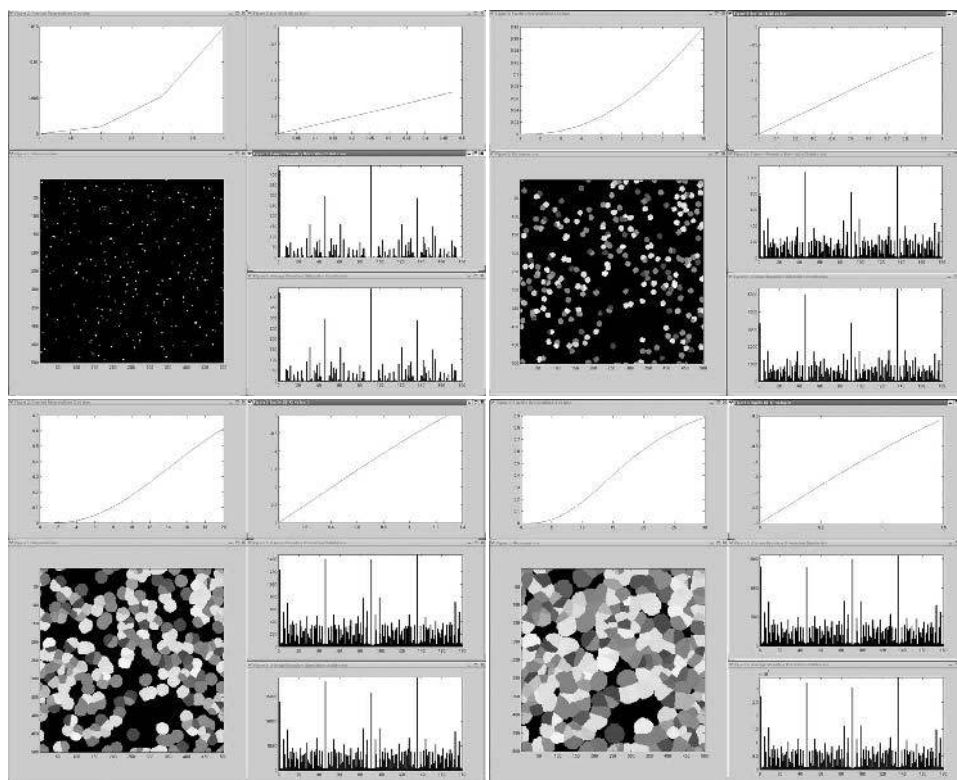


FIGURE 4-11 *Figure 4-10 continued.*

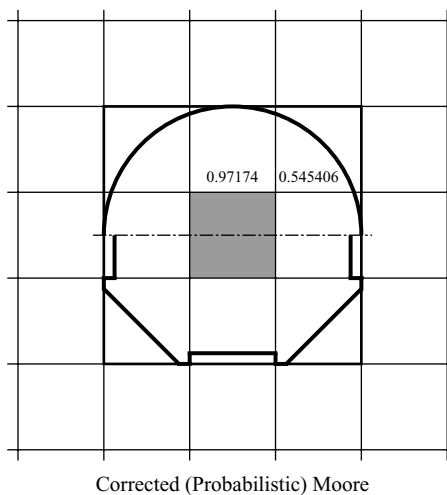


FIGURE 4-12 *A corrected Moore-type neighborhood and state transformation function.*

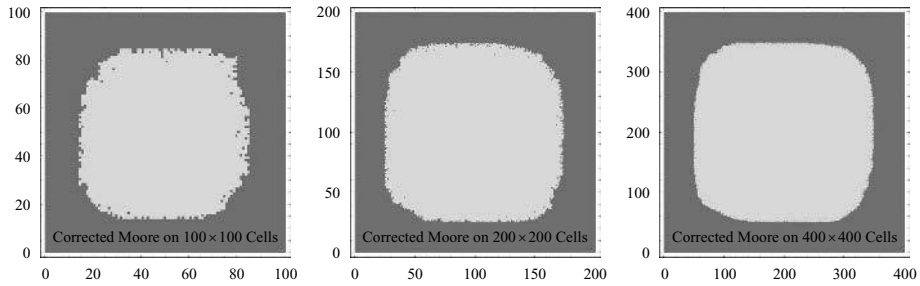


FIGURE 4-13 Application of the neighborhood and state transformation function as depicted in Figure 4-12 and for different cellular automaton grid resolutions.

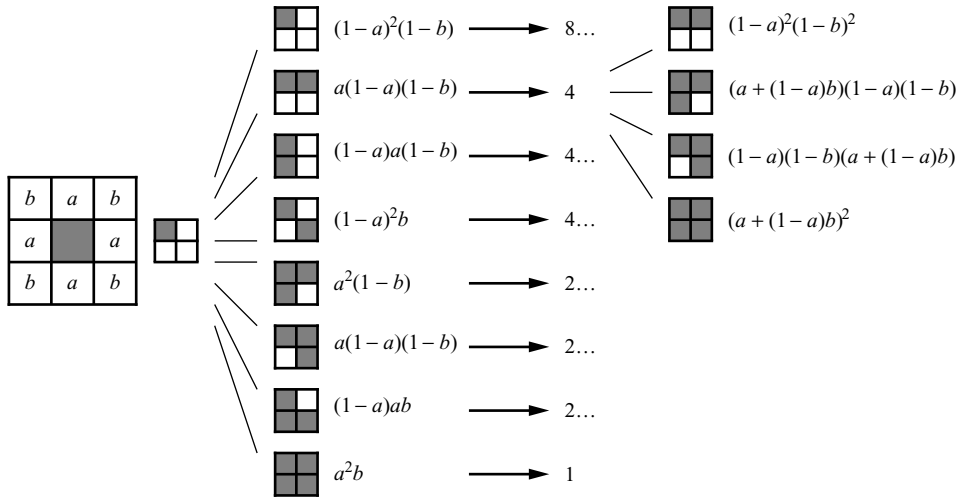


FIGURE 4-14 CA probabilities.

Now why is this intuitive solution not correct? Consider a 2D Moore-type neighborhood definition as shown in Figure 4-14, and define the state transformation function (with the state of cell i denoted by ξ_i) as

$$\begin{aligned} \xi_i(t) = 0 \rightarrow \xi_i(t + \Delta t) = 1 & \begin{cases} \text{with } P = a \text{ for each facing cell with } \xi = 1 \\ \text{with } P = b \text{ for each diagonal cell with } \xi = 1 \end{cases} \\ \xi_i(t) = 1 \rightarrow \xi_i(t + \Delta t) = 1 & \end{aligned} \quad (4.10)$$

Now consider a 2×2 grid of cells in which the top left cell has state 1 and the other three cells have state 0. Figure 4-14 partially lists the probabilities of different scenarios that can occur in two consecutive increments. Observe that the cell at the bottom-right corner of the 2×2 grid can obtain state 1 along different computational paths in the cellular automaton. To find the values of a and b with which state transformation function (4.10) leads to circular growth of the recrystallizing grain, the total probability that a cell obtains state 1 may only depend on its distance from the cell at which the grain was nucleated, and may *not* depend on

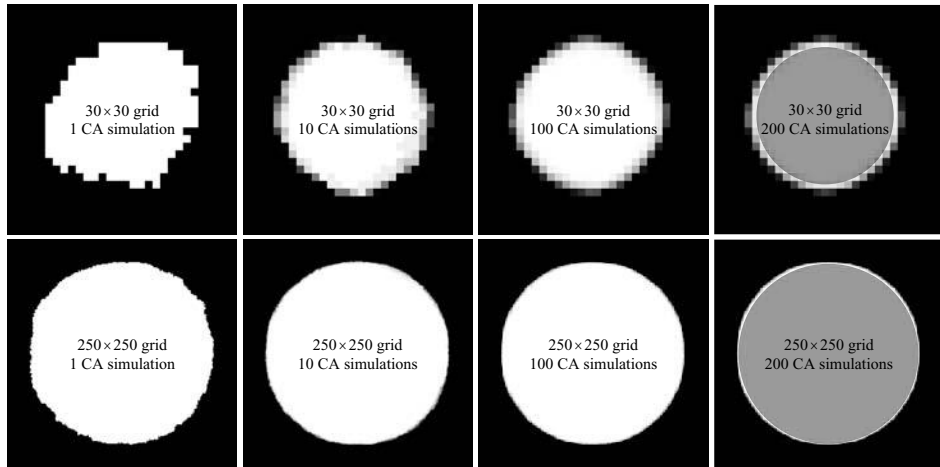


FIGURE 4-15 Application of the neighborhood depicted in Figure 4-12 but with a state transformation function as in Figure 4-13 (with $a = 0.8$ and $b = 0.25$). Different cellular automaton grid resolutions are shown, and from left to right the simulation result is averaged over an increasing number of cellular automaton simulations. In the last column a perfect circle is plotted in gray over the results.

the relative direction of its location. Figure 4-15 shows how close to a perfect circle one can get with trial and error. A closer observation also reveals that the edge of the circular grain in the averaged images shows a broader transition range in the diagonal direction. This is due to the fact that the grid density is lower in those directions. Other types of periodic grids, like the frequently used hexagonal one, may improve on this but will not completely solve the interface discretization problem. It should be emphasized that any computational method, including the ones addressed in other chapters in this book, are all influenced by the grid on which the simulation is performed.

PROBLEM 4-6: Hexagonal Cellular Automata

Design and program a cellular automaton that operates on a hexagonal grid (the cells are the six triangles subdividing a hexagon). The most straightforward neighborhood definition is a von Neumann-type one including four cells: the cell itself and its three facing neighbors. Note that the visualization of a nonrectangular grid is not so “easy,” as pixels are no longer “square.”

This interface discretization problem is a very serious one, and its impact only deepens once one realizes that the evolution of a microstructure, through grain boundary mobility and energy, depends on the *orientation* of exactly these lines or surfaces. Whether approximations with probabilistic corrections suffice to model microstructure evolution depends on the details of that evolution.

PROBLEM 4-7: Simply from a 2D to a 3D Cellular Automation

Did you remember to provide for 3D in your solution of Problem 4-5? Good! Now upgrade and see the influence of neighborhood definition in 3D space.

4.4 +2D CA Modeling of Grain Growth

One could argue that precise modeling of grain boundary surfaces, that is, making sure the computational results do not suffer all too much under the error on their mathematical description, is not all that important as the volume-based driving pressures in recrystallization and phase transformation are usually much larger than the energy stored in the grain boundaries themselves. Question remains whether that still holds for cases where the boundary orientation and curvature κ play a more important role. For example, in curvature driven grain growth the energy stored in the grain boundaries directly determines the evolution of the microstructure. This energy depends on the crystallographic nature—the misorientation—of the boundary, and consequently the error on their mathematical representation may be of increased relevance. In addition, mobility of grain boundaries is also known to depend both on the misorientation and the relative boundary plane orientation, so one should at least be careful not to make too rough an approximation. In this section grain growth is put forward as a case study.

The driving pressure for curvature driven grain growth can be written as a function of the curvature κ and the grain boundary energy γ :

$$p = \kappa\gamma \quad (4.11)$$

Locally approximating curvature using a sphere with radius R , one finds that $\kappa = 2/R$, that is, the grain boundary velocity locally computes to

$$v = m\kappa\gamma = m\frac{2\gamma}{R} \quad (4.12)$$

An example of a cellular automata algorithm that models curvature driven grain growth is shown in Algorithm 4-4.

ALGORITHM 4-4: 2D Cellular Automata for Grain Growth

```
1: Input or generate an initial microstructure
2: for  $n$  cellular automaton steps do
3:   Make a copy of the current cell states and name it previous microstructure {All
   computations in this step are made based on this copy}
4:   for Each Cell do
5:     if Cell is at a grain boundary then
6:       Compute local boundary curvature
7:       if Curvature  $\kappa > 0$  then
8:         Compute new state of cell
9:         {i.e. the probability that it switches grain}
10:      end if
11:    end if
12:  end for
13: end for
```

A cell is at a grain boundary if one of its neighboring cells belongs to another grain. Negative curvature induces the grain boundary to move away from the cell at which it is evaluated; therefore the state of that cell does not change and one only needs to proceed with the cells where the curvature is positive. Given a positive curvature one must compute the probability that its state changes to the identification number of (one of) the neighbor grain(s). To do so one must

first compute the maximum curvature κ_{\max} that can occur (or effectively occurs) in the next increment. Assuming isotropic grain boundaries, that is, the value of γ is constant, the probability that the boundary moves through the cell is

$$P = \frac{\kappa}{\kappa_{\max}} \quad (4.13)$$

4.4.1 Approximating Curvature in a Cellular Automaton Grid

To scale the cellular automaton to the underlying physics, one has to mathematically compute the local curvature based on the information available from the model. *How can one do that?* It should be obvious that one cannot directly use the faces of a cell, as these have no curvature in a periodic grid of cells. But can one approximate curvature based on the information in the local neighborhood of a cell at or near a grain boundary?

The Kink-Template Approach

This approach (e.g., see refs. [LLL06, Kre98]) does not bother with a separate construction of surfaces based on the CA grid; rather it directly approximates the curvature κ as illustrated in Figure 4-16. The algorithm used to compute the curvature is about as follows:

1. A *kink-template* neighborhood is defined. In the example in Figure 4-16 its definition includes the center cell for which the local curvature is to be approximated and the two layers of neighbors as indicated by the bold line in the figure. Each position in this template is assigned a number (1 for all positions in the example).
2. This template is mapped over a cell at a grain boundary and observes to which grain the center cell belongs.
3. Subsequently, the template then assigns a zero to all positions belonging to other grains, and the number in the template to positions belonging to the same grain as the center one.
4. The sum of all positions in the resulting 5×5 matrix (adding up to 12 in the example in Figure 4-16) returns a number which can be scaled to an approximation of the curvature:

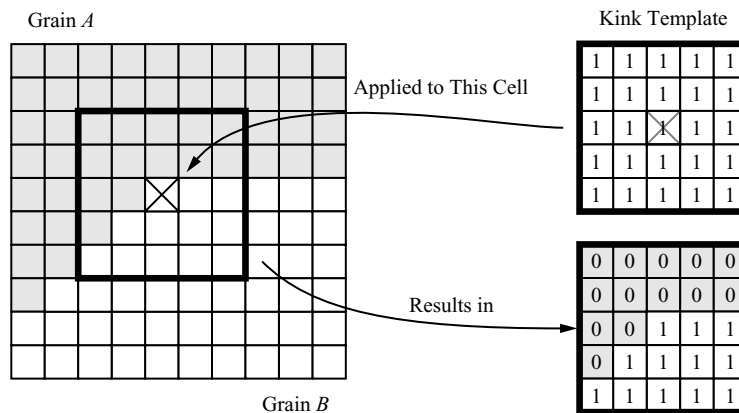


FIGURE 4-16 A method for direct approximation of local curvature at a grain boundary. See text for more details.

$$\kappa = 15 - \sum_{\substack{k \in \text{1st nn} \\ k \in \text{2nd nn}}} \phi_k \varepsilon_k \quad (4.14)$$

which returns 0 for a locally flat grain boundary.

Combining equation (4.14) with Algorithm 4-4 leads to simulation results as illustrated in Figures 4-17 and 4-18. Figure 4-17 illustrates the average rate at which the grain area cross

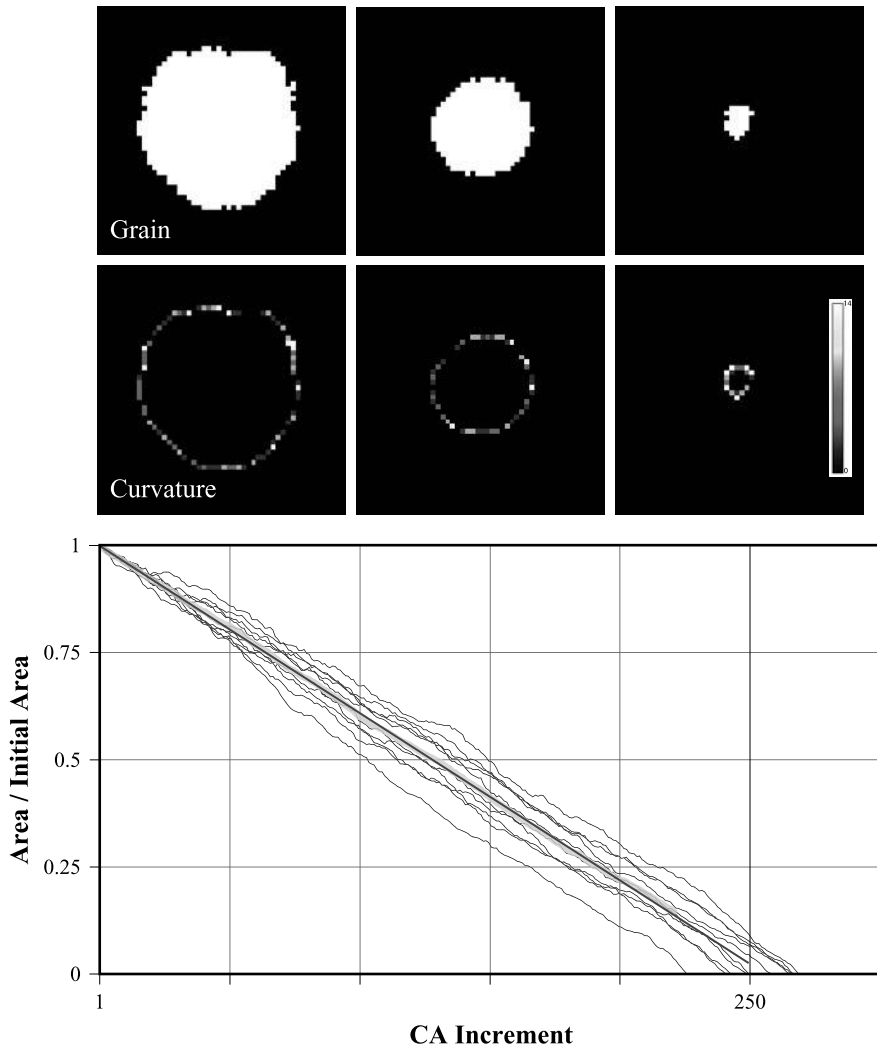


FIGURE 4-17 50×50 2D cellular automaton simulations of a single grain shrinking in a matrix. The curvature in the second row of images is as locally computed using equation (4.14) and plotting values smaller than zero in black. The thick line in the bottom plot is the average of the other lines representing different simulations, simulations for which only the random number sequence in the simulation differs.

section reduces for a single, spherical (circular in 2D) grain. The cross-section area reduces with a linear rate, which is as can be derived from equation (4.12). This is somewhat surprising, given the simplicity of the curvature approximation that was implemented. Another, well-known feature of grain growth can be reproduced as illustrated in Figure 4-18: the average grain area should increase with a linear rate.

Figure 4-19 shows the same simulation starting from 1000 grains in a 200×200 grid for 1000 CA increments. As can be observed the simulation starts deviating from the theoretical course

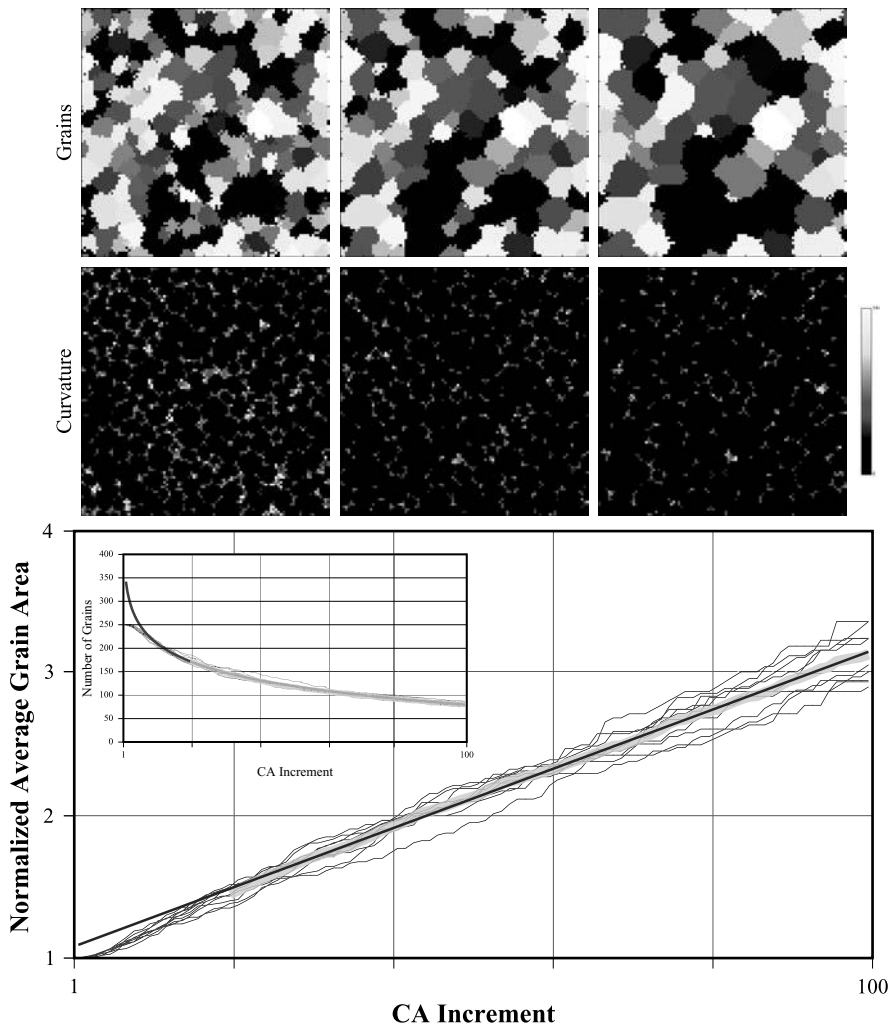


FIGURE 4-18 100×100 2D cellular automaton simulations of grain growth starting with 250 grains. The curvature in the second row of images is as locally computed using equation (4.14) and plotting values smaller than zero in black. The bottom plot illustrates how the average grain area increases linearly after a short transition time. The initial microstructure was generated using a cellular automata recrystallization simulation. The inset plot shows the reduction of the number of grains left in the simulation, which seems to fit a logarithmic rate.

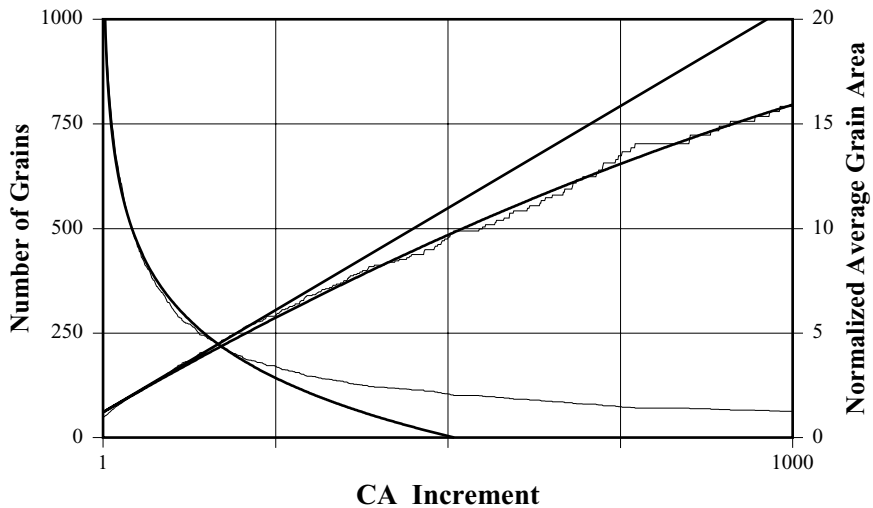
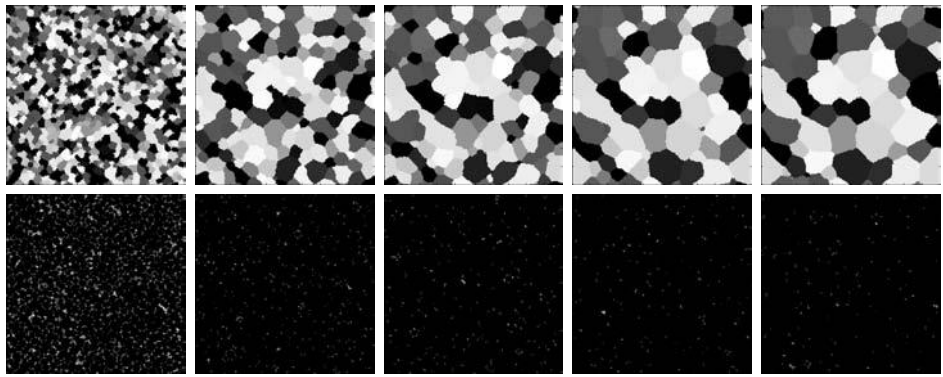


FIGURE 4-19 200×200 2D cellular automaton simulation of grain growth starting with 1000 grains leading to about 70 grains after 1000 CA increments. Note the average grain area increases slower than linear (second order polynomial fit is shown), and that the number of grains decreases slower than logarithmically.

after about 200 increments, the grain growth slows down to a sublinear rate and also the number of grains does not decrease as quickly as predicted by theory. Usually this deviation is assessed to be due to the low number of grains, and hence the bad statistics, at the end of the simulation. But the correctly behaving simulation shown in Figure 4-18 starts at about the number of grains at which the simulation shown in Figure 4-19 starts to fail, so one must conclude the low number of grains is *not* the cause of the problem.

The answer is to be found, again, in the periodic grid on which the cells are distributed and on which the neighborhood and curvature approximation is based. The cellular automaton simulation is doing nothing else than minimizing the amount of grain boundary energy present in the system, *and on a periodical grid only a limited set of orientations can have zero curvature; consequently these orientations are less probable to move.* In addition, or in other words, a periodic grid of cells cannot resolve curvature below a neighborhood-size-dependent cutoff, and this minimal curvature is dependent on the orientation of the grain boundary surface.

Conclusion: as the grains get large relative to the cell size, the grain boundary curvature decreases relative to the cell size, and consequently the cellular automaton underestimates the driving pressure.

Obviously this problem is of higher order, but it can become a problem for time-extended simulations, as no linear scaling between CA increment and underlying physics is available. Whether one needs such exact scaling, or whether the knowledge of the problem is enough to estimate the deviation of course depends on the goal of the simulation at hand. At least after reading this section you are aware of it—so please stop waiving your hands.

Separate Surface Maps?

Many researchers turn to one or the other method of separate mapping of the grain boundary surfaces. For example, in a 2D simulation one can approximate the exact location of the grain boundaries by tracking the positions of the triple points and constructing cubic splines between them for the grain boundaries. This separate map of grain boundaries can then be used to compute curvature at the cells intersected by these splines. The exact nature of the interaction between such separate grain boundary maps, the periodic grid, neighborhood definition, and state transformation function can be made almost infinitely complex, and will obviously reduce the magnitude of the interface discretization problem.

However complex and intricate the boundary mapping may be, it is nothing more than a modification based on an incorrect starting point, and it does not fundamentally solve the problem. In my opinion it is better to leave it be and in doing so remain aware the simulation results are biased [see also equation (1.1)]. One should be especially careful when using cellular automata (or any other periodic grid-based computational method for that matter) for a statistical evaluation of microstructure evolution under circumstances where boundary orientation plays a relevant role. But that is a discussion beyond the scope of this book.

Later in this chapter irregular grid cellular automata are described, which are this author's first attempt at solving the interface discretization problem—objectivity thus not guaranteed. Irregular grid based cellular automata are not all that evident to program and also come at a computational cost—it is up to the researcher to decide whether this is worth it.

4.5 A Mathematical Formulation of Cellular Automata

In the context of microstructure evolution, cellular automata are, in essence, mathematical algorithms that describe the discrete spatial and temporal evolution of complex systems. From the preceding it should be clear that they do so by applying transformation rules to compute new states for each cell based on the states in previous time step(s) of cells in a defined, finite neighborhood. In this section an attempt is made to formulate a general mathematical description of cellular automata, mostly based on Dierk Raabe's formal description, for example, as in ref. [Raa02].

A General Definition

An evolving microstructure, modeled using any type of cellular automaton, is defined by the global state function *upsilon* Υ , which is spatially partitioned into a finite number of cells i with state ξ_i and neighborhood η_i at discrete moments in time t_{k_i} , where the index k_i points out that these points in time need not be the same for all cells i . For Υ to be completely defined it is *necessary* that each cell is given an initial state ξ_i^0 and initial neighborhood definition η_i^0 at time t_{0_i} , thereby defining the initial state Υ_0 , and for each cell i at any time t_{k_i} for $k > 0$, its state ξ_i^k and neighborhood η_i^k are *computable* from information available solely through its state(s) and neighborhood(s) at a previous time or times t_{h_i} , $h < k$. This implies that for each

cell state ξ_i at time t_{k_i} a state transformation function f_i^k (also called state transformation “rule”) specifies how the new states can be computed from previous ones. In summary:

$$\Upsilon \triangleq \Upsilon \left(\xi_i^k, \eta_i^k, f_i^k; \forall i : k = 1, \dots, n_i \right) \quad (4.15)$$

with

$$\xi_i^k \triangleq \xi_i(t_{k_i}) = f_i^k \left(\Xi_i^k \right) \quad (4.16)$$

where Ξ_i^k is the collection of all states ξ_m^h at times t_{h_m} computed in earlier steps and pointed at by the neighborhood η_i^k . The superscript k in f_i^k specifies that it concerns the definition of this function at time t_{k_i} , which may differ from point to point in space-time (analogous for k in η_i^k). Note that this definition does not require the explicit definition of the shape and position of each cell.

Explanation of the Definition

Let us take a 2D, square grid cellular automaton using a von Neumann neighborhood definition to study recrystallization with site saturation for nucleation as a case study to explain the preceding definition, a definition, that is, admittedly, very general.

Our global state function Υ now consists of a collection of square cells partitioning our 2D microstructure. Let us name the cells c_{ij} with i the row number and j the column number of the cell. Each cell c_{ij} has a state ξ_{ij} , which can either have the state deformed or the state recrystallized. The neighborhood of each cell is defined as:

$$\eta_{ij} = \{c_{i,j}, c_{i,j-1}, c_{i-1,j}, c_{i,j+1}, c_{i+1,j}\} \quad (4.17)$$

and remains constant throughout the entire simulation. Note that additional definitions must be specified for the neighborhoods of cells at the edge of Υ , and that these additional definitions depend on whether the boundary conditions are periodic or not. For example, for a nonperiodical Υ , $i = 0$ and $0 < j < n$ with n the number of the last column, the neighborhood definition is

$$\eta_{0j} = \{c_{0j}, c_{0,j-1}, c_{0,j+1}, c_{1,j}\} \quad (4.18)$$

As in a normal cellular automata new states are computed from the states of cells in the previous time step only, the state transformation function f is defined as

$$\xi_{ij}^k \triangleq \xi_{ij}(t_k) = f_{ij}^k \left(\Xi_{ij}^k \right) \quad (4.19)$$

with, using the neighborhood definition as in equation (4.17),

$$\Xi_{ij}^k = \{\xi_{i,j}^{k-1}, \xi_{i,j-1}^{k-1}, \xi_{i-1,j}^{k-1}, \xi_{i,j+1}^{k-1}, \xi_{i+1,j}^{k-1}\} \quad (4.20)$$

4.6 Irregular and Shapeless Cellular Automata

Irregular or Random-Grid Cellular Automata

The concept behind *irregular* cellular automata is given away by its name: instead of using a periodic distribution, cells are distributed randomly [Jan03]. Apart from this random distribution

irregular cellular automata do not need to be much different from conventional ones, at least not if one uses the Voronoi cells computed starting from a random distribution of points as first implemented by Flache [Fla01] and illustrated in Figure 4-20. The neighborhood of a cell can then, for example, be defined to include each cell with which it shares an interfacial plane or edge, or including additional cells that are farther away using another definition. Once the definition of neighborhood is clear, irregular cellular automata can be used in exactly the same way as the regular, periodic grid ones described before. Using an irregular partition of space adds to the complexity of the algorithms. The neighborhood η_i is now different for each cell c_i , and therefore must be computed and/or stored for each cell separately.

Shapeless or Point Cellular Automata

Shapeless or point cells are a further development from Voronoi-based cell shapes. Shapeless cell shapes are not computed explicitly; only a point position is used. Shapeless cells have the advantage of being much more flexible in their definition of neighborhood. A feasible instantiation of shapeless cellular automata could go like this:

- The starting point is a distribution of points with an average density per unit area or volume. Note that the distribution may be equidistant, like in conventional automata, or random/irregular.
- Each point represents a cell. The volume of a cell can either be an average area or volume that follows directly from the average density, or a more precise definition depending on the local (varying) cell density.
- The neighborhood definition consists of a shape and its according size specification, for example, a sphere and its radius. The neighborhood of a cell includes all point cells whose positions are contained within this sphere. As only point coordinates are used, shapes and sizes can easily be varied, even locally. Position dependent variation of the neighborhood is essential for its application to grain growth, as will become clear in the rest of this chapter.

With this definition, shapeless cellular automata can be used as conventional ones.

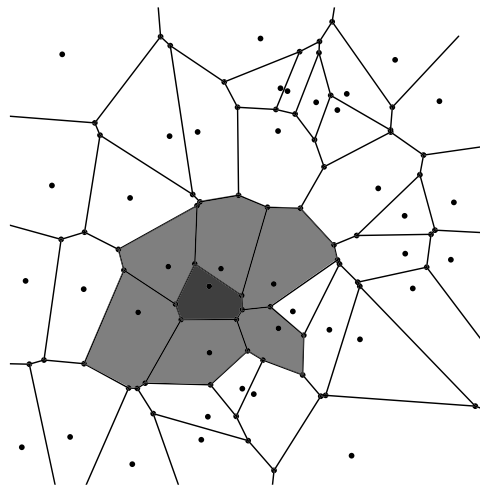


FIGURE 4-20 An example of a two dimensional irregular cellular automata distribution based on the Voronoi cells computed from a random distribution of points. The neighborhood for one cell using the definition as given in the text is indicated.

4.6.1 Irregular Shapeless Cellular Automata for Grain Growth

Before giving an example of an irregular, shapeless cellular automaton model for grain growth in general, one needs to differentiate between volume driven growth, as is the case for recrystallization and phase transformation, and curvature driven grain growth, in which case the driving pressure can be understood as coming from the microstructure minimizing its global energy by reducing the total surface area of grain boundaries present. In respect to modeling grain growth using cellular automata one thus needs to differentiate between three different energetic conditions:

- Pure curvature driven grain growth in the absence of any volume-stored energy.
- Volume driven growth in which the volume-stored energy is much larger than the energy of the grain boundaries, and thus the influence of the energy of the boundaries themselves is negligible.
- A mixed condition, in which grain boundary and additional energy lead to comparable driving pressures, and therefore both must be taken into account.

Depending on the condition, a different modification of cellular automata can be used to construct a model based on equation (4.2).

Pure Curvature Driven Grain Growth

When the motion of grain boundaries is governed by their curvature the driving pressure can be mathematically represented in the most simplified manner by:

$$p = \frac{2\gamma}{R} \quad (4.21)$$

with γ the grain boundary energy and R the local radius of the grain boundary curvature. The issue here is the representation of surfaces in the cellular automata model: in conventional cellular automata the cells are faceted, and any grain boundary surface directly modeled using these facets is in turn faceted and thus discontinuous. As a consequence a high cell density per volume unit is needed to properly model grain boundaries, even in irregular cellular automata. As an example take the grain boundary as modeled in a 2D cellular automata as shown in Figure 4-21. Determine the curvature at the locations A , B , and C . Is curvature zero at location B ? That means it does not move at all. Is it also zero at location C ? And what about location A , how precise can one approximate boundary curvature at that location? That evidently depends on how large a neighborhood of cells one includes in the local approximation of the surfaces. I am sure the better student easily comes up with a variety of approaches to solve this problem. But there exists a completely different, probabilistic state transformation function f that solves the surface problem without the need to first approximate it:

1. Consider the cells which are located within the neighborhood.
2. Determine which grains are present in the neighborhood and how many neighborhood cells belong to each of these grains.
3. The probability that, in the current time step, the cell changes state belonging to grain X is proportional to the number of cells of grain X that are located in its neighborhood. This is algorithmically very efficient as it is the equivalent of randomly picking a cell in its neighborhood and changing state to the grain to which the picked cell belongs.

In the context of an irregular, shapeless cellular automaton, consider a point cell on a curved grain boundary and its neighborhood as depicted in Figure 4-22. The cells above the grain

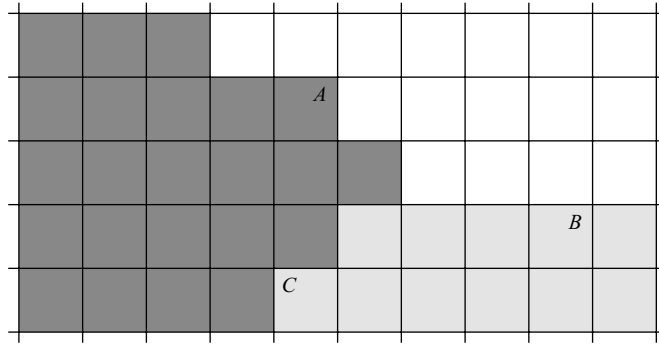


FIGURE 4-21 Grain boundaries as modeled in a 2D square grid cellular automata.

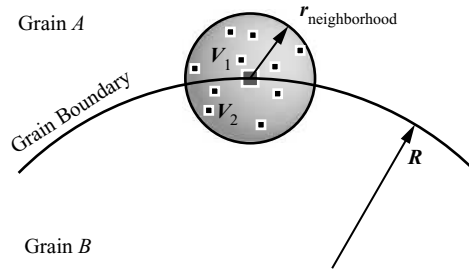


FIGURE 4-22 A shapeless cell on a curved grain boundary and its spherical neighborhood.

boundary belong to grain *A*, the cells below to grain *B*. Assuming the center cell, located on the grain boundary also belongs to grain *B*, its probability to end up in grain *A* is 6/11, so on average the grain boundary will move down as expected (statistically more cells should be found in the larger volume partition).

Does this state change rule mimic the physics in equation (4.21)? The answer to these questions is surprisingly simple. Start with abstracting the cells in Figure 4-22 as volume partitions V_1 and V_2 of the neighborhood of a cell. A point cell at the grain boundary will jump to grain *A* with a probability $P_A = V_1/(V_1 + V_2)$, and to grain *B* with a probability equal to $P_B = V_2/(V_1 + V_2)$. This means that at every increment the grain boundary moves one layer of cells into grain *B*, and that with a probability

$$P = \frac{V_1 - V_2}{V} \quad (4.22)$$

with $V = V_1 + V_2$. What is interesting is that using r_η as the radius of the neighborhood, equation (4.22) computes to

$$P = \frac{3r_\eta}{8R} \quad (4.23)$$

in which the grain boundary curvature radius R shows up again. Given a volume cell density, the average volume that a cell represents can be computed. Approximating the cell volume with an equivalent spherical volume, an equivalent radius r_c can be computed, from which in turn the velocity of the grain boundary in the automaton model can be derived, namely, $v = 2Pr_c/\Delta t$

with Δt the time step for a computational increment. Combined with equations (4.21) and (4.23) this results in

$$m \frac{2\gamma}{R} = v = \frac{3r_\eta r_c}{4R\Delta t} \quad (4.24)$$

In other words, the time step Δt is related to the physics (grain boundary mobility m and energy γ) and the details of the cellular automaton:

$$\Delta t = \frac{3r_\eta r_c}{8m\gamma} \quad (4.25)$$

comply with the theoretical grain growth equation (4.21), while the local variation of the grain boundary curvature is *intrinsically* accounted for by the cellular automaton, and the curvature need not be calculated explicitly. In addition, local variation of the grain boundary mobility m and energy γ can be straightforwardly introduced by locally varying the neighborhood radius r_η .¹

Equation (4.25) constitutes a relation between r_η , Δt , and r_c (r_c fixes the spatial resolution); note that this link forces the model to be an explicit one by connecting the time increment to the spatial resolution. Also, r_η cannot be chosen entirely freely. Its maximum value is limited by considerations of computing resources: a large neighborhood increases the number of cells that one must keep track of. The minimum value of r_η is also limited, as for too small a value the neighborhood contains no cells. Though a burden to the programmer, the latter limitation can be overcome by setting a minimal value for r_η . In case the value needed to satisfy equation (4.25) is smaller, then only perform the state change with a proportional probability.

Grain Boundary Plane Normals?

The careful reader may have already come up with the observation that grain boundary energy and mobility are not only dependent on the misorientation between the neighboring crystals, but also depend on the local orientation of the grain boundary plane itself. If that is so, then how can this be introduced into the cellular automaton model without first approximating the surface itself? A procedure solving this problem could go as follows:

1. Set a feasible neighborhood radius r_η .
2. Check whether cells belong to more than one grain in this neighborhood, if so the current cell is on a grain boundary.
3. If more than two grains are observed within the neighborhood, consider the two grains with the most cells as constituting the grain boundary.
4. For each of these grains, compute the center of mass of all cells belonging to the same grain.
5. The direction of the plane normal runs through these two mass centers.

Solution of Problem 4-8 can tell you in which case this procedure is useful, and in which case you need to approximate the plane normal with *smoother* methods, unfortunately requiring more computational resources.

¹ r_η is systematically underestimated in a discrete spatial representation as used in cellular automata, which introduces an additional correction factor into equation (4.25).

PROBLEM 4-8: Plane Normal Error

Artificially create a plane grain boundary surface represented in a random cloud of points, and use the preceding procedure to compute the plane normals. Analyze the statistics of the error on the plane normal found in the random grid as a function of neighborhood size.

Triple Junctions

Equation (4.25) also works for the case of triple or multiple junctions. Consider the triple junctions depicted in Figure 4-23. The vertical grain boundary constitutes a driving force for the triple junction to move down. The cellular automaton now determines for each cell to which grain boundary it belongs and then computes the neighborhood size using equation (4.25). It should be clear from the figure that cells close to the triple junction have a higher probability to end up in grain *A* than in grain *B* or *C*. Variable grain boundary energy and mobility can also be taken into account as shown in the right part of Figure 4-23: a lower energy for the boundary between grains *C* and *D* leads to a smaller neighborhood and thus to a smaller number of cells that are able to change to grain *A* in a time increment (i.e., they must be closer to the triple junction to be able to make the jump), hence a lower grain boundary velocity. Figure 4-24 shows a simple example of two triple junctions in an evolving microstructure. Their velocity is constant as the driving pressure, namely, the energy stored in the horizontal grain boundaries, is constant.

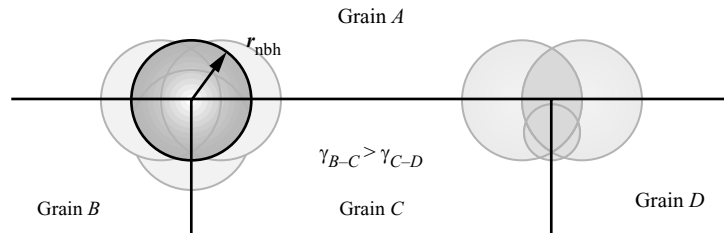


FIGURE 4-23 An unstable triple junction configuration, for (left) equal grain boundary energy and (right) a relatively lower energy for the boundary between grains *C* and *D*. Neighborhoods for cells at different positions are shown.

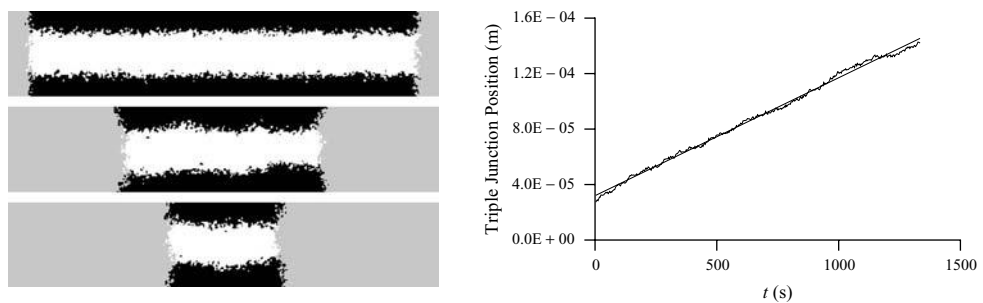


FIGURE 4-24 The evolution of a microstructure (symmetrical boundary conditions) showing the motion of triple junctions. The diagram on the right illustrates the velocity of the triple junctions is constant (arbitrary physical constants were used).

PROBLEM 4-9: Triple Junction Modeling

Using different types of cellular automata models, simulate the triple junction configuration shown in Figure 4-23 and analyze the differences in predicted triple junction velocity. If you also simulate a shrinking spherical grain and calibrate its grain boundary velocity against theory, does the CA model predict the correct triple junction velocity? Assume there is no additional triple junction drag and make sure mobility is independent of misorientation. In doing so the energy of the horizontal grain boundary in Figure 4-23 defines the theoretical velocity of the vertical ones.

4.6.2 In the Presence of Additional Driving Forces

Large Driving Forces and an Inverse Update Scheme

In conventional cellular automata all the cell states are updated in every step of the computation. This computing scheme requires that in each increment a new status must be computed for all cells. In the presence of a large driving pressure, compared to which the driving pressure from the grain boundary curvature can be neglected, the direction of motion of the grain boundaries is predetermined and unchanging. This is the case for transformations where nucleated grains grow into an existing microstructure, including recrystallization and certain phase transformations, that is, in those microstructure transformations concerning the growth of nucleated grains into an existing microstructure. In the presence of such a large driving pressure updating all cells is not necessary. To optimize the algorithm the cell state can differentiate between its location being *behind* (or in a newly nucleated grain, i.e., `in--nucleated`) or *in front* of the moving grain boundary. The most straightforward algorithm optimization that can be made is a simple preselection of cells: those having a state labeled behind the moving boundary are excluded from the updating scheme performing the recrystallization. An alternative update rule [Jan03] approaches the update problem from another direction: if a cell has an `in--nucleated` state, then annex all the cells in its neighborhood to the grain to which the current cell belongs. By introducing this rule the numerical method is taken outside the normal definition of cellular automata and actually shows more similarities with an Eden-type model ([Ede61], [JB85]). Looking at the modified cellular automata more closely, it becomes clear that it is no longer possible to update all the cells in parallel: a cell with state `not--in--nucleated` in the neighborhood of two different nucleated grains could be annexed by the first or by the second grain. Randomizing the order in which the cells are evaluated for neighborhood updating averages out any influence on the statistical outcome of a simulation. An additional problem could be that, again depending on the evaluation order, a cell annexed in an update step could in turn annex the cells in its own neighborhood within the same step. Fortunately this kind of cascading can be circumvented, again at the expense of the programmer's nerves, by excluding all cells that were not annexed before the current step from being used. To do so an additional state is introduced: `completed`. At all times cells now have one of three states: `in--existing--grain`, `in--nucleated--grain`, or `completed`; the updating rule is now: to consider all cells with state `in--nucleated--grain` (one after the other in a random order), to change their state to `completed` and to change the state of all `in--existing--grain` cells in their neighborhood to `in--nucleated--grain`.

Small Driving Forces and Drag Forces

The combination of a grain boundary driving pressure with an additional driving or drag pressure can be solved using two different approaches. The first option is to combine the inverse and the conventional update rule within the same time step, more precisely by performing one

after the other. Such sequencing has been implemented with success by many other authors, for example, Landau and Binder [LB01] or Rollett and Raabe [RR01] to name just two, more recent implementations. The second option is to modify equation (4.22) to include an additional driving force [Jan03]:

$$P = \frac{V_1 - V_2}{V(1+f)} \pm \frac{f}{1+f} \quad (4.26)$$

This modification adds an imaginary volume to the existing neighborhood of size fV , which represents an additional driving force p_a and can be straightforwardly introduced into the algorithm at the time the neighborhood is evaluated. Equation (4.26) guarantees that the absolute value of P is between zero and one, independent of the value chosen for f . The sign in front of the second term depends on the direction of the additional driving force p_a and is positive if acting in the same direction as the curvature pressure does. Equation (4.24) then becomes:

$$m \left(\frac{2\gamma}{R} \pm p_a \right) = v = \frac{2r_c}{\Delta t} \left(\frac{3r_\eta}{8R(1+f)} \pm \frac{f}{1+f} \right) \quad (4.27)$$

For a given time step Δt thus follows that

$$r_\eta = \frac{8m\gamma(1+f)\Delta t}{3r_c} \text{ and } f = \frac{3p_a r_\eta}{16\gamma} \quad (4.28)$$

which can be rewritten as:

$$r_\eta = \frac{16m\gamma\Delta t}{3(2r_c - p_a\Delta t)} \text{ and } f = \frac{p_a\Delta t}{2r_c - p_a\Delta t}, \text{ with } \Delta t < \frac{2r_c}{p_a} \quad (4.29)$$

A locally varying driving force thus leads to a locally varying value for f , which provides an entry point in the update rule to locally correct for the presence of any additional driving force. Note that in case the value of p_a gets very small, the equations reflect pure curvature-driven grain growth and the maximum value for Δt is limited by the computationally feasible maximum value of the neighborhood radius r_η .

4.7 Hybrid Cellular Automata Modeling

With *hybrid* cellular automata we mean combinations of different types of CA and other computational methods. When a microstructure transforms many different processes can occur concurrently. The process of dynamic recrystallization is an example in which plastic deformation, recovery, recrystallization, and precipitation occur simultaneously. In this section a combination of CA with volume finite difference modeling is described, which models both the grain growth process and the diffusion of solutes.

4.7.1 Principle

Before modeling concurrent microstructural processes you need an approach to split up the different processes in your computational model. Focusing on our case study: it is clear we can use cellular automata to model grain growth and another method to model the diffusion of solute elements in the microstructure, but how can we model two processes *and* their interaction simultaneously? Well, the answer is rather simple: we do not. What we do is compute both processes *sequentially* for a small but finite time-step, which in the limiting case of an infinitesimally small

time step is the same as computing them concurrently. Both models operate on the same data structure representing the microstructure, and the interaction between the processes should follow through that data structure. An obvious remark to make is that *both* methods must converge; in other words a cellular automaton step may not interfere with the numerical convergence of the other method and vice versa.

4.7.2 Case Example

Problem Description

A single phase microstructure is submitted to a heat treatment in which grain growth occurs. The composition of the material is such that a single solute element is present in the microstructure, a solute element which preferentially sits on the grain boundaries, the equilibrium concentration being 10 times (relative) that of its concentration in the grain bulk, until a saturation concentration is reached, say 10%. The solute element diffuses faster on the grain boundaries than in the grain bulk. The mobility and energy of the grain boundaries depends on the amount of solute on the grain boundary (dependency given by some arbitrary function).

Modeling Solute Diffusion

Modeling the diffusion of solute elements can be done using different approaches, for example, using phase field modeling (see refs. [CKYY02], [ZCCT01]), or using Potts-type Monte Carlo simulation [STH97]. In this section we propose a method, first published in Janssens *et al.* [JHF04], which is computationally highly compatible with an irregular, shapeless cellular automaton, but also works fine with a regular grid one, and is fairly easy to program.

Independent of its type, a cellular automaton consists of a +2-dimensional grid of shapeless cells distributed in space. Each cell i represents a volume V_i of material with locally attributed properties, which also can include a concentration for one or more solute elements. The diffusion model is discretized as illustrated in Figure 4-25. On the left, two cells are depicted, each given a certain activity $a_i = \mu_i c_i = n_i/V_i$ of a solute element (n_i is the number of solute atoms, μ_i the activity coefficient and c_i the concentration of solute element i). Solute segregation to grain boundaries can be modeled straightforwardly by introducing different activity coefficients in the bulk and at the grain boundary. The cells are at a known distance d apart. A is the surface area through which diffusion can occur. The flux of solute atoms through an area A from cell 1 to 2 is defined by Fick's first law:

$$J_{1 \rightarrow 2} = A \frac{D}{d} \left(\frac{n_1}{V_1} - \frac{n_2}{V_2} \right) \quad (4.30)$$

Integration of this equation assuming mass conservation results in:

$$n_1(t) = \frac{V_1}{V_1 + V_2} n + \left(n_1(t=0) - \frac{V_1}{V_1 + V_2} n \right) e^{-A \frac{D}{d} \frac{V_1 + V_2}{V_1 V_2} t} \quad (4.31)$$

with constant total number of atoms $n = n_1 + n_2$ and variable time t . Special cases can be formulated, such as a constant concentration in one of the cells. These *source* or *sink* boundary conditions lead to an alternative equation:

$$n_1(t) = \frac{V_1}{V_2} n_2 + \left(n_1(t=0) - \frac{V_1}{V_2} n \right) e^{-A \frac{D}{d} \frac{1}{V_1} t} \quad (4.32)$$

for c_2 constant. Next an algorithm is designed to calculate the diffusion process in +2 dimensions: For each cell do the following:

1. Determine all neighboring cells within a predetermined radius r_η as illustrated in Figure 4-25.
2. Sequentially for each cell in this neighborhood, compute the diffusion flux using equation (4.31).

Note that this algorithm evaluates each pair of cells twice. To link a simulation step to a space-time the simulation needs to be calibrated by performing a computational experiment of an analytically solvable problem. This calibration links the value of the surface area A to a value for the neighborhood radius r_η . r_η can be chosen arbitrarily within such limits that on average a feasible number of cells is found.

In Figure 4-26 the results of a three-dimensional simulation are shown, illustrating the approach is quantitatively correct. The experiment consists of a plane sheet of thickness $2l$ with initial solute concentration C_0 , surrounded by a gas with a constant solute concentration C_1 . The non steady state solution is given by J. Crank [Cra03]:

$$\frac{C - C_0}{C_1 - C_0} = 1 - \frac{4}{\pi} \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \exp\{-D(2n+1)^2 \pi^2 t / 4l^2\} \cos \frac{(2n+1)\pi x}{2l} \quad (4.33)$$

Modeling Growth

If you have been a hard working student you probably have already programmed Algorithm 4-4 in detail—if not you can download it from this book’s Web site, that is of course if I have been working hard as well

Combining Both Computational Models into One, Hybrid CA

Before combining both computational models, you need to make sure the microstructure is represented using a data structure on which both models can operate:

- Partition the 2D microstructure into square cells.
- The state of a cell can be any grain identification number.
- Attribute to a cell is the concentration of a solute atom, which can be any rational number between 0 and 0.1, the latter being the saturated concentration on a grain boundary. Note that this limits the range of applicability of the model to low solute concentrations—but that is the sensible range anyhow.

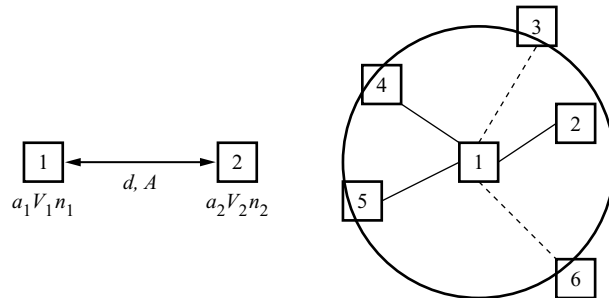


FIGURE 4-25 Schematic of discretized diffusion. Each cell i represents a volume V_i with a concentration of solute atoms c_i or a number of solute atoms n_i .

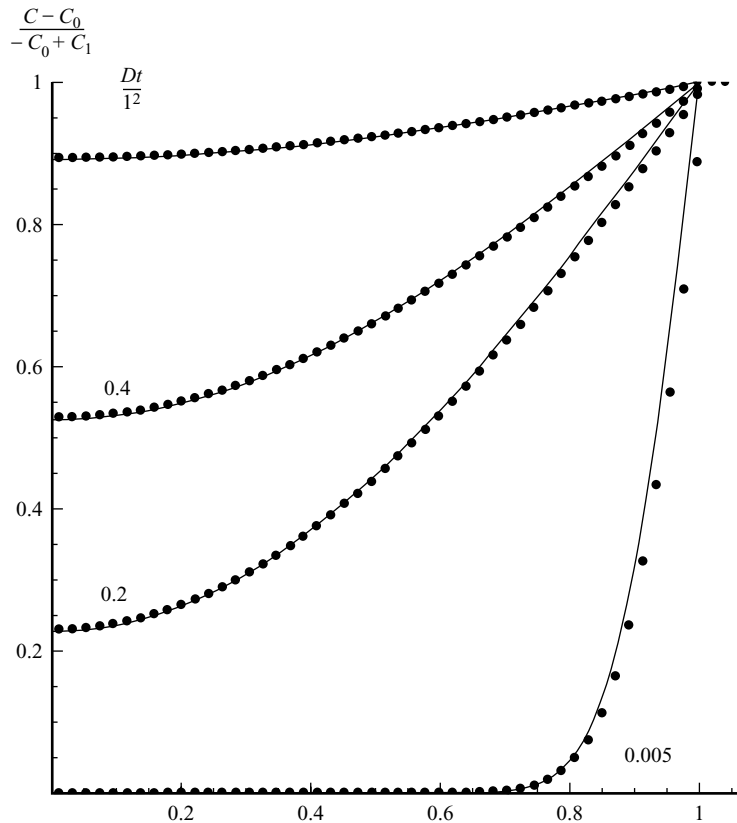


FIGURE 4-26 Comparison of theory (lines) and simulation (disks) of the experiment described in the text.

We now modify Algorithm 4-4 and add a line for the diffusion step. The diffusion step in the new Algorithm 4-5 consists of the previously described model, locally switching between a low activity coefficient when the cell lies on the grain boundary, and a high activity coefficient when not. In doing so, the solute atoms will segregate to the grain boundaries, and that while these move. Note that locating the grain boundaries is specified explicitly, as this is used in both the diffusion and the grain growth step. Algorithm 4-5 computes first the diffusion and then the grain growth for the same time step, but it still does not couple both models. The only modification we still have to implement is to make the probability that a cell switches grain dependent on its momentary solute atom concentration. Experiment with making grain boundary mobility and/or energy a function of the solute atom concentration and see what happens.

ALGORITHM 4-5: Two-Dimensional Cellular Automata for Grain Growth

- 1: Input or generate an initial microstructure
 - 2: **for** n cellular automaton steps **do**
 - 3: Determine and store which cells are on a grain boundary
 - 4: Compute the diffusion step
-

```

5:   Make a copy of the current cell states and name it previous microstructure
    {All computations in this step are made based on this copy}
6:   for Each Cell do
7:     if Cell is at a grain boundary then
8:       Compute local boundary curvature
9:       if Curvature  $\kappa > 0$  then
10:        Compute new state of cell
11:        {i.e., the probability that it switches grain}
12:      end if
13:    end if
14:  end for
15: end for

```

A complication, which pops up when performing extended, quantitatively correct simulations, is making sure time steps for both models are effectively equal. Relative conclusions, namely, what changes when, for example, the solute element's diffusion coefficient is increased, can of course be drawn without much further effort. This is a mere matter of calibrating the models correctly. You should be warned that such is a tedious task that must be performed with great care, and that its relevance increases the longer in time your simulation needs to predict.

4.8 Lattice Gas Cellular Automata

Lattice gas cellular automata, LGCA in short, are a form of CA in which position and momentum of interacting particles are modeled on a discrete grid of points. This book is not about fluid dynamics, but I feel it is relevant to at least treat the basics of this branch of cellular automata as LGCA can be straightforwardly combined with conventional CA. This *may* have advantages regarding algorithm complexity and/or computational resources over the implementation of hybrid modeling, simply because only one type of numerical approach is used instead of two or more. On the other hand, other particle-type methods (e.g., ref. [LL03]) are used more frequently and therefore have been developed to a more advanced level. Nevertheless, LGCA can be used to model the fluid flow of liquids or gasses, and as such it is of interest for modeling microstructure processes in which such flow plays a role, as can be the case in, for example, deposition of thin layers from an electrolyte, solidification processes, sintering, and thixo-forming processes. The following sections are limited to an explanation of the basic principles. The reader is referred to other literature (e.g., ref. [Raa04b]) for more advanced developments, like the Boltzmann-type LGCA.

4.8.1 Principle—Boolean LGCA

In LGCA the material is modeled as a collection of fictive particles, each of which represent a small volume of liquid or gas. The particle state includes the specification of their momentum vector. In Boolean LGCA, these fictive particles can only move on a grid of directions, while their mass and the magnitude of the velocity vector are both constant for all particles. As a first example, consider a 2D square grid as shown in Figure 4-27. The cells are the nodes on the grid. The motion of the particles is limited to the geometry of the grid. In a Boolean LGCA all particles have the same (unit) mass and velocity. On a square grid each particle can only move in one of four directions. A conventional LGCA also imposes that at any time, no node on the grid may contain more than one particle moving in the same direction, that is, not more than

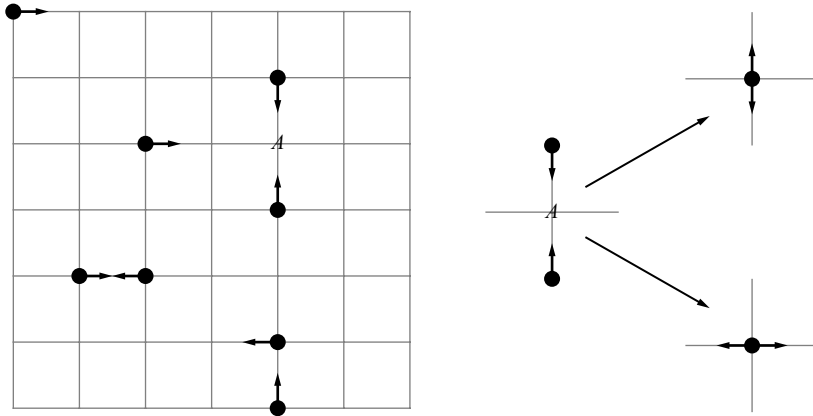


FIGURE 4-27 Two-dimensional lattice gas cellular automata.

one particle can move along the same vertex in the same direction at the same time. This is to be guaranteed by the initial distribution of particles and velocity vectors, and by the state change rules. Algorithm 4-6 is a simplified example of an LGCA algorithm. Assuming Figure 4-27 is the initial state, the following events occur in position *A* in the grid:

ALGORITHM 4-6: Simple Boolean LGCA

- 1: Define a cell's state as the number of particles and their direction of motion
 - 2: Define a cell's neighborhood as the four neighboring cells
 - 3: Populate the square grid with cells {vertex occupation limits!}
 - 4: **for** *n* LGCA steps **do**
 - 5: **for** Each cell **do**
 - 6: Transfer incoming particles from neighboring cells—*hop*
 - 7: Transform incoming particle directions to outgoing directions using the state change rules—*scatter*
 - 8: **end for**
 - 9: **end for**
-

Line 3: Initially, there are no particles in the cells at location *A*.

Line 6: Both from the cell above and the cell below a particle is incoming.

Line 7: A state change rule is needed. As the rule must be conservative, we note that the sum of direction of motion at position *A* is zero, which must remain so when the particles are transformed (one could also say *scattered*) to an outgoing set. On a square grid that leaves us with only two possibilities for the current configuration at *A*, as depicted in the right part of Figure 4-27. One can thus define either a deterministic state change rule, picking only one of the possibilities, or one can define a probabilistic state change rule and let either occur by chance with probability 1/2.

PROBLEM 4-10: Two-Dimensional LGCA

Program a simple 2D LGCA. Add some features like choices of boundary conditions and obstacles ("solid" particles at a fixed position from which particles bounce back).

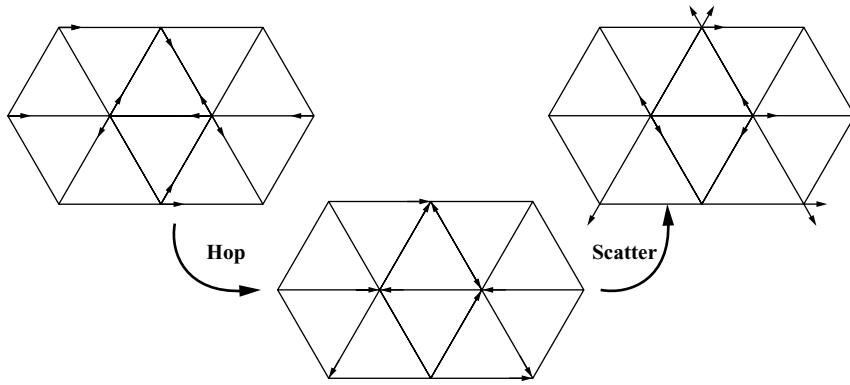


FIGURE 4-28 Hexagonal LGCA.

The example of an LGCA on a square grid was first published by Hardy *et al.* [HPdP73] in 1973. The model has three major problems preventing it from correctly modeling the behavior of fluids: a lack of Galilean invariance, a lack of rotational isotropy, and a crossover dimension problem. These problems are described in detail by Frisch *et al.* [FHP86]. In the same article the authors show that switching to a *hexagonal* LGCA, see Figure 4-28, solves the problem of rotational isotropy to a satisfactory level, while the violation of Galilean invariance is not relevant at low Mach numbers, and the crossover dimension problem no longer exists when one constructs a 3D model.

In conclusion to this very brief introduction to LGCA, here is a partial quote from a very complete book by Rothman and Zaleski [RZ97] on the matter (the authors write about a hexagonal grid LGCA with particular state change rules):

Although the microscopic makeup of fluids ranges from the simplest mon-atomic gas to, say, a complex mixture such as milk, nearly all fluids flow in a way that obeys the same equations of fluid mechanics. . . . we introduce a microscopic model of a fluid that is far simpler than any natural fluid. . . .—at a macroscopic scale it flows just like them!

Indeed, a lattice gas cellular automaton, given a particular combination of grid *and* a specific set of state change rules, does behave as a real fluid—the secret of the trade is what rules to apply. The interested reader is referred to, for example, reference [RZ97] for those secrets.

4.8.2 Boolean LGCA—Example of Application

Assume we have an electrolyte which flows over a substrate with existing thin layer structures, and that from this electrolyte an additional thin layer is to be deposited. Simplifying the problem to the maximum, assume one has direct control over the amount of electrolyte that flows. Also assume that, given the composition of the electrolyte, one knows (e.g., setting the deposition current) exactly how much material is deposited. The situation described is depicted in Figure 4-29, and can be put into formula as follows:

1. The (incoming) electrolyte contains a deposition-element α with concentration $c_\alpha < 1$ per volume unit.
2. The flow rate of the incoming electrolyte is f volume units per time units.
3. The deposition rate is linearly dependent on the element concentration, and, as an example, is specified by $d = d_\infty(1 - \exp^{-sc_\alpha})$, with d_∞ and s constants.
4. The α layer to be deposited is pure α .

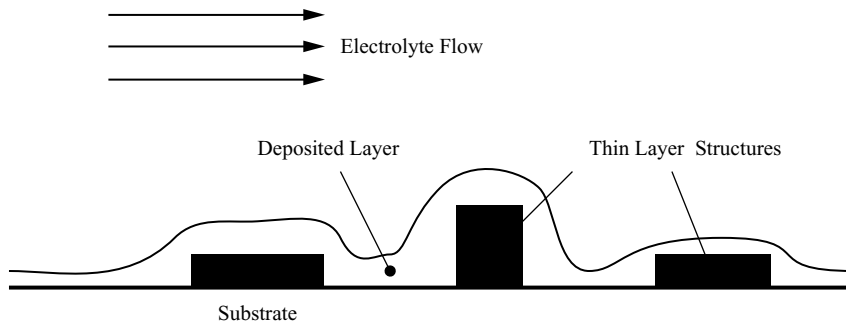


FIGURE 4-29 A simplified deposition process.

First tackle the problem of the electrolyte flow by solving Problem 4-11.

PROBLEM 4-11: Thin Layer Deposition from an Electrolyte

Program a 2D LGCA modeling the flow of electrolyte over thin layer structures as shown in Figure 4-29. Simulate the deposition process by stopping the particles' motion, for example, when they hit a nucleation site on the substrate and when they hit particles which already stopped moving (i.e., these are the "solidified" particles).

To model the deposition process we add a Boolean attribute to the state of the LGCA which can take the values of either `solid` or `liquid`. Algorithm 4-7 is a possible solution to Problem 4-11. When running this algorithm you may end up with holes in the deposited layer. To solve this problem you will need to add a step which corrects this, for example, by shuffling solid cells until the hole ends up at the edge of the layer.

ALGORITHM 4-7: A Solution to Problem 4-11

- 1: Define a cell's state to include the direction of motion of particles and the attribute `solid/liquid`
 - 2: Define a cell's neighborhood as the four neighboring cells
 - 3: Initialize the substrate shape including the thin layer structures by setting the appropriate cells' attribute accordingly (i.e., substrate and structures `solid`, the rest `liquid`)
 - 4: Populate the square grid with cells {vertex occupation limits!}
 - 5: **for** n LGCA steps **do**
 - 6: **for** Each cell **do**
 - 7: **if** cell is `liquid` **then**
 - 8: Transfer incoming particles from neighboring cells
 - 9: Compute whether the cell changes state to `solid`
 - 10: **if** so **then**
 - 11: Select one of the incoming particles and bounce off the remaining ones.
 - 12: **else**
 - 13: Transform incoming particle directions to outgoing directions using the state change rules
 - 14: **end if**
 - 15: **end if**
 - 16: **end for**
 - 17: **end for**
-

4.9 Network Cellular Automata—A Development for the Future?

Network cellular automata (NCA) are a fairly recent development of cellular automata [Wol02]. Here is an attempt at a definition:

- Cells are defined as nodes in a network of connections. Different from conventional cellular automata is that cells do not necessarily constitute a volume partition of space; they are abstracted to a higher level. As a matter of fact, a network can usually be mapped onto any number of dimensions. In Figure 4-30 the cells are drawn on a 2D-periodical grid, but it should be noted that this grid specification is *not* part of the definition of the network, it is merely a way of drawing the network orderly.
- A cell's neighborhood definition is determined by the local configuration of the network. An example of a definition could be: the cell itself plus all cells directly connected to it and all the connections between these cells, as illustrated for the gray cell in the cutout in Figure 4-30.
- The state of a network cell is the configuration of the connections within its neighborhood.
- A state change rules operates on the network configuration within the neighborhood. Connections may be added, deleted, and modified. The only restriction is that no connections should be added to cells outside of the neighborhood. To do so one would need information on the location in the network of cells outside the neighborhood, which is the same as considering a larger neighborhood, for example, by including n th level, indirectly connected cells.

4.9.1 Combined Network Cellular Automata

Combined network cellular automata (CNCA) are a logical extension of NCA. The concept is straightforward: the state of a cell is a combination of the network configuration within the neighborhood *and* a separate state, optionally with attributes, assigned to the cell itself. As depicted in Figure 4-31, the main difference with standard NCA is that a state change rule now also operates on these separate cell states and their attributes. This figure also hints at the tremendous amount of possibilities that *can* be used to define the state transformation

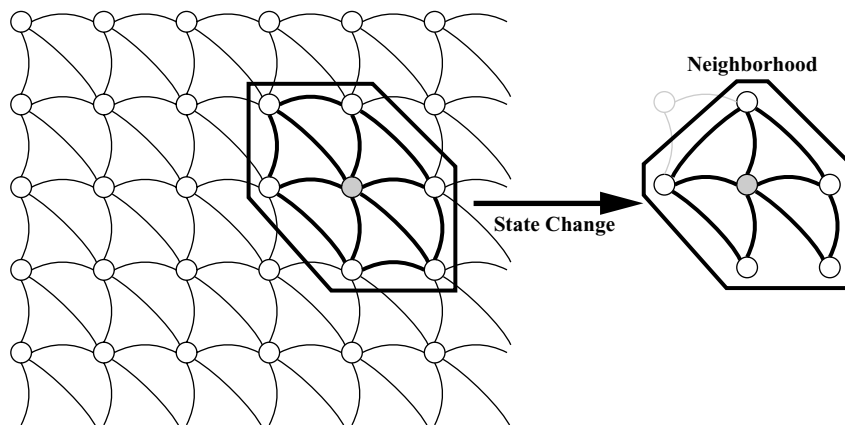


FIGURE 4-30 Network cellular automata.

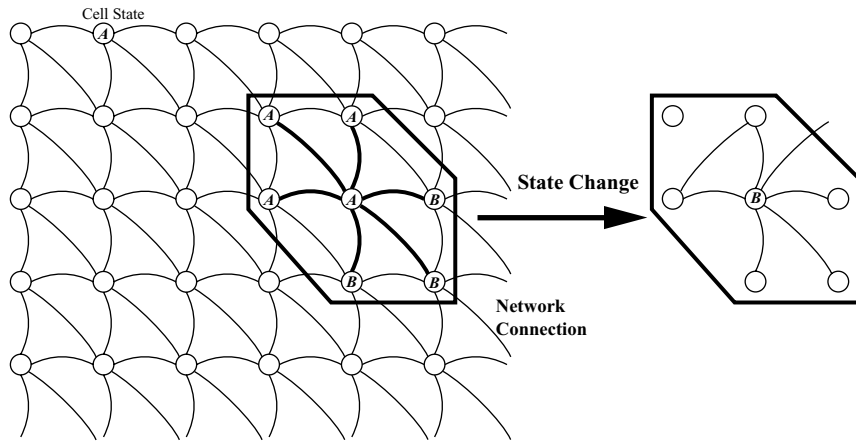


FIGURE 4-31 Combined network cellular automata.

function f , as the combination of network configuration and separate cell states has a high number of possible combinations. It is to be expected that, when using CNCA, one will limit the possible combinations, but the openness of the definition allows one to select those needed. Taking Figure 4-31 as an example:

- f takes as input the states of the cells in the neighborhood and/or the local configuration of the connections between the center cell and its neighbors.
- f needs to be defined for all possible combined state-network configurations, for example, the initial configuration on the left in the cutout in Figure 4-31 leads to the modified situation in the right part of the figure:
 - The state of the cell is changed from A to B .
 - Some network connections are deleted while others are created, thereby changing the local network layout but also redefining the neighborhood for the subsequent CNCA step.

It is important to note that, depending on the type of network reconfiguration, in some cases the state transformation cannot be performed on all cells in parallel, as conflicts may appear when the local network is changed simultaneously for neighboring cells. This topic needs more research on the mathematical method level, but a practical approach is to either construct a state transformation function which makes sure such conflicts cannot occur, or either not to perform state transformations on all cells in parallel.

The crucial question now is, Can such CNCA be useful to model microstructure evolution, or are they only useful in the search for a grand unified theory [Wol02]?

4.9.2 CNCA for Microstructure Evolution Modeling

Different network-like structures can be observed in microstructures:

- Microstructures are grains *connected* by grain boundaries.
- A 2D microstructure is a collection of triple junctions connected by grain boundaries.
- *Possibly*, a 3D microstructure is representable as a collection of quadruple and triple junctions connected by grain boundaries.

- Some microstructures of composite or multiphase materials clearly show networks in their structure.

Taking the first example in the list, here is a plausible way to model grain growth using CNCA:

- Give all the grains in the microstructure a unique identification number.
- The state of a cell can be any of these identification numbers, and represents one whole grain in the microstructure.
- Attribute to a cell's state is its crystallographic orientation. Additional attributes can be the volume of the grain, its average chemical composition, temperature, dislocation density, phase, etc.
- The network configuration is such that each grain cell is connected to the grain cells with which it is in contact through a grain boundary surface (one could add triple junctions).

An example of such a model is given in Figure 4-32.

It is without question that the usefulness of CNCA for the analysis and modeling of evolving microstructures still has to be investigated. But as a last act in this chapter about cellular automata, I would like to put forward the vision that:

1. It is possible to *analyze* the evolution of complex 3D microstructures using CNCA, and that much more detail is to be found about the dependence of microstructure evolution on the local neighborhood of grains, which are, after all, the building blocks of microstructures.
2. Furthermore, that using the results of such analysis, at least in some simplified cases, it is possible to *predict* the evolution of such microstructures.

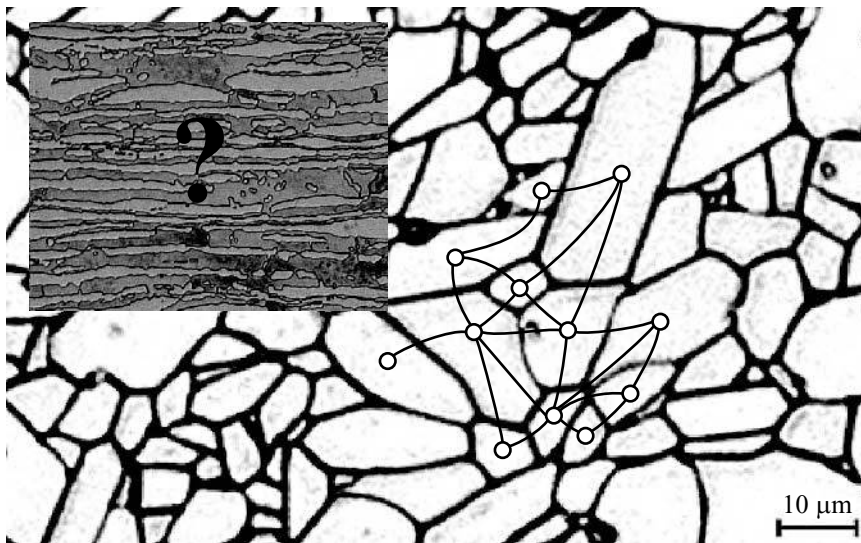


FIGURE 4-32 Combined network cellular automata applied to a simple microstructure of aluminum oxide. The inset shows a more complex microstructure of a rolled duplex steel, where determining the network is also much more complex.

List of Symbols

η	neighborhood
f	state transformation function
γ	grain boundary energy
k	Boltzmann constant ($= 1.38065812 \cdot 10^{-23}$ J/K)
κ	curvature
l_{CA}	one-dimensional length of a cell
m	grain boundary mobility
m_0	temperature independent prefactor of grain boundary mobility
\dot{n}	nucleation rate
P	probability
p	driving pressure
p_a	additional driving pressure
Q	activation energy
R	grain boundary radius
r_η	radius of the neighborhood
r_c	equivalent radius of a cell
T	temperature
t	time
Υ	global state function of a cellular automaton
v	grain boundary velocity
ξ_i	state of a cell

4.10 Further Reading

Cellular Automata in General

Although the book *A New Kind of Science* by Stephen Wolfram [Wol02] is quite controversial, could have been written without the first 200 pages, and is written in a new kind of style, which one may find irritatingly arrogant, I consider Stephen Wolfram one of the brightest pioneers of cellular automata and his book teaches a lot about the method. If you find yourself looking for references to other works in that book, there are none! Stephen Wolfram tells you to go look for those on your own on page 850. Pity.

More Advanced Literature on Cellular Automata and Microstructure Evolution

After reading this chapter, which really only covers the very basics, you are ready to become an expert. Here is a list of books that may be a help in doing so:

- D. Raabe. *Computational Materials Science*. Wiley, New York, 1998.
- D. Raabe, F. Roters, F. Barlat, and L.-Q. Chen (ed.). *Continuum Scale Simulation of Engineering Materials*. Wiley-VCH, New York, 2004.
- B. Chopard and M. Droz. *Cellular Automata Modeling of Physical Systems*. Cambridge University Press, Cambridge, 1998. This book also gives an overview of the history of cellular automata, but quickly focuses its attention on LGCA-type cellular automata.

Literature in Journals

Here is a list of articles in which the use of cellular automata is reported in the context of microstructure evolution, incomplete but nevertheless a start. Hesselbarth (1991) [HG91], Hesselbarth (1993) [HKH93], McCarthy (1994) [McC94], Pezzee (1994) [PD94], Davies (1995) [Dav95], Bullard (1995) [BGCJ95], Liu (1996) [LBP96], Davies (1997) [Dav97], Kremeyer (1998) [Kre98], Davies (1999) [DH99], Marx (1999) [MRG99], Ostrovsky (2001) [OCSBY01], Rollett (2001) [RR01], Xu (2001) [XL01], Janssens (2002) [JVR02], Raabe (2002) [Raa02], Vandyoussefi (2002) [VG02], Zhang (2002) [ZZXLW02], Janssens (2003) [Jan03], Zhang (2003) [ZWZ⁺03], Guillemot (2004) [GGCH04], Janssens (2004) [JHF04], Khvastunkov (2004) [KL04], Lan (2004) [LLHL04], Li (2004) [LA04], Raabe (2004) [Raa04a, Raa04a], Saunier (2004) [SCSB04], Lan (2006) [LLL06].

Lattice Gas Cellular Automata

Find all you need to know in the very pleasantly written book by D. H. Rothman and S. Zaleski, *Lattice-Gas Cellular Automata Simple Models of Complex Hydrodynamics*. Cambridge University Press, Cambridge, 1997. Of course, if you really want to understand what they write, you better be more of an expert in fluid dynamics than the author of this chapter, which you can easily achieve by reading refs. [Suc01] and [Wol00].

Bibliography

- [BGCJ95] J. W. Bullard, E. J. Garboczi, W. C. Carter, and E. R. Fuller, Jr. Numerical methods for computing interfacial mean curvature. *Comp. Mater. Sci.*, 4:103–116, 1995.
- [CD98] B. Chopard and M. Droz. *Cellular Automata Modeling of Physical Systems*. Cambridge University Press, Cambridge, 1998.
- [CKYY02] P.-R. Cha, S. G. Kim, D.-H. Yeon, and J.-K. Yoon. A phase field model for the solute drag on moving grain boundaries. *Acta Mater.*, 50:3817–3829, 2002.
- [Cra03] J. Crank. *The Mathematics of Diffusion*, 2nd Ed., p. 47. Oxford University Press, Oxford, 2003.
- [Dav95] C. H. J. Davies. The effect of neighborhood on the kinetics of a cellular automaton recrystallization model. *Scripta Metall. Mater.*, 33(7):1139–1143, 1995.
- [Dav97] C. H. J. Davies. Growth of nuclei in a cellular automaton simulation of recrystallization. *Scripta Mater.*, 36(1):35–40, 1997.
- [DH99] C. H. J. Davies and L. Hong. The cellular automaton simulation of static recrystallization in cold rolled AA1050. *Scripta Mater.*, 40(10):1145–1150, 1999.
- [Ede61] M. Eden. In F. Neyman, ed. *Proc. 4th Berkeley Symp. on Mathematics, Statistics and Probability*, Vol. 4, p. 223. University of California Press, Berkeley, 1961.
- [FHP86] U. Frisch, B. Hasslacher, and Y. Pomeau. Lattice-gas automata for the Navier–Stokes equation. *Phys. Rev. Lett.*, 56(14):1505–1508, 1986.
- [Fla01] A. Flache. Do irregular grids make a difference? Relaxing the spatial regularity assumption in cellular models of social dynamics. *JASSS–J. Artif. So. S.*, 4(4), 2001. <http://jasss.soc.surrey.ac.uk/4/4/6.html>.
- [GGCH04] G. Guillemot, Ch-A. Gandin, H. Combeau, and R. Heringer. A new cellular automaton-finite element coupling scheme for alloy solidification. *Model. Simul. Mater. Sci. Eng.*, 12:545–556, 2004.
- [HG91] H. W. Hesselbarth and I. R. Gobel. Simulation of recrystallization by cellular automata. *Acta Metall. Mater.*, 39(9):2135–2143, 1991.
- [HKH93] H. Hesselbarth, L. Kaps, and F. Haessner. *Mater. Sci. Forum*, 317:113–115, 1993.
- [HPdP73] J. Hardy, Y. Pomeau, and O. de Pazzis. Time evolution of a two-dimensional model system. i. Invariant states and time correlation functions. *J. Math. Phys.*, 14(12):1746–1759, 1973.
- [Jan03] K. G. F. Janssens. Three dimensional, space-time coupled cellular automata for the simulation of recrystallization and grain growth. *Model. Simul. Mater. Sci.*, 11(2):157–171, 2003.
-

- [JB85] R. Jullien and R. Botet. Scaling properties of the surface of the eden model in $d=2, 3, 4$. *J. Phys. A*, 18:2279–2287, 1985.
- [JHF04] K. G. F. Janssens, E. A. Holm, and S. M. Foiles. Introducing solute drag in irregular cellular automata modeling of grain growth. In *Proceedings of the 2nd International Conference on Recrystallization and Grain Growth, Annecy, France, Recrystallization and Grain Growth*, Vol. 2, pp. 1045–1050. Trans Tech Publications, Zurich, Switzerland, 2004.
- [JVR02] K. G. F. Janssens, F. Vanini, and J. N. Reissner. Thermodynamic and kinetic coupling of a random grid cellular automaton for the simulation of grain growth. *Adv. Eng. Mater.*, 4(4):200–202, 2002.
- [KL04] M. S. Khvastunkov and J. W. Leggoe. Adapting cellular automata to model failure in spatially heterogeneous ductile alloys. *Scripta Mater.*, 51:309–314, 2004.
- [Kre98] K. Kremeyer. Cellular automata investigations of binary solidification. *J. Comp. Phys.*, 142:243–262, 1998.
- [LA04] Q. Li and P. M. Anderson. A three-dimensional cellular automaton model of dislocation motion in FCC crystals. *Modelling Simul. Mater. Sci. Eng.*, 12:929–943, 2004.
- [LB01] D. P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, Cambridge, 2001.
- [LBP96] Y. Liu, T. Baudin, and R. Penelle. Simulation of normal grain growth by cellular automata. *Scripta Mater.*, 34(1):1679–1683, 1996.
- [LL03] G. R. Liu and M. B. Liu. *Smoothed Particle Hydrodynamics*. World Scientific, London, 2003.
- [LLHL04] Y. J. Lan, D. Z. Li, C. J. Huang, and Y. Y. Li. A cellular automaton model for austenite to ferrite transformation in carbon steel under non-equilibrium interface conditions. *Model. Simul. Mater. Sci. Eng.*, 12:719–729, 2004.
- [LLL06] Y. J. Lan, D. Z. Li, and Y. Y. Li. A mesoscale cellular automaton model for curvature driven grain growth. *Metall. Mater. Trans. B*, 37:119–129, 2006.
- [Mat] Mathematica. <http://www.wolfram.com/>.
- [McC94] J. F. McCarthy. Lattice gas cellular automata method for flow in the interdendritic region. *Acta Metall. Mater.*, 42(5):1573–1581, 1994.
- [MRG99] V. Marx, F. R. Reher, and G. Gottstein. Simulation of primary recrystallization using a modified three-dimensional cellular automaton. *Acta Mater.*, 47(4):1219–1230, 1999.
- [OCSBY01] B. Ostrovsky, G. Crooks, M. A. Smith, and Y. Bar-Yam. Cellular automata for polymer simulation with application to polymer melts and polymer collapse including implications for protein folding. *Parallel Computing*, 27:613–641, 2001.
- [PD94] C. F. Pezzee and D. C. Dunand. The impingement effect of an inert, immobile second phase on the recrystallization of a matrix. *Acta Metall. Mater.*, 42(5):1509–1524, 1994.
- [Raa02] D. Raabe. Cellular automata in materials science with particular reference to recrystallization simulation. *Annu. Rev. Mater. Res.*, 32:53–76, 2002.
- [Raa04a] D. Raabe. Mesoscale simulation of spherulite growth during polymer crystallization by use of cellular automaton. *Acta Mater.*, 52:2653–2664, 2004.
- [Raa04b] D. Raabe. Overview of the lattice Boltzmann method for nano- and microscale fluid dynamics in materials science and engineering. *Model. Simul. Mater. Sci.*, 12:R13–R46, 2004.
- [RR01] A. D. Rollett and D. Raabe. A hybrid model for mesoscopic simulation of recrystallization. *Comp. Mater. Sci.*, 21:69–78, 2001.
- [RZ97] D. H. Rothman and S. Zaleski. *Lattice-Gas Cellular Automata—Simple Models of Complex Hydrodynamics*. Cambridge University Press, Cambridge, 1997.
- [SCSB04] J. Saunier, A. Chauss, J. Stafiej, and J. P. Badali. Simulations of diffusion limited corrosion at the metal–environment interface. *J. Electroanalytical Chemistry*, 563:239–247, 2004.
- [STH97] T. P. Swiler, V. Tikare, and E. A. Holm. *Mater. Sci. Eng. A*, 238:85, 1997.
- [Suc01] S. Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Clarendon Press, Oxford, 2001.
- [VG02] M. Vandyoussefi and A. L. Greer. Application of cellular automaton-finite element model to the grain refinement of directionally solidified Al-4.15 wt% Mg alloys. *Acta Mater.*, 50:1693–1705, 2002.
- [Wol00] D. A. Wolf-Gladrow. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models*. Springer-Verlag, Berlin, 2000.
- [Wol02] S. Wolfram. *A New Kind of Science*. <http://www.wolframscience.com/>, 2002.

- [XL01] Q. Y. Xu and B. C. Liu. Modeling of as-cast microstructure of Al alloy with a modified cellular automaton method. *Mater. Trans.*, 42(11):2316–2321, 2001.
- [ZCCT01] J. Zhu, L.-Q. Chen, J. Chen, and V. Tikare. Microstructure dependence of diffusional transport. *Comp. Mater. Sci.*, 20:37–47, 2001.
- [ZWZ⁺03] L. Zhang, Y. M. Wang, C. B. Zhang, S. Q. Wang, and H. Q. Ye. A cellular automaton model of the transformation from austenite to ferrite in low carbon steels. *Model. Simul. Mater. Sci. Eng.*, 11:791–802, 2003.
- [ZZXLW02] L. Zhang, C. Zhang, G. Wang X. Liu, and Y. Wang. Modeling recrystallization of austenite for C-Mn steels during hot deformation by cellular automaton. *J. Mater. Sci. Technol.*, 18(2):163–166, 2002.

5 Modeling Solid-State Diffusion

—*Ernst Kozeschnik*

Diffusion is a phenomenon which we face in everyday life on so many occasions that we rarely take notice of it, and it is a physical process that comes into play in almost all disciplines of CME. Diffusion concerns the transport of manifold things, being either material and easy to recognize or immaterial and sometimes difficult to observe. Diffusion is a phenomenon that can move matter from one place to another as well as properties, information, or knowledge. And diffusion is experienced as a macroscopic phenomenon although the mechanisms of diffusion are founded in microscopic exchange and/or collision processes.

When treating solid-state diffusion, the process of redistribution of matter can be looked at from two different length scales:

1. When considering the movement of individual atoms through the solid, we seek physical laws that give a quantitative description of the migration kinetics of individual atoms; for example, the mean root square displacement in random walks. We trace the movement of these atoms through the material and quantify the microscopic diffusion fluxes.
2. When investigating diffusion in macroscopic pieces of matter, we look for a quantitative description of the macroscopically observed diffusion fluxes. The transport of matter is treated as a consequence of a diffusion force acting on the atoms and accelerating them into the direction of the force.

In the following sections, the mechanisms of diffusion and its consequences will be reviewed on the microscopic and the macroscopic scale. Like in the previous chapter, we need not go into too much detail because a number of textbooks have dealt with theory and application of diffusion in great detail (see, e.g., refs. [She86, KY87, Gli00, Phi86, Cra75]). We will focus on these aspects of diffusion theory, which are important for problems in CME as discussed in this book.

5.1 Diffusion Mechanisms in Crystalline Solids

Solid-state mater is frequently constituted in the form of crystals, where atoms are arranged on a periodic lattice. The individual atoms vibrate around their ground state positions because of

thermal excitation. Their average location is fairly constant in time because each atom is caught in a potential trough. However, due to thermal excitation, there is a certain probability for the atom to escape and leave its current lattice position. To make this process happen, an energy barrier has to be overcome and the probability P for this process can be written as

$$P \approx \exp \frac{Q}{RT} \quad (5.1)$$

R is the universal gas constant, T is the absolute temperature, and Q is the *activation energy* for diffusion. The value of Q is typically a constant for a given material, crystal structure, and diffusion mechanism.

Figure 5-1 shows three possible mechanisms of how atoms can exchange lattice sites in a crystal. The left image sketches the *direct exchange mechanism*, which exchanges the positions of the two atoms I and II. To start the exchange process, the first atom I has to overcome the potential barrier of the periodic lattice and squeeze into an off-lattice position. The schematic potential diagram on the bottom of the figure indicates that this initial step requires a significant amount of activation. If the atom overcomes the barrier, it jumps out of its regular position into a neighboring interlattice position (step 1). It is evident that this configuration is only metastable since the atoms are now in a highly stressed state. The potential of the interlattice position is indicated by a dashed line. In the next step, atom II moves to the empty initial position of atom I and atom I moves from its interlattice position into the initial site of atom II (steps 2 and 3). Thus, the two atoms switch place and perform a single diffusion step. However, since the activation energy for the initial step of the direct exchange mechanism is rather high, the probability that a diffusion step occurs by this mechanism is rather small. The possible diffusion kinetics driven by direct exchange is much too slow to explain the macroscopically observed kinetics of diffusion.

Another proposed mechanism of diffusion is the ring-exchange mechanism. It is displayed in the center image of Figure 5-1. In the ring mechanism, four neighboring atoms rotate simultaneously around their middle position and thus exchange place. This mechanism is advantageous over the direct exchange mechanism for two atoms because the activation barrier is much lower. It is still not the observed predominant mechanism, as will be shown in Section 5.3.4.

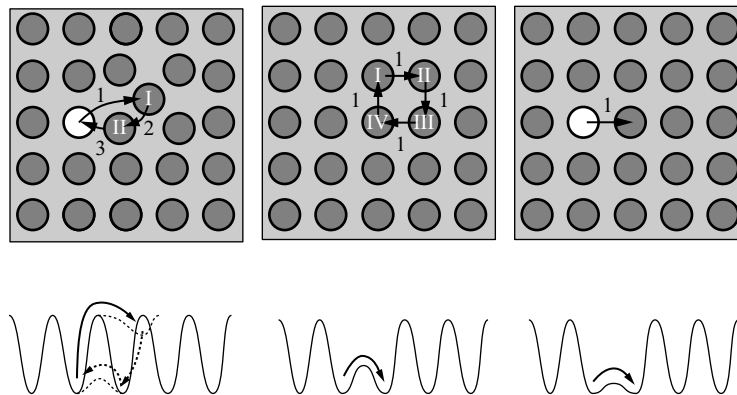


FIGURE 5-1 Three possible mechanisms driving solid-state diffusion: (left) Direct exchange mechanism. (middle) Ring mechanism. All atoms rotate simultaneously. (right) Vacancy exchange mechanism. The arabic numbers denote the sequence of jumps. Bottom: Schematic of interatomic potentials.

TABLE 5-1 Enthalpies for Vacancy Formation in Some Metals^a

	ΔH_{V_a}		ΔH_{V_a}
<i>fcc</i>	(kJ/mol)	<i>bcc</i>	(kJ/mol)
Al	65	Mo	289
Fe	135	Fe	154
Ni	172	Nb	256
Cu	124	W	386

^a From ref. [Gli00].

The right side of Figure 5-1 sketches a diffusion mechanism, which takes advantage of the fact that real crystals are never perfect and that they always contain a certain number of lattice imperfections. If, for instance, atom I is located next to a *substitutional vacancy*,¹ that is, a missing atom on a regular lattice site of the substitutional sublattice, the potential barrier for moving it into the empty lattice site is much lower than for a move into an interlattice site. By exchanging positions between vacancy and atom I, a single diffusion step is carried out since transport of this atom over one mean atomic distance has occurred. In addition, with this significantly lower activation energy, the probability for the occurrence of this last mechanism is orders of magnitude higher than the probability for direct exchange or even the ring-exchange mechanism. In the late 1940s, the *vacancy exchange mechanism* was undoubtedly identified as the predominant diffusion mechanism in solid materials (see Section 5.3.4). Diffusion occurring by this type of mechanism is also denoted as *vacancy-assisted diffusion*.

It is very important to recognize that diffusion in solids is inherently related to the movement of vacancies and that the diffusion kinetics are directly related to the density of vacant lattice sites. Consequently, the process of solid-state diffusion can also be envisaged as a random walk of vacancies through the crystal lattice. Each vacancy-atom exchange moves one atom over one atomic distance. The equilibrium mole fraction $X_{V_a}^{\text{eq}}$ of empty lattice sites can be expressed as

$$X_{V_a}^{\text{eq}} = \exp\left(-\frac{\Delta H_{V_a} - T\Delta S_{V_a}}{RT}\right) = \exp\left(-\frac{\Delta H_{V_a}}{RT}\right) \cdot \exp\left(\frac{\Delta S_{V_a}}{R}\right) \quad (5.2)$$

where ΔH_{V_a} is the enthalpy of formation of one mole of monovacancies and ΔS_{V_a} is the corresponding entropy of formation. The second exponential term in equation (5.2) is almost constant in metals, and we typically observe $\Delta S_{V_a} \approx 1R$ to $2R$. $X_{V_a}^{\text{eq}}$ is thus dominated by the enthalpy of formation term. We therefore find the temperature dependence of the equilibrium mole fraction of monovacancies in good approximation with

$$X_{V_a}^{\text{eq}}(T) = \exp\left(-\frac{\Delta H_{V_a}}{RT}\right) \quad (5.3)$$

Table 5-1 summarizes typical values for the enthalpy of vacancy formation in some metals, and Figure 5-2 displays the equilibrium vacancy concentration over temperature. As a rule of thumb, in many low-melting metals, the vacancy mole fraction close to the melting point is $X_{V_a}^{\text{eq}} \approx 10^{-4}$.

So far, we have investigated the diffusion mechanisms of substitutional atoms in a periodic crystal lattice. For interstitial elements, the situation is different since, in metals, most of the

¹ Substitutional vacancies are not to be confused with interstitial vacancies, which have been introduced in Section 2.2.8 as a hypothetical component within a multisublattice thermodynamic solution model.

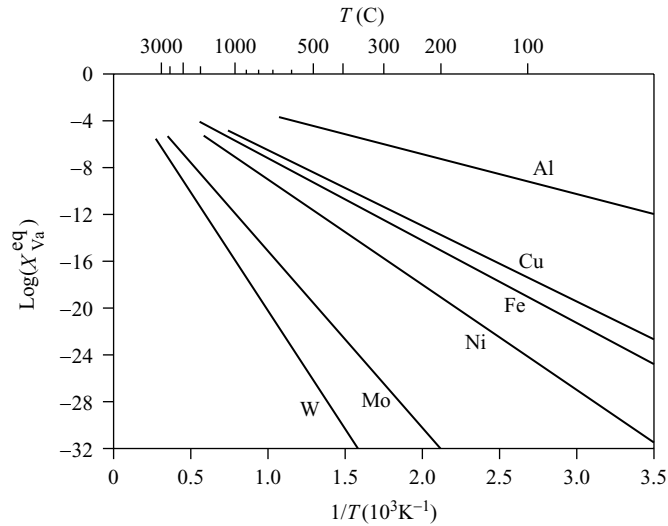


FIGURE 5-2 Equilibrium vacancy concentration in some metals.

interstitial lattice positions are usually vacant. The interstitial atoms can move within the crystal by simply overcoming the potential barriers of the periodic interstitial sublattice. The corresponding activation energy is significantly lower than for substitutional diffusion and interstitial diffusion is usually orders of magnitude faster than substitutional diffusion. Moreover, the temperature dependence of the diffusion kinetics is different because the density of vacant interstitial sites is *not* a function of temperature. The density of vacant interstitial sites is determined by the chemical composition of the material, and it is independent of the substitutional vacancy concentration. With decreasing temperature, substitutional diffusion becomes rapidly sluggish due to a rapidly decreasing *equilibrium* vacancy density (see Figure 5-2), whereas interstitial diffusion can still occur within a reasonable time scale.

5.2 Microscopic Diffusion

5.2.1 The Principle of Time Reversal

In the previous section we have identified the interchange of vacancies and atoms as the governing mechanism in solid-state diffusion. We have seen that each individual exchange process results in the transport of one single atom over one atomic distance. If looked at this process on the atomic length scale, one could conclude that the net transport of atoms when repeatedly exchanging atoms by this mechanism is zero because the probability of one atom being moved to the left is compensated by the equal probability of this atom being moved to the right. On an average basis, the atom should thus not move. However, when investigating this issue in more detail, we will find that the *random walk* of vacancies indeed leads to a net transport (flux) of atoms. Explanation of this phenomenon is found in the irreversible nature of diffusion, which has already been indicated in Section 2.2.1 in the discussion of the entropy of mixing.

As a thought experiment, let us record a sequence of 20 random vacancy jumps in a small ensemble of atoms (see Figure 5-3). If we play the sequence of jumps randomly in either forward or backward direction, it will be indistinguishable to us, whether the sequence is going

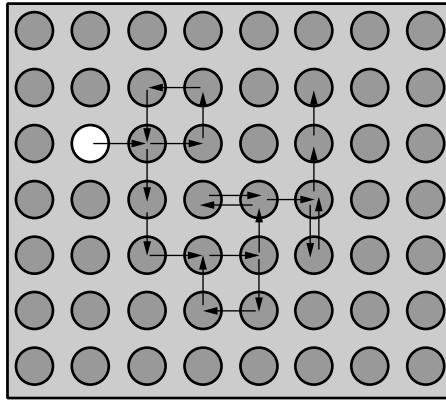


FIGURE 5-3 Random walk sequence of a vacancy traveling in a microscopic ensemble of atoms illustrating the principle of time reversal.

into the original direction or whether it is going backwards. Even if we mark the individual atoms of the ensemble, by simply looking at the sequence on a microscopic scale it is impossible to determine the *arrow of time*. The pattern of the forward and the backward sequence will look identical to us. This is the so-called *principle of time reversal*.

However, we have already recognized earlier in the discussion of the nature of entropy that, if a process is viewed on a sufficiently large scale, the arrow of time becomes visible and the process gains irreversible character. Figure 5-4 shows a sequence of concentration profiles in a macroscopic diffusion couple experiment where two initially pure substances (dark and white color) mix due to interdiffusion of the two substances. If we are exposed to the four images A–D, we will instantaneously be able to bring these into the right order A–B–C–D with the arrow of time going from A to D. From *experience* we know that we will never observe a different sequence. In this experiment, the arrow of time is easily identified and although the principle of time reversal is still valid on a microscopic scale, on a *macroscopic* scale, diffusion is an irreversible process.²

5.2.2 A Random Walk Treatment

In the previous sections, we have discussed diffusion on a qualitative basis. In this section we will *quantify* this process using statistical methods and we investigate the relation between diffusion on the microscopic and macroscopic scale in more detail.

Let us first consider k one-dimensional random walkers that move along the x -direction with constant steps of $\Delta\lambda = \pm 1$. Table 5-2 summarizes all possible sequences of one to four individual steps and lists the effective displacements of the walkers from the origin. The table shows that after four steps, the walkers are either back where they started at $\lambda = 0$ or they are found at the positions $\lambda = \pm 2$ or $\lambda = \pm 4$. If we repeat the experiment over and over again, the mean displacement $\langle \lambda_n \rangle$ after n steps, which is given by the mean value of the displacements of all individual k walkers, is

$$\langle \lambda_n \rangle = \frac{1}{k} \sum \lambda_k = 0 \quad (5.4)$$

² The apparent conflict between microscopic reversibility and macroscopic irreversibility is also known as *Maxwell's dilemma*. This scientific puzzle has been subject of controversy for many decades after it was introduced in 1867 by the Scottish physicist James Clerk Maxwell (1831–1879).

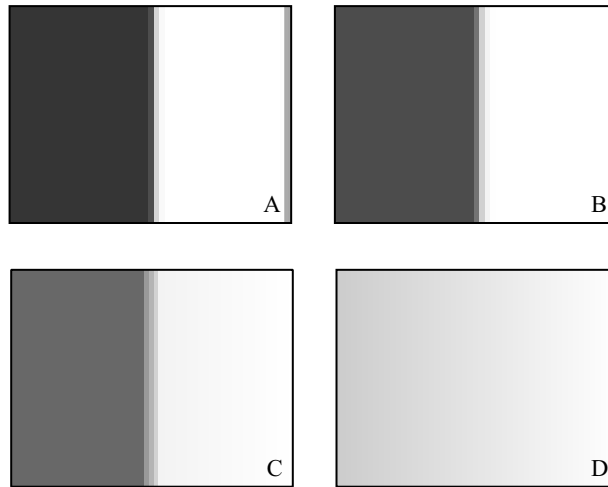


FIGURE 5-4 Diffusion couple experiment. Concentration profiles at different stages of the mixing process.

Apparently, $\langle \lambda_n \rangle$ does not tell anything about the magnitude of spreading of the walker entity. However, Table 5-2 shows that, although the mean displacement is zero, some of the walkers can be many steps away from the origin. To get a measure of this quantity, the *root mean square* (RMS) value of the individual displacements can be employed. The RMS is defined as the square root of the average value of the *squared* displacements $\langle \lambda_n^2 \rangle = 1/k \sum \lambda_k^2$ and it provides a measure of the *average straight-line distance* between the origin and the position of the walkers. For the RMS, we find (compare Table 5-2)

$$\sqrt{\langle \lambda_n^2 \rangle} = \sqrt{n}\lambda \quad (5.5)$$

Equations (5.4) and (5.5) are important results which tell us that, although the mean displacement in one spatial dimension is zero, the walkers nevertheless start spreading out into space.³ The region where we can expect to find them expands with time, that is, with the square root of the number of random walk steps. Figure 5-5 illustrates these results on the example of eight consecutive two-dimensional random walk sequences with six individual walkers. Each step has a length of $\sqrt{2}$, since the walkers move diagonal with $\lambda = \sqrt{\Delta x^2 + \Delta y^2}$ and $\Delta x = \pm 1$, $\Delta y = \pm 1$. The circles indicate the radius where we expect to find the walkers after 200 steps according to equation (5.5). From the figure, we can also see that the preceding relations are only valid on an average basis since we recognize that individual walkers can spread out far behind the expected value of the root mean square displacement.

PROBLEM 5-1: One-Dimensional Random Walk

Consider a fixed number of 250 one-dimensional random walkers. At time $t = 0$, let all walkers be at a position $x = 0$. Perform a given number of random steps of each walker with a distance of $\Delta x = \pm 1$. Plot the density of the walkers around the origin in a simple XY diagram. Compare the density distribution of walkers with a normalized Gaussian distribution.

³ Although the derivation of equation (5.5) has been demonstrated for the one-dimensional case, it can be shown that this relation is also valid in two and three dimensions.

TABLE 5-2 One-Dimensional Random Walk

n	Sequence	Displacement (λ)	Square displacement (λ^2)	Root mean square displacement ($\sqrt{1/k \sum \lambda^2}$)
1	+1	+1	1	$[(1/2) \cdot 2]^{1/2}$ $= (1)^{1/2}$
	-1	-1	1	
2	+1 + 1	+2	4	$[(1/4) \cdot 8]^{1/2}$ $= (2)^{1/2}$
	-1 + 1	0	0	
	+1 - 1	0	0	
	-1 - 1	-2	4	
3	+1 + 1 + 1	+3	9	$[(1/8) \cdot 24]^{1/2}$ $= (3)^{1/2}$
	-1 + 1 + 1	+1	1	
	+1 - 1 + 1	+1	1	
	+1 + 1 - 1	+1	1	
	-1 - 1 + 1	-1	1	
	-1 + 1 - 1	-1	1	
	+1 - 1 - 1	-1	1	
	-1 - 1 - 1	-3	9	
4	+1 + 1 + 1 + 1	+4	16	$[(1/16) \cdot 64]^{1/2}$ $= (4)^{1/2}$
	+1 - 1 + 1 + 1	+2	4	
	+1 + 1 - 1 + 1	+2	4	
	+1 + 1 + 1 - 1	+2	4	
	+1 - 1 - 1 + 1	0	0	
	+1 - 1 + 1 - 1	0	0	
	+1 + 1 - 1 - 1	0	0	
	+1 - 1 - 1 - 1	-2	4	
	-1 + 1 + 1 + 1	+2	4	
	-1 - 1 + 1 + 1	0	0	
	-1 + 1 - 1 + 1	0	0	
	-1 + 1 + 1 - 1	0	0	
	-1 - 1 - 1 + 1	-2	4	
	-1 - 1 + 1 - 1	-2	4	
-1 + 1 - 1 - 1	-2	4		
-1 - 1 - 1 - 1	-4	16		

5.2.3 Einstein's Equation

Diffusion is inherently related to the vibration of atoms around their ground state. If an atom experiences sufficient thermal activation, it can move to a neighboring lattice position.⁴ If the vibration frequency of the atom is ν and the atom has Z nearest neighbors, the total number of jump attempts is νZ . However, only a small fraction of the attempts will be successful, with a probability depending on the ratio between the necessary activation energy for a single jump Q_D and the thermal activation $k_B T$. The effective jump frequency Γ_D is then

⁴ Since the predominant diffusion mechanism in crystals is the vacancy exchange mechanism, you can equivalently apply the following treatment to the exchange of a vacancy with any of the neighboring atoms.

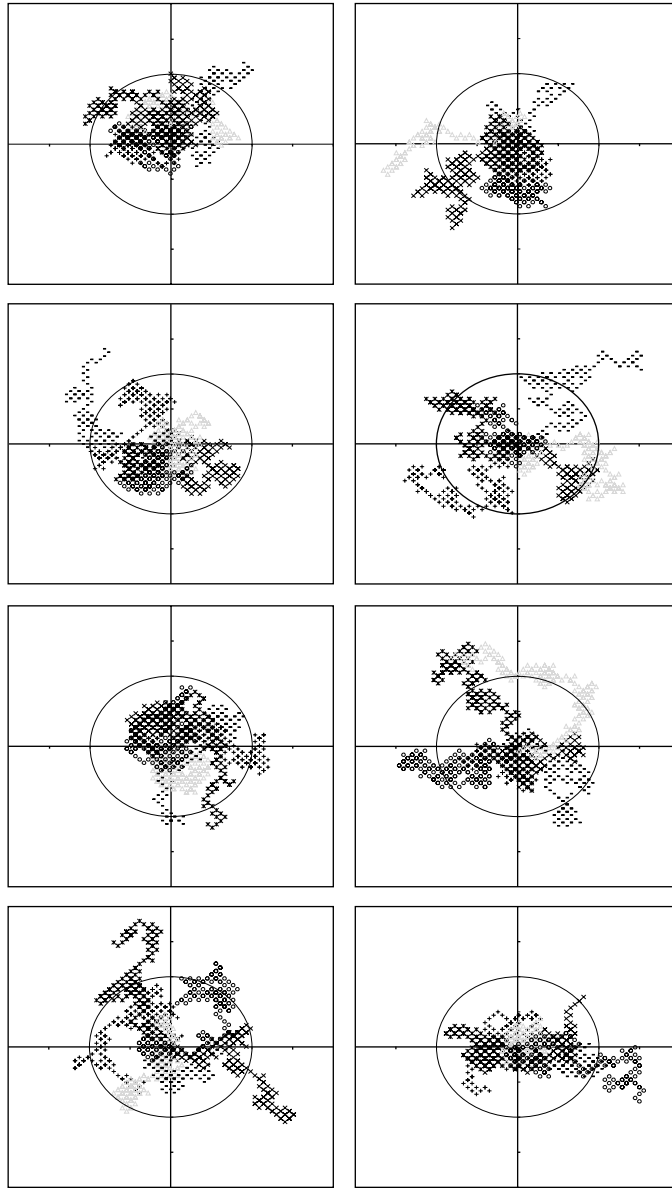


FIGURE 5-5 *Random walk of six particles with diagonal steps of $\lambda = \sqrt{2}$ and $n = 200$.*

$$\Gamma_D = \nu Z \exp\left(-\frac{Q_D}{k_B T}\right) \quad (5.6)$$

With each successful jump, the atom travels one atomic distance λ and the total traveling distance in unit time is thus $\Gamma_D \lambda$. Substituting the jump frequency Γ_D into the expression for the root mean square displacement of a random walker [equation (5.5)] and using the spatial coordinate r leads to

$$\langle r^2 \rangle = \Gamma_D \lambda^2 t \quad (5.7)$$

On the other hand, the root mean square displacement is also defined as the *second moment* of a distribution of values with

$$\langle r^2 \rangle = \int_0^\infty r^2 f(r, t) dr = \frac{1}{(4\pi Dt)^{d/2}} \int_0^\infty r^4 \exp\left(\frac{-r^2}{4Dt}\right) dr \quad (5.8)$$

$f(r, t)$ is the probability density and we have used equation (5.34), which will be derived in a later section. After substitution of

$$z = \frac{r^2}{4Dt} \quad (5.9)$$

the definite integral on the right-hand side of equation (5.8) can be evaluated with

$$\int_0^\infty z^4 \exp(-z^2) dz = \frac{3\sqrt{\pi}}{8} \quad (5.10)$$

and the RMS displacement of the spreading diffusion cloud is

$$\langle r^2 \rangle = 2dDt \quad (5.11)$$

d denotes the dimensionality (1, 2, or 3) of the problem. At this point we have to emphasize that this relation has been obtained from solution of Fick's second law for a three-dimensional point source, making use of the statistical definition of the root-mean-square value of a distribution. Equation (5.11) is thus a purely phenomenological relation expressing a macroscopic diffusion law. In the section on random walk, we have derived the RMS displacements during diffusion from microscopic considerations. Comparison of equations (5.7) and (5.11) leads to

$$\Gamma_D \lambda^2 = 2dD \quad (5.12)$$

It is thus possible to describe the macroscopic, phenomenological quantity D purely from atomistic quantities, such as the atomic jump frequency Γ_D and the mean atomic distance λ with

$$D = \frac{1}{2d} \Gamma_D \lambda^2 \quad (5.13)$$

or, vice versa, to express the atomic jump frequency Γ_D based on the macroscopic quantity D with

$$\Gamma_D = \frac{2dD}{\lambda^2} \quad (5.14)$$

Equation (5.12) goes back to Albert Einstein (German physicist, 1879–1955), who derived this famous relation in 1905 [Ein05]. By relating microscopic and macroscopic quantities of diffusion theory, the *Einstein equation* provides an important link for understanding the connection between microscopic reversibility and macroscopic irreversibility of diffusion.

PROBLEM 5-2: Multidimensional Random Walk

Consider a small number of five random walkers on a two-dimensional grid. At time $t = 0$, let all walkers be at the center position with $x = y = 0$. Perform a given number of random steps ($n = 100$ to 1000) for each walker over a distance of $\Delta x = \pm 1$ and $\Delta y = \pm 1$. Plot the path of the walkers around the origin in a simple XY diagram. Compare with the RMS displacement as given in equation (5.5). Compare the results also with Figure 5-5.

5.3 Macroscopic Diffusion

In this chapter we are concerned with the phenomenological theory of diffusion, that is, the laws of diffusion for the transport of atoms in materials on a macroscopic scale. We will introduce the reader to the phenomenological diffusion laws as found by A. Fick. We will present a few classic analytical solutions to Fick's second law and discuss a finite differences algorithm to solve general diffusion problems. A discussion of multicomponent diffusion coefficients for application in materials modeling complements this chapter.

5.3.1 Phenomenological Laws of Diffusion

On investigating the diffusive mixing of salt and water, the German physiologist Adolf Eugen Fick (1829–1901) found that the diffusive flux J of atoms is proportional to the gradient in concentration c of the diffusing species. Fick published his result in 1855 [Fic55] and accordingly, for the one-dimensional case with the spatial coordinate r , we have

$$J = -D \frac{\partial c}{\partial r} \quad (5.15)$$

Equation (5.15) is known as *Fick's first law*. The proportionality constant D is known as the *diffusion coefficient*. It is important to recognize that this law was deduced from pure observation and empirical facts. It is thus a phenomenological equation and it represents the mathematical relation between the diffusive flux and the observed concentration gradient of the diffusing species.

If we consider diffusion in multiple dimensions, the diffusive flux becomes a vector and with the nabla operator ∇ defined as

$$\nabla = \begin{pmatrix} \partial/\partial x \\ \partial/\partial y \\ \partial/\partial z \end{pmatrix} \quad (5.16)$$

we obtain the general form of Fick's first law as⁵

$$J = -D \nabla c \quad (5.17)$$

In diffusion processes without sinks and sources for atoms, mass is a conserved quantity. Consider a point P in space surrounded by a small rectangular control volume $\Delta x \Delta y \Delta z$ (see Figure 5-6) and the one-dimensional fluxes J_x^{in} and J_x^{out} . If the two fluxes are equal, the same amount of matter enters and exits the control volume and the concentration of atoms remains constant. If J_x^{in} is different from J_x^{out} , mass will be accumulated or dispersed. The accumulated number of moles of atoms is then

⁵ Note that bold symbols represent vector quantities.

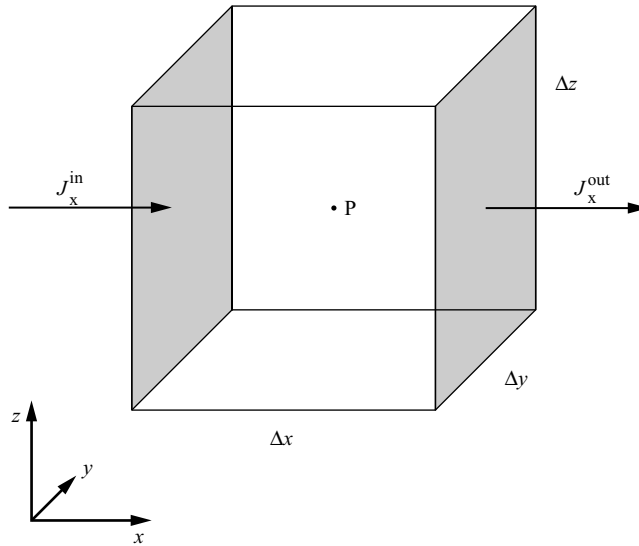


FIGURE 5-6 Mass conservation in diffusion.

$$\Delta \text{moles} = (J_x^{\text{in}} - J_x^{\text{out}}) \Delta y \Delta z \cdot \Delta t \quad (5.18)$$

On the other hand, the concentration c of atoms in the control volume is

$$c = \frac{\# \text{moles}}{\Delta x \Delta y \Delta z} \quad (5.19)$$

and the change of concentration is

$$\Delta c = \frac{\Delta \# \text{moles}}{\Delta x \Delta y \Delta z} \quad (5.20)$$

After equating (5.18) and (5.20) and using $\Delta J = J_x^{\text{out}} - J_x^{\text{in}}$, we arrive at

$$\frac{\Delta c}{\Delta t} = - \frac{\Delta J}{\Delta x} \quad (5.21)$$

Generalizing to three dimensions and using differentials instead of differences gives

$$\frac{\partial c}{\partial t} = -\nabla J \quad (5.22)$$

and inserting equation (5.17) leads to

$$\frac{\partial c}{\partial t} = -\nabla (-D \nabla c) \quad (5.23)$$

With the assumption that the diffusion coefficient D is independent of composition, we finally have

$$\frac{\partial c}{\partial t} = D \nabla^2 c \quad (5.24)$$

Equation (5.24) is known as *Fick's second law*. It is a second-order linear partial differential equation describing transient diffusion processes, and it is interesting that Fick's second law follows directly from Fick's first law and mass conservation considerations.

From this general form of the diffusion equation, Fick's second law can also be written in planar, cylindrical, and spherical coordinates by substituting for the Laplace operator ∇^2 . Using the symbols x , y , and z for cartesian coordinates, r , θ , and z for cylindrical coordinates, and r , θ , and ϕ for spherical coordinates, the diffusion equation reads

$$\frac{\partial c}{\partial t} = D \left(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} + \frac{\partial^2 c}{\partial z^2} \right) \quad (5.25)$$

$$\frac{\partial c}{\partial t} = \frac{D}{r} \left[\frac{\partial}{\partial r} \left(r \frac{\partial c}{\partial r} \right) + \frac{1}{r} \frac{\partial^2 c}{\partial \theta^2} + r \frac{\partial^2 c}{\partial z^2} \right] \quad (5.26)$$

$$\frac{\partial c}{\partial t} = \frac{D}{r^2} \left[\frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) + \frac{1}{\sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial c}{\partial \theta} \right) + \frac{1}{\sin^2 \theta} \frac{\partial^2 c}{\partial \phi^2} \right] \quad (5.27)$$

When considering diffusion in only one dimension, that is, linear diffusion with rotational symmetry, and the spatial variable r , equations (5.25)–(5.27) reduce to

$$\frac{\partial c}{\partial t} = D \left(\frac{\partial^2 c}{\partial r^2} \right) \quad (5.28)$$

$$\frac{\partial c}{\partial t} = \frac{D}{r} \left[\frac{\partial}{\partial r} \left(r \frac{\partial c}{\partial r} \right) \right] \quad (5.29)$$

$$\frac{\partial c}{\partial t} = \frac{D}{r^2} \left[\frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) \right] \quad (5.30)$$

Equations (5.28)–(5.30) can be further reduced when substituting $\lambda = r/\sqrt{t}$. The general form of the linear diffusion equations then becomes

$$-\frac{\lambda}{2} \frac{dc}{d\lambda} = \frac{D}{\lambda^{p-1}} \frac{d}{d\lambda} \left[\lambda^{p-1} \frac{dc}{d\lambda} \right] \quad (5.31)$$

which is an ordinary parabolic differential equation in the independent variable λ . The index p has the values 1, 2, and 3 for planar, cylindrical, and spherical geometry. For the case of $p = 1$, this equation is also known as the *Boltzmann diffusion equation*.

5.3.2 Solutions to Fick's Second Law

In this section, we will briefly review some important solutions to Fick's second law [equation (5.24)].

Spreading of a Diffusant from a Point Source

Consider one-dimensional diffusion of a mass M along the x -direction in an infinite sample under the assumption that the entire mass is initially concentrated in a single plane. According to the law of mass conservation, we have

$$\int_{-\infty}^{+\infty} c(x, t) dx = 1 \quad (5.32)$$

The solution to equation (5.24) with the constraint (5.32) is closely related to the problem of random walk of atoms and it is elaborated in many textbooks (e.g., ref. [Gli00]). Accordingly, for unit amount of mass, we have

$$c(x, t) = \frac{M}{2\sqrt{\pi Dt}} \exp\left(\frac{-x^2}{4Dt}\right) \quad (5.33)$$

Equation (5.33) shows that, if atoms spread according to Fick's second law, they will form a Gaussian distribution (see Figure 5-7). The general solution to a unit mass swarm of atoms spreading into an infinite sample of dimension d is

$$c(r, t) = \frac{1}{(4\pi Dt)^{d/2}} \exp\left(\frac{-r^2}{4Dt}\right) \quad (5.34)$$

where r is the distance from the point source. Equation (5.34) represents the *probability distribution* for a diffusant spreading into an infinite sample of dimension d from a general point source. This equation has also been used in the derivation of Einstein's equation in Section 5.2.3.

Diffusion into a Semi-infinite Sample

Consider one-dimensional diffusion of a diffusant into a semi-infinite sample. This is a different situation now compared to the spreading diffusant from before, since we assume a continuous supply of atoms entering the sample from the outside boundary. This type of boundary condition is typical for surface treatments, such as carburization. Again, we start from Fick's second law. The present boundary conditions are now a constant concentration of atoms at the semi-infinite boundary, that is,

$$c(0, t) = c_0 \quad (5.35)$$

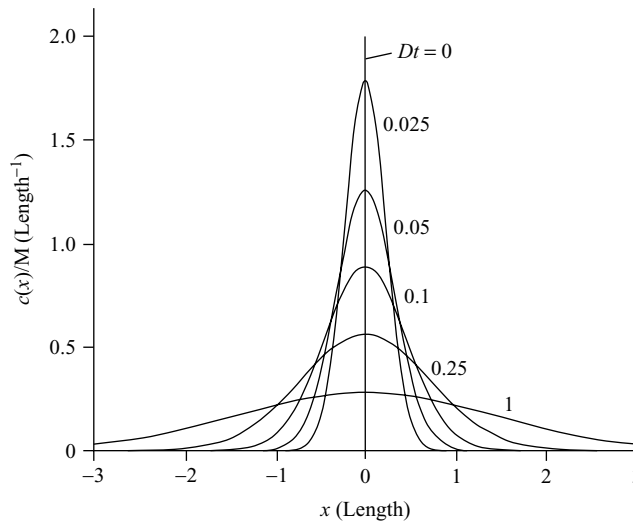


FIGURE 5-7 Solutions to Fick's second law for a point source of diffusant in an infinite sample.

The solution for this diffusion problem is

$$c(x, t) = c_0 \cdot \operatorname{erfc}\left(\frac{x}{\sqrt{4Dt}}\right) \quad (5.36)$$

with the *error function*, $\operatorname{erf}(z)$ and its complement, $\operatorname{erfc}(z)$, defined as

$$\operatorname{erfc}(z) = 1 - \operatorname{erf}(z) = 1 - \frac{2}{\sqrt{\pi}} \int_0^z e^{-\eta^2} d\eta \quad (5.37)$$

Figure 5-8 displays the evolution of the concentration profile during diffusion of solute into the semi-infinite sample. Note that all these curves degrade into one self-similar curve if the variable $x/\sqrt{4Dt}$ is plotted on the x -axis [compare equation (5.36)].

5.3.3 Diffusion Forces and Atomic Mobility

If a force F acts on a body, according to Newtonian mechanics, this force accelerates the body along the direction of the force. Typical examples of such forces are, for instance, gravitation acting on a mass or an electric field acting on a charged particle. Generally, a force can be written as the gradient of a potential Φ and we have

$$F \propto \nabla\Phi \quad (5.38)$$

In diffusion, the force F acting on one mole of atoms is identified as the gradient of the chemical potential μ and we have

$$F = -\nabla\mu \quad (5.39)$$

On an atomistic level, the force F accelerates the atoms along the direction of the force. However, the majority of atoms will resist this force because, according to the findings of Section 5.1, they can only migrate in the presence of a vacancy. If we interpret the resistance

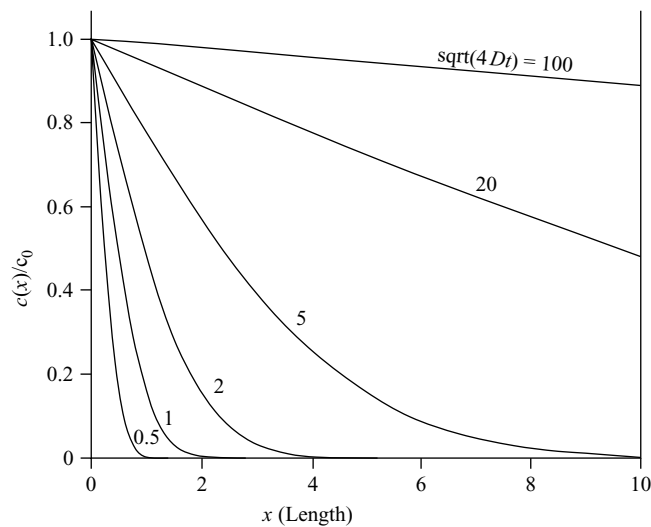


FIGURE 5-8 Solutions to Fick's second law for semi-infinite sample.

against movement in terms of *friction* with the coefficient of friction g_f , we find the effective drift velocity u of a large number of atoms under the force F from the force balance and we can write

$$F = g_f u \quad (5.40)$$

With the mobility B , which is generally defined as the inverse of the friction coefficient $B = 1/g_f$, we have

$$u = BF = -B\nabla\mu \quad (5.41)$$

The mobility B is an intrinsic property of the atomic species in a material and it defines the response of the species to a force F , that is, the gradient in chemical potential $\nabla\mu$. The flux J of atoms passing through unit area in unit time is then

$$J = cu = -cB\nabla\mu = -V_m X B \nabla\mu \quad (5.42)$$

The unit of J is moles/m² s and c is the concentration in moles/m³. V_m is the molar volume and X is the mole fraction of the atomic species. In general solutions, the chemical potential is defined in terms of the activity a [see equation (2.66)] with

$$\mu = {}^0\mu + RT \ln a \quad (5.43)$$

and with the definition of the activity coefficient $a = fX$ [compare equation (2.67)] we obtain

$$\nabla\mu = RT (\nabla \ln f + \nabla \ln X) \quad (5.44)$$

Equation (5.44) can be rearranged as

$$\nabla\mu = \frac{RT}{X} \left(\frac{X\nabla \ln f}{\nabla X} + 1 \right) \nabla X \quad (5.45)$$

or

$$\nabla\mu = \frac{RT}{X} \left(\frac{\nabla \ln f}{\nabla \ln X} + 1 \right) \nabla X \quad (5.46)$$

When substituting equation (5.46) into equation (5.42) and replacing the nabla operator by partial differentials, we obtain

$$J = -V_m X B \nabla\mu = -BRT \left(1 + \frac{\partial \ln f}{\partial \ln X} \right) \nabla c \quad (5.47)$$

When comparing with the definition of Fick's first law [equation (5.17)], we identify the relation between the diffusion coefficient D and the mobility B as

$$D = BRT \left(1 + \frac{\partial \ln f}{\partial \ln X} \right) \quad (5.48)$$

In diffusion systems without chemical interaction between the atoms, for example, in ideal solutions or in systems with radioactive isotopes, the activity coefficient $f = 1$ is constant and the term in brackets is unity. In this case, equation (5.48) reduces to

$$D = RTB \quad (5.49)$$

Equation (5.48) shows that the *macroscopic* diffusion coefficient D , from its purely phenomenological definition of Fick's first law, is directly related to the *microscopic* mobility B , which has been defined based on atomistic considerations, by a *thermal* factor RT and a *chemical* contribution, which describes the deviation from solution ideality in form of the logarithmic derivative of the activity coefficient f .

An interesting consequence of the preceding considerations is that, even in the absence of chemical interactions between the species, atoms experience a force which causes a net flux of atoms. Since this force cannot be measured directly, it is often considered as a *generalized* force. In an ideal solution, the generalized force F is directly proportional to the concentration gradient. In nonideal solutions, the chemical contribution can be included in the thermodynamic factor ϕ , which is

$$\phi = 1 + \frac{\partial \ln f}{\partial \ln X} \quad (5.50)$$

and the relation between D and B in general solutions can be written as

$$D = RTB\phi \quad (5.51)$$

Although equation (5.51) is of general validity, we have to be aware of the fact that the derivative in the expression for the thermodynamic factor [equation (5.50)] is taken with respect to the mole fraction variable X . We have recognized already in Section 2.1.3, that, in this case, the constraint $\sum X_i = 1$ [equation (2.22)] must be taken into account and the variation ∂X_i of one element can only be performed against the equivalent variation $-\partial X_{\text{ref}}$ of some reference element.

Consider a binary system A–B. From equation (5.42), the flux of atoms A is

$$J_A = -V_m X_A B \nabla \mu_A \quad (5.52)$$

Using $X_A + X_B = 1$ and $\nabla X_A = -\nabla X_B$, the gradient in chemical potential $\nabla \mu_A$ is

$$\nabla \mu_A = \frac{\partial \mu_A}{\partial X_A} \nabla X_A + \frac{\partial \mu_A}{\partial X_B} \nabla X_B = \left(\frac{\partial \mu_A}{\partial X_A} - \frac{\partial \mu_A}{\partial X_B} \right) \nabla X_A \quad (5.53)$$

After multiplication with RT and rearranging, the flux of atoms A in the binary system is obtained with

$$J_A = -\frac{X_A}{RT} \left(\frac{\partial \mu_A}{\partial X_A} - \frac{\partial \mu_A}{\partial X_B} \right) RTB \nabla c_A \quad (5.54)$$

Comparison of equations (5.51) and (5.54) shows that the thermodynamic factor ϕ in a binary system and in terms of mole fraction variables X_i is

$$\phi_A = \frac{X_A}{RT} \left(\frac{\partial \mu_A}{\partial X_A} - \frac{\partial \mu_A}{\partial X_B} \right) \quad (5.55)$$

or in a general multicomponent solution with the reference component indicated by the subscript “ref”

$$\phi_i = \frac{X_i}{RT} \left(\frac{\partial \mu_i}{\partial X_i} - \frac{\partial \mu_i}{\partial X_{\text{ref}}} \right) \quad (5.56)$$

The choice of reference component in equation (5.56) is somewhat arbitrary and needs further attention. If we consider the chemical contribution to the diffusion coefficient in a dilute alloy, it is reasonable to use a single reference component since the majority of all exchange processes will include this one single component. In concentrated alloys, however, the flux of component A in the laboratory frame of reference (see Section 5.3.4) is compensated by a flux of atoms of different kinds depending on the alloy composition. In this case, a weighted mean value instead of a single reference component is more appropriate, and the thermodynamic factor ϕ can then be expressed as

$$\phi_i = \frac{X_i}{RT} \left(\frac{\partial \mu_i}{\partial X_i} - \frac{1}{1 - X_i} \sum_{j \neq i} X_j \frac{\partial \mu_j}{\partial X_j} \right) \quad (5.57)$$

An interesting question that has not yet been addressed is: How can we measure the diffusion coefficient D in a single-component system? Analysis of concentration profiles and net diffusional fluxes is apparently impossible, if we are unable to distinguish between the individual atoms and if we are unable to trace their paths through the material. A common technique for “marking” individual atoms of the same species is to use radioactive isotopes. The advantage of this technique is that isotopes are chemically near-identical and thus do not introduce additional chemical interactions. The concentration profiles in corresponding experiments are evaluated as a function of time and penetration depth using mechanical sectioning and radioactivity measurement. The diffusion coefficient is obtained from straightforward comparison with appropriate solutions to Fick’s second law [e.g., equation (5.36)]. The diffusion coefficient in a single-component system is denoted as the *self-diffusion* coefficient D^* . The self-diffusion coefficient gives a measure for the effective displacement of atoms caused by random vacancy movement.

The radioactive isotope technique can likewise be used to measure “self-diffusion” coefficients in solutions of multiple components. Accordingly, the specimen is prepared as a chemically homogeneous solution of given composition and some of the atoms are replaced by radioactive isotopes. The diffusion coefficient measured by this method is commonly denoted as the *tracer* or *impurity* diffusion coefficient. The same symbol D^* is commonly used for this quantity and, due to the absence of chemical interactions, the general relation between atomic mobility B and D^* holds⁶

$$D^* = RTB \quad (5.58)$$

When finally looking at the thermodynamic factor in the asymptotic limit of dilute solution, we have to recall the analysis of solution thermodynamics in the dilute solution limit (Section 2.2.5). With the help of the regular solution model, we have found that the activity coefficient f approaches a constant value if the solute content goes to zero (Henry’s law). Consequently, the logarithmic derivative of the activity coefficient in the definition of the thermodynamic factor ϕ in equation (5.50) becomes zero and the thermodynamic factor thus approaches unity. The importance of the thermodynamic factor comes into play only in concentrated alloys, where ϕ accounts for the influence of solution nonideality.

⁶ When measuring the diffusion coefficient based on tracer elements, this value is always smaller than the *true* self-diffusion coefficient, which is defined on basis of random vacancy-atom exchanges. This effect is known as the *correlation effect*. The factor relating correlated and uncorrelated jumps is a constant for each type of crystal lattice and it is always less than unity (e.g., $f_{\text{bcc}} = 0.727$ and $f_{\text{fcc}} = 0.781$). In this book, we will not further distinguish between the two and assume that the correlation effect is implicitly taken into account.

5.3.4 Interdiffusion and the Kirkendall Effect

In Section 5.3.3, the net motion of atoms that is caused by a generalized diffusion force has been discussed. The chemical potential has been identified as the source of this driving force, and the thermodynamic factor has been introduced to account for nonideal thermodynamic behavior. In this section diffusion will be analyzed in situations where diffusion of multiple atomic species occurs simultaneously.

In the 1940s, a severe controversy about the mechanism of diffusion in crystalline solids was going on. One group of scientists promoted the traditional view of diffusion, which assumed that the diffusional transport of atoms occurs on basis of an atom by atom exchange mechanism. Thereby, one atom exchanges place with another atom based on direct exchange (see Figure 5-1) or the ring-exchange mechanism, which involves four atoms that rotate simultaneously and thus change place. The second group of scientists believed that diffusion occurs by the vacancy-exchange mechanism, that is, atoms switch place only with a neighboring empty lattice site (vacancy), and the transport of atoms occurs as a consequence of the random walk of these vacancies. In fact, the type of diffusion mechanism has substantial influence on the rate of diffusion of individual atomic species. Consequently, by careful analysis of appropriate diffusion experiments, it should be possible to identify the predominating diffusion mechanism in solid-state matter. These experiments and the corresponding theoretical analysis will now be discussed.

Consider a binary diffusion couple with the pure substances A and B. After bringing A and B into contact, the A atoms will spread into the B-rich side and vice versa. If diffusion occurs by direct atomic exchange, the macroscopically observed diffusivities of the two atomic species must be identical, because one single exchange process moves the same amount of A and B atoms and diffusion of A and B occurs at the same rate. In contrast, if diffusion is carried by atom/vacancy exchange, the A and B atoms can move independently and the diffusivity of the two species can be different.

Moreover, if one species diffuses from the left to the right side of the diffusion couple by vacancy/atom exchange, the flux of atoms must be accompanied by a counterflux of vacancies. If the diffusivities of A and B are identical, the two vacancy fluxes balance and annihilate. In the case where the diffusivities differ, a net flux of vacancies must occur, which “blows” through the sample. The net flux of vacancies is commonly known as *vacancy wind*. As a result of the vacancy wind, the lattice of the sample moves in the parallel direction of the vacancy flux. If, by some experimental technique, individual lattice planes, for example, the initial contact area of the diffusion couple, are marked, movement of the lattice planes can be recorded as a function of the difference of the diffusive fluxes of the A and B atoms. These experiments have been carried out by Ernest Kirkendall (1914–2005), an American metallurgist, between 1939 and 1947.

In a series of three papers [KTU39, Kir42, SK47], Kirkendall and co-workers investigated the diffusion of copper and zinc in brass. In the experiments of the third paper, the initial contact plane of a diffusion couple between pure copper and brass with 70wt% copper/30wt% zinc has been marked with thin molybdenum wires, such that the movement of the inert markers can be observed at different stages of the experiment. Kirkendall showed that the markers moved relative to the *laboratory* frame of reference, which is a reference frame that is fixed to the sample surrounding. An experimentalist looking at a diffusion couple will observe the diffusion process in the laboratory frame of reference.

From the results of the experiments, Kirkendall drew the following two conclusions:

1. the diffusion of zinc is much faster than the diffusion of copper, and
2. the movement of the markers is related to the difference in the diffusion coefficients.

The results of his experiments have been published against the strong resistance of individual researchers, in particular Robert Franklin Mehl, an American metallurgist (1898–1976). Nowadays, the observed *Kirkendall drift* of the marker plane is considered to be the first striking proof of the predominance of the vacancy exchange mechanism over direct atomic exchange in diffusion.

A short time after publication of Kirkendall's papers, L. Darken [Dar48] published the first quantitative analysis of Kirkendall's experiments, which will briefly be outlined later. Consider the two *intrinsic* fluxes J_A and J_B (i.e., the fluxes that are observed when looking at diffusion from a frame of reference that is fixed to an individual lattice plane, the *lattice* frame of reference) according to Figure 5-9. In the steady state case, using Fick's first law, we have

$$\begin{aligned} J_A &= -D_A \frac{\partial c_A}{\partial r} \\ J_B &= -D_B \frac{\partial c_B}{\partial r} \end{aligned} \quad (5.59)$$

The net flux of atoms across this lattice plane J_{net} is given as the sum of the intrinsic fluxes of the components and we obtain

$$J_{\text{net}} = J_A + J_B = -D_A \frac{\partial c_A}{\partial r} - D_B \frac{\partial c_B}{\partial r} = -J_{V_a, \text{net}} \quad (5.60)$$

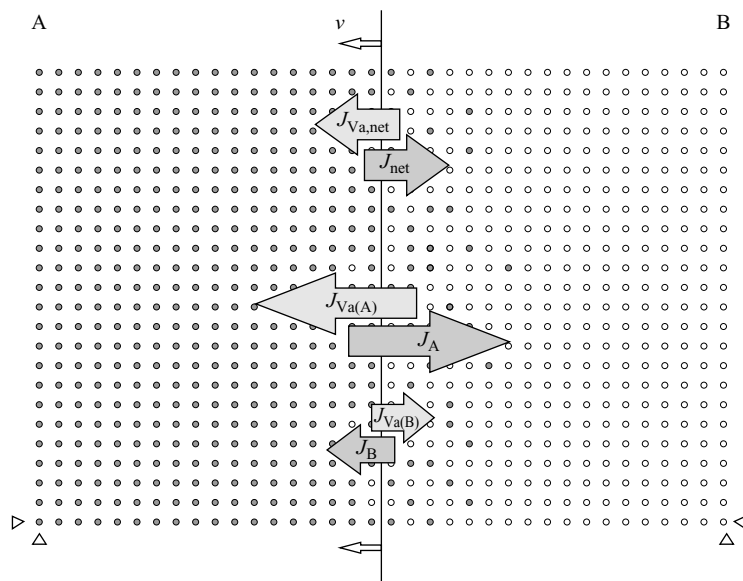


FIGURE 5-9 Schematic of the Kirkendall effect: Atoms A (gray circles) diffuse into the B-rich side at a higher rate than the B atoms (white circles) into the A-rich side. The net flux of atoms J_{net} causes a shift of the position of the initial contact plane with a velocity v relative to the laboratory frame of reference, that is, the fixed corners of the specimen. The movement of the marker plane is parallel to the net flux of vacancies $J_{V_a, \text{net}}$.

Equation (5.60) is known as *Darken's first equation*. From mass conservation, it is apparent that a net flux J_{net} of atoms causes accumulation of matter on one side of the marker plane. If mass is conserved, this accumulation must be compensated by a shift of the marker plane into the opposite direction. With the mole fraction $X_i = c_i V_m$ and with the assumption that the partial molar volumes $V_i = V_m$ of each element are identical, the velocity of the marker plane v is obtained with

$$v = J_{\text{Va,net}} V_m = \left(D_A \frac{\partial X_A}{\partial r} + D_B \frac{\partial X_B}{\partial r} \right) \quad (5.61)$$

Since we have the mole fraction constraint $X_A + X_B = 1$, the relation holds $dX_A = -dX_B$. Consequently, equation (5.61) can also be written in equivalent form as

$$\begin{aligned} v &= (D_A - D_B) \frac{\partial X_A}{\partial r} \\ v &= (D_B - D_A) \frac{\partial X_B}{\partial r} \end{aligned} \quad (5.62)$$

Equations (5.59) are defined in a coordinate system that is fixed to an individual lattice plane. This frame of reference can be transformed into the laboratory frame of reference with addition of a convective term or *drift term* with the convective flux $J_{\text{conv}} = vc$. This type of transformation is known as *Galilean transformation*. In the laboratory frame of reference, the flux for species A can be written as

$$\begin{aligned} J_A^{\text{lab}} &= -D_A \frac{\partial c_A}{\partial r} + c_A v \\ &= -D_A \frac{\partial c_A}{\partial r} + c_A (D_A - D_B) \frac{\partial X_A}{\partial r} \\ &= -D_A \frac{\partial c_A}{\partial r} + X_A D_A \frac{\partial c_A}{\partial r} - X_A D_B \frac{\partial c_A}{\partial r} \\ &= -(1 - X_A) D_A \frac{\partial c_A}{\partial r} - X_A D_B \frac{\partial c_A}{\partial r} \\ &= -[X_B D_A + X_A D_B] \frac{\partial c_A}{\partial r} \end{aligned} \quad (5.63)$$

On comparison with the flux equation (5.59) in the lattice frame of reference, the interdiffusion coefficient \tilde{D} is introduced with

$$\tilde{D} = X_B D_A + X_A D_B \quad (5.64)$$

Equation (5.64) is known as *Darken's second equation*. It is interesting to note that through an analysis of the concentration profiles in a diffusion couple experiment, only the interdiffusion coefficient \tilde{D} can be observed. To determine the individual intrinsic diffusion coefficients D_A and D_B , additional information is necessary, which is exactly the velocity or displacement of the marker plane.

The *interdiffusion* coefficient \tilde{D} describes the diffusive fluxes in the laboratory frame of reference, a convenient frame of reference for human experimentalists. The *intrinsic* diffusion coefficients D_A and D_B describe the fluxes in the lattice frame of reference, a convenient measure when operating on the atomic scale.

5.3.5 Multicomponent Diffusion

The concepts that have been introduced in the previous section for binary diffusion are now generalized to multicomponent systems. The diffusive flux of atoms has already been defined in Section 5.3.3 and for a component i we have

$$J_i = -c_i B_i \frac{\partial \mu_i}{\partial r} \quad (5.65)$$

This general form relates the diffusive flux to a mobility B and a generalized force $\partial\mu/\partial r$. Since the chemical potential $\mu = \mu(\mathbf{X})$ is a function of the local chemical composition \mathbf{X} , equation (5.65) already contains all multicomponent influences that come from solution nonideality.

In many practical applications it is convenient to express the diffusive flux in terms of the local concentrations \mathbf{c} or mole fractions \mathbf{X} instead of the potential gradient $\partial\mu/\partial r$. In this case, the flux of component i can be obtained by chain derivation of equation (5.65) with

$$J_i = -c_i B_i \frac{\partial \mu_i}{\partial r} = -c_i B_i \sum_{j=1}^n \frac{\partial \mu_i}{\partial X_j} \frac{\partial X_j}{\partial r} = -X_i B_i \sum_{j=1}^n \frac{\partial \mu_i}{\partial X_j} \frac{\partial c_j}{\partial r} \quad (5.66)$$

Equation (5.66) relates the one-dimensional flux J_i of component i to the composition gradients $\partial X_j/\partial r$ of all components j . Comparison of coefficients with Fick's first law defines the intrinsic diffusion coefficients in multicomponent diffusion with

$$D_{ij} = X_i B_i \frac{\partial \mu_i}{\partial X_j} \quad (5.67)$$

The *intrinsic* diffusion coefficients D_{ij} now form a matrix of dimension $[n \times n]$, that is,

$$D_{ij} = \begin{bmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & & \\ \vdots & & \ddots & \\ D_{n1} & & & D_{nn} \end{bmatrix} \quad (5.68)$$

and the intrinsic fluxes can be written in compact form as

$$J_i = D_{ij} \nabla c_j, \quad i, j = 1 \dots n \quad (5.69)$$

or the flux of component i in one dimension and explicitly writing the summation over j

$$J_i = \sum_{j=1}^n D_{ij} \frac{\partial c_j}{\partial r} \quad (5.70)$$

In equation (5.70), the summation is performed over the composition gradients $\partial c_j/\partial r$ of all n elements, thus summing up the influence of each atomic species j on the diffusion behavior of element i .

However, not all of the fluxes J_i are independent, which is easily demonstrated. Since the sum over all n concentrations c_i equals the molar density $1/V_m$ (V_m is the molar volume),

$$\sum_{i=1}^n c_i = \frac{1}{V_m} \quad (5.71)$$

and for the derivative $\partial/\partial r$, we obtain

$$\sum_{i=1}^n \frac{\partial c_i}{\partial r} = 0 \quad (5.72)$$

Consequently, the concentration gradient of one component can always be expressed in terms of the gradients of the other ones. When choosing component n as the dependent element, we have

$$\frac{\partial c_n}{\partial r} = - \sum_{i=1}^{n-1} \frac{\partial c_i}{\partial r} \quad (5.73)$$

The constraint (5.73) is substituted into Fick's first law in the multicomponent framework (5.69). After collecting terms, the equation system of intrinsic fluxes is overdetermined and the flux of one component can be expressed in terms of the $n - 1$ independent fluxes. With the dependent component n , we thus obtain the *reduced* intrinsic diffusivity matrix D'_{ij} , which is a matrix of dimension $[(n - 1) \times (n - 1)]$

$$D'_{ij} = \begin{bmatrix} D_{11} - D_{1n} & D_{12} - D_{1n} & \cdots & D_{1(n-1)} - D_{1n} \\ D_{21} - D_{2n} & D_{22} - D_{2n} & & \\ \vdots & & \ddots & \\ D_{(n-1)1} - D_{(n-1)n} & & & D_{(n-1)(n-1)} - D_{(n-1)n} \end{bmatrix} \quad (5.74)$$

and the $n - 1$ independent fluxes given with

$$J_i = D'_{ij} \nabla c_j, \quad i, j = 1 \dots n - 1 \quad (5.75)$$

or in one dimension

$$J_i = \sum_{j=1}^{n-1} D'_{ij} \frac{\partial c_j}{\partial r} \quad (5.76)$$

The intrinsic fluxes J_i can be transformed from the lattice frame of reference into the laboratory frame of reference with a Galilean transformation, that is, the addition of a convective flux term. This term has already been introduced in Section 5.3.4 in the discussion of the Kirkendall effect. The velocity v by which the intrinsic coordinate system moves relative to the laboratory frame of reference is given by the net flux of vacancies, which can be calculated from

$$J_{V_a} = - \sum_{j=1}^n J_j \quad (5.77)$$

and

$$v = J_{V_a} V_m \quad (5.78)$$

The flux \tilde{J}_i in the laboratory frame of reference can then be expressed by the sum of the intrinsic flux J_i^{intr} and the convective flux J_i^{conv} with

$$\tilde{J}_i = J_i^{\text{intr}} + J_i^{\text{conv}} \quad (5.79)$$

With the intrinsic flux given by equation (5.69) and the convective flux given as

$$J_i^{\text{conv}} = c_i v = -X_i \sum_{j=1}^n J_j \quad (5.80)$$

the flux of component i reads

$$\tilde{J}_i = - \sum_{j=1}^n D_{ij} \frac{\partial c_j}{\partial r} + X_i \sum_{k=1}^n \sum_{j=1}^n D_{kj} \frac{\partial c_j}{\partial r} \quad (5.81)$$

After expansion of the first term, introduction of the Kronecker delta δ_{ij} , with the usual meaning of $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$, and rearrangement of terms, equation (5.81) becomes

$$\tilde{J}_i = - \sum_{k=1}^n \sum_{j=1}^n \delta_{ik} D_{kj} \frac{\partial c_j}{\partial r} + X_i \sum_{k=1}^n \sum_{j=1}^n D_{kj} \frac{\partial c_j}{\partial r} \quad (5.82)$$

and

$$\tilde{J}_i = - \sum_{k=1}^n \sum_{j=1}^n (\delta_{ik} - X_i) D_{kj} \frac{\partial c_j}{\partial r} \quad (5.83)$$

Again, the constraint (5.72) applies, and one of the concentration gradients can be expressed in terms of the $n - 1$ independent concentration gradients. Using the reference element n and insertion of equation (5.73) into (5.81) leads to

$$\tilde{J}_i = - \sum_{j=1}^{n-1} (D_{ij} - D_{in}) \frac{\partial c_j}{\partial r} + X_i \sum_{k=1}^{n-1} \sum_{j=1}^{n-1} (D_{kj} - D_{kn}) \frac{\partial c_j}{\partial r} \quad (5.84)$$

After some algebra, the *independent* fluxes in the laboratory frame of reference can be written as

$$\tilde{J}_i = - \sum_{j=1}^{n-1} \tilde{D}'_{ij} \frac{\partial c_j}{\partial r} \quad (5.85)$$

with the diffusion coefficient matrix

$$\tilde{D}'_{ij} = \sum_{m=1}^{n-1} \sum_{s=1}^{n-1} (\delta_{js} - X_j)(\delta_{mi} - X_i) D_{ms} \quad (5.86)$$

Finally, the relations between the atomic mobility and the different diffusion coefficients are summarized in Table 5-3.

TABLE 5-3 Relation between Mobility and Different Diffusion Coefficients

<i>Kinetic quantity</i>	<i>Symbol</i>	<i>Frame of reference</i>	<i>Comment</i>
Mobility	B	Lattice	Mobility of atoms
Tracer diffusion - <i>self diffusion</i> - <i>impurity diff.</i>	D^*	Lattice	Chemically homogeneous material, usually radioactive isotopes (tracer atoms) $D^* = RT \cdot B$
Intrinsic diffusion	D	Lattice	Diffusion in chemical potential gradient. $D = D^* \cdot \phi$
Interdiffusion <i>chemical diff.</i>	\tilde{D}	Laboratory	Transformation of intrinsic diffusion coefficients into laboratory frame of reference.

5.4 Numerical Solution of the Diffusion Equation

The problems that are collected and discussed in this section represent typical examples of application of the relations and concepts described earlier in this chapter. Since, in this section, we aim at solving the relations numerically, the technique of finite differences is outlined first. This technique is a classic, simple, and yet powerful method to transform differential equations into difference expressions, which can easily be solved numerically. Other methods are presented in other chapters of this book.

Reconsider Fick's first law [equation (5.15)] in one dimension:

$$J = -D \frac{\partial c}{\partial x} \quad (5.87)$$

This equation relates the flux of atoms j at an arbitrary position x_i to the diffusion coefficient D (which is assumed to be independent of composition here) and the local concentration gradient $\partial c/\partial x$. On applying the finite differences concept, the *continuous*, "real" concentration profile (as displayed exemplarily in Figure 5-10) is *discretized* by equidistant⁷ segments of size Δx . With the values of the concentrations in the discrete points, the gradient $\partial c/\partial x$ can be replaced by the approximate expression

$$\left(\frac{\partial c}{\partial x} \right)_{x=x_i} = \frac{c_{i+1} - c_{i-1}}{2\Delta x} \quad (5.88)$$

The difference expression (5.88) thus represents the linearized concentration gradient at point x_i determined from the concentrations in the neighboring points x_{i-1} and x_{i+1} .

⁷ The distances are assumed to be equidistant for simplicity here. The finite difference methodology can be applied to variable discretization intervals as well; however, the expressions become more involved then.

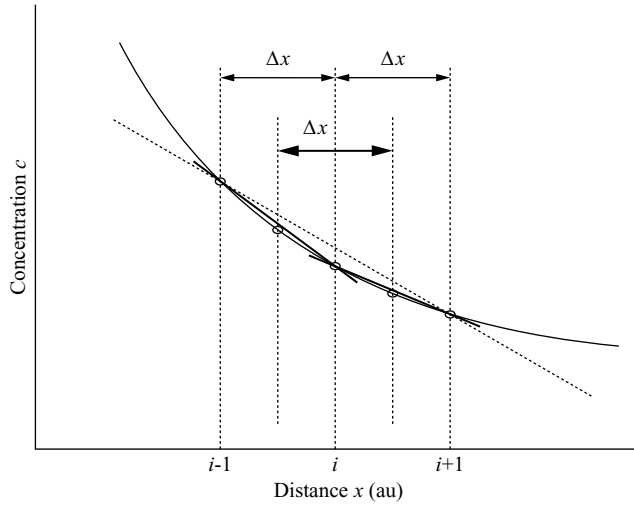


FIGURE 5-10 Discretization of the diffusion equation with finite differences.

The individual fluxes in the segments neighboring the center point x_i can be evaluated analogously and we obtain

$$\begin{aligned} J_{x_{i-1} \rightarrow x_i} &= -D \frac{c_i - c_{i-1}}{\Delta x} \\ J_{x_i \rightarrow x_{i+1}} &= -D \frac{c_{i+1} - c_i}{\Delta x} \end{aligned} \quad (5.89)$$

These fluxes can now be utilized to transform the transient diffusion equation, that is, Fick's second law, into finite differences notation also. Reconsider equation (5.22) in one dimension

$$\frac{\partial c}{\partial t} = \frac{\partial J}{\partial x} = \frac{J_+ - J_-}{\Delta x} \quad (5.90)$$

with the fluxes J_- on the left side and J_+ on the right side of the center point x_i . The fluxes are taken at the center of the respective segments at positions $x_i - \Delta x/2$ and $x_i + \Delta x/2$ and a spatial distance of Δx . Since these quantities are given by equations (5.89), the difference expression is straightforwardly obtained after insertion of the fluxes in the neighboring segments with

$$\frac{\partial c_i}{\partial t} = D \frac{\partial^2 c}{\partial x^2} = D \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} \quad (5.91)$$

Finally, equation (5.91) is further discretized in time. With the individual time steps Δt , we obtain

$$\frac{c_{i,t+\Delta t} - c_{i,t}}{\Delta t} = D \frac{c_{i+1,t} - 2c_{i,t} + c_{i-1,t}}{\Delta x^2} \quad (5.92)$$

Equation (5.92) represents a convenient relation for numerical integration of the evolution of the concentration profiles in diffusion simulations. In the most simple case, Euler forward integration (or *explicit* integration) can be used and the concentrations at time $t + \Delta t$ are given in terms of the concentrations at time t with

$$c_{i,t+\Delta t} = c_{i,t} + D \frac{c_{i+1,t} - 2c_{i,t} + c_{i-1,t}}{\Delta x^2} \Delta t \quad (5.93)$$

It can be shown that the maximum time step Δt_{\max} for a numerically stable integration must obey the condition

$$\Delta t_{\max} \leq \frac{(\Delta x)^2}{2D}. \quad (5.94)$$

PROBLEM 5-3: Single-Component Diffusion into a Semi-infinite Sample. Explicit Forward Integration of Fick's Second Law in One Dimension

Consider a semi-infinite sample with an initial mole fraction of an arbitrary diffusant given with $X_{\infty} = 0.1$. The surface of the sample is brought into contact with an infinite reservoir of atoms of the diffusant such that the surface at $x = 0$ has a constant mole fraction $X_0 = 0.5$. Assume that the diffusion constant is $D = 10^{-8} \text{ m}^2/\text{s}$. Discretize the space domain with 100 equidistant segments of $\Delta x = 10^{-6} \text{ m}$. Evaluate the concentration profiles (in terms of mole fractions) at times $t = 1, 10, 100, 1000 \text{ s}$. Note that the maximum time integration step size is given by equation (5.94). Compare with the analytical solution to this problem given by equation (5.36) and the typical transient diffusion profiles shown in Figure 5-8.

Although Euler forward integration using equation (5.93) allows for a straightforward calculation of the values of the concentration profile at time $t + \Delta t$ directly from the values of the previous time step t , this method of integration suffers from the severe drawback that the maximum time step criterion [equation (5.94)] can severely limit the forward progression of integration in time. To overcome this limitation, equation (5.92) can be rewritten in terms of the unknown concentrations at time $t + \Delta t$, which yields

$$\frac{c_{i,t+\Delta t} - c_{i,t}}{\Delta t} = D \frac{c_{i+1,t+\Delta t} - 2c_{i,t+\Delta t} + c_{i-1,t+\Delta t}}{\Delta x^2} \quad (5.95)$$

After separation of variables with respect to time and introduction of the factor α with

$$\alpha = \frac{2D}{\Delta x^2} \Delta t \quad (5.96)$$

the implicit form of the finite differences representation of Fick's second law reads

$$-\alpha c_{i+1,t+\Delta t} + (1 + 2\alpha) c_{i,t+\Delta t} - \alpha c_{i-1,t+\Delta t} = c_{i,t} \quad (5.97)$$

Equation (5.97) defines a linear system of equations in the unknown concentrations c_n at time $t + \Delta t$, which can be written as

$$\begin{bmatrix} 1 + 2\alpha & -2\alpha & 0 & 0 & \cdots \\ -\alpha & 1 + 2\alpha & -\alpha & 0 & \cdots \\ 0 & -\alpha & 1 + 2\alpha & -\alpha & \cdots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \vdots \end{pmatrix}_{t+\Delta t} = \begin{pmatrix} c_{0,t} \\ c_{1,t} \\ c_{2,t} \\ c_{3,t} \\ \vdots \end{pmatrix}_t \quad (5.98)$$

Note that the first row in the coefficient matrix differs from the other rows. The factor of 2 in the second coefficient of this row is due to the assumption that symmetric boundary conditions

hold at the left boundary of the sample with the index 0. This kind of boundary condition is realized with introduction of a hypothetical point at coordinate x_{i-1} . This point is assumed to have the same properties as the point at position x_{i+1} . In this case, the coefficient at the *real* point x_{i+1} must be taken into account twice, which leads to the factor 2.

It can be shown that a time integration scheme based on the implicit representation of the transient diffusion equation is unconditionally stable for arbitrary time steps Δt .⁸ For diffusion simulations, implicit time integration is usually the method of choice due to its superior behavior in the later stages of simulation compared to explicit or semi-implicit methods.

PROBLEM 5-4: Single-Component Diffusion into a Semi-infinite Sample. Implicit Integration of Fick's Second Law in One Dimension

Consider the same situation that is outlined in Problem 5-3. Use implicit integration and try different time steps. Test the superior stability of the implicit time integration scheme over the explicit version.

So far, we have considered diffusion of a single component. In the case of multiple components, the relations presented in Sections 5.3.4 and 5.3.5 must be applied. In general, the simulation becomes more involved due to the mutual interaction of the individual diffusion species. In simple cases, the necessary input parameters for simulation can be given in explicit form.

In many practical cases, the interactions between the multiple components cannot easily be given, and numerical software to solve the diffusion problem must be employed. Typical software for this purpose are the software “profiler” by M. E. Glicksman (see ref. [Gli00]), which employs a database for multicomponent diffusion coefficients, or the commercial software package DICTRA [AHJA90]. The latter is coupled to a thermodynamic engine, which calculates multicomponent thermodynamic potentials based on the CALPHAD methodology (see also Section 2.2.8). Other software that can be used is the package MatCalc (<http://matcalc.tugraz.at>).

Bibliography

-
- [AHJA90] J. O. Andersson, L. Höglund, B. Jönsson, and J. Agren. *Computer Simulation of Multicomponent Diffusional Transformations in Steel*, pp. 153–163. Pergamon Press, New York, 1990.
- [Cra75] J. Crank. *Mathematics of Diffusion*. Oxford University Press, Oxford, 1975.
- [Dar48] L. S. Darken. Diffusion, mobility and their interrelation through free energy in binary metallic systems. 175:184–201, 1948.
- [Ein05] A. Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der Physik*, 17:549, 1905.
- [Fic55] A. E. Fick. *Über Diffusion*, Vol. 94, p. 59. 1855.
- [Gli00] M. E. Glicksman. *Diffusion in Solids*. Wiley, New York, 2000.
- [Kir42] E. O. Kirkendall. Diffusion of zinc in alpha brass. *Trans. AIME*, 147:104–110, 1942.
- [KTU39] E. Kirkendall, L. Thomassen, and C. Upthegrove. Rates of diffusion of copper and zinc in alpha brass. *Trans. AIME*, 133:186–203, 1939.
- [KY87] J. S. Kirkaldy and D. J. Young. *Diffusion in the Condensed State*. The Institute of Metals, London, 1987.
- [Phi86] J. Philibert. *Diffusion et Transport de Matière dans les Solides*. Editions de Physique, 1986.
- [She86] P. G. Shewmon. *Diffusion in Solids*. The Minerals, Metals & Materials Society, Warrendale, PA, 1986.
- [SK47] A. D. Smigelskas and E. O. Kirkendall. Zinc diffusion in alpha brass. *Trans. AIME*, 171:130–142, 1947.

⁸ However, one must be aware that, with increasing time integration intervals, the accuracy of the solution suffers.

This page intentionally left blank

6 Modeling Precipitation as a Sharp-Interface Phase Transformation

—Ernst Kozeschnik

During phase transformations, a new phase grows at the expense of an existing phase. The new phase and the existing phase, or *parent phase*, are distinguished by either a different state of matter (e.g., liquid water droplets in a saturated H₂O vapor, solid crystals in a liquid melt), different crystal structure (e.g., solid bcc iron (ferrite) in solid fcc iron (austenite) in steels) and/or chemical composition (e.g., coherent L1₂-ordered fcc Ni₃Al precipitates in an fcc Ni–Al alloy).

In this chapter, it is assumed that the growing and the shrinking phases are clearly and unambiguously separated by a phase boundary, the *interface*. The thickness of this interface is considered to be infinitely small (*sharp-interface* limit). This assumption is in contrast to what is assumed, for instance, in the phase-field model, which is discussed in Chapter 7, and where the transition from one phase into the other is assumed to be continuous. The term *sharp* implies that the interfacial region is sufficiently small to allow for a theoretical treatment of phase transformations with a *stepwise* change of material properties from one phase into the other.

During phase transformation, the interface between the two phases migrates from the growing phase into the parent phase. Simultaneously, physical processes such as transport of energy (heat conduction) or transport of matter (atomic diffusion) occur in the bulk of the two phases as well as across the interface. This fact makes sharp-interface phase transformations a classical *moving boundary* problem, a class of problems which is generally known as *Stefan* problem and which is named after the Austrian–Slovene physicist and mathematician Jožef Stefan (1835–1893) in the analysis of simultaneous liquid–solid interface movement and heat transfer during solidification of water (ice formation).

Precipitation is a special case of phase transformation, where the spatial extension of the new phase is usually small (few nanometers to few micrometers) compared to the parent phase (micrometers to millimeters). Usually, the parent phase remains widely unaltered during the precipitation process. In this context, the parent phase is also often denoted as the *matrix phase*. Typical precipitates can be intermetallic phases or oxides, carbides, and nitrides. The term “phase transformation” is often, yet not exclusively, used in the context of phase changes that occur on the scale of the polycrystalline microstructures, that is, typically micrometers to millimeters. Otherwise, in the context of this book, phase transformation and precipitation denote the same class of problem.

Figure 6-1 shows the typical appearance of a microstructure after bulk phase transformation (top) and precipitation (bottom). The phase transformation example shows ferritic grains after the austenite (fcc iron) to ferrite (bcc iron) transformation in an ultralow carbon steel. The bottom image shows γ' -precipitates in the nickel-base superalloy UDIMET 720Li. Whereas the austenite to ferrite transformation in the steel example creates a new polycrystalline *bulk* microstructure, the γ' precipitation process creates a dispersion of small second phase particles inside the unchanged polycrystalline fcc microstructure.

The approaches to modeling phase transformations and precipitation kinetics are manifold and have quite different levels of abstraction. This chapter starts with a section that introduces a generalized treatment of phase transformation processes in the framework of the extended

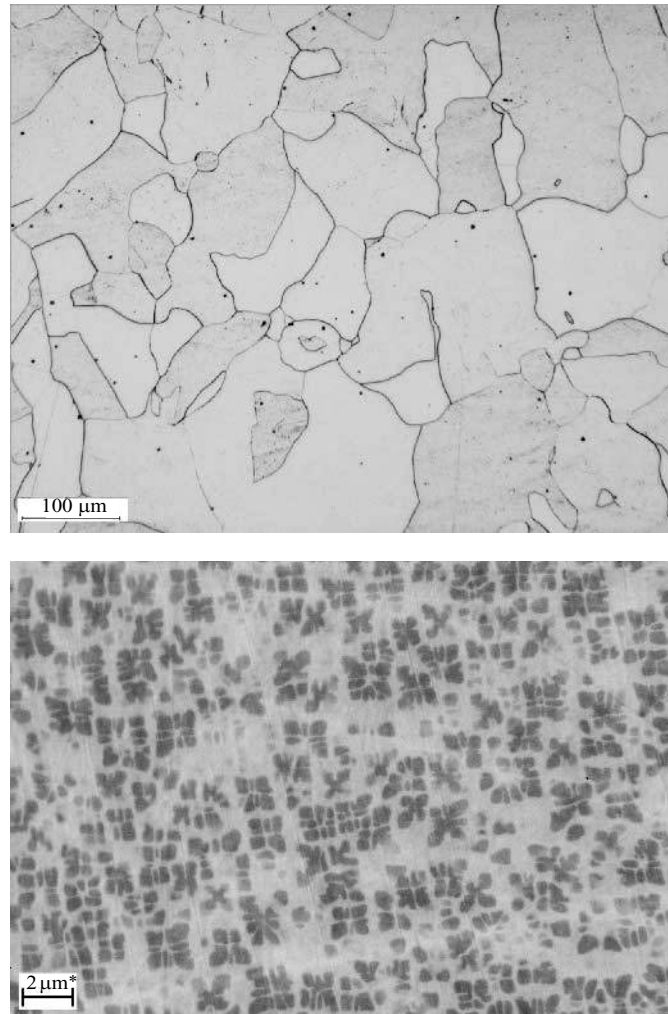


FIGURE 6-1 Facetes of phase transformations: polycrystalline microstructure (top, ultralow carbon Fe–C after austenite to ferrite transformation, optical microscopy) and precipitates in a multiphase alloy (bottom, γ' -precipitates in nickel-base superalloy UDIMET 720, scanning electron microscopy).

volume concept. The second section of this chapter deals with nucleation theory, followed by modeling of precipitation processes and a description of the kinetics of precipitation with the traditional three stages of precipitate life: nucleation, growth, and coarsening. A numerical scheme is introduced then, which allows treating precipitate populations and size distributions. Finally, the growth kinetics as predicted from the different models in this chapter are compared.

For additional treatments of phase transformations, the reader is referred to textbooks such as refs. [PE04, Hil98, Aar99, Kos01].

6.1 Statistical Theory of Phase Transformation

When treating the process of phase transformations in a most generalized way, a variable ξ can be introduced, which describes the progress of transformation in terms of a number between 0 (no transformation) and 1 (transformation completed). If the rate \dot{G} , at which the new grains/precipitates grow, and the rate \dot{N} , at which new grains/precipitates are created, are known, the overall progress of transformation can be calculated based on the *Kolmogorov–Johnson–Mehl–Avrami* (KJMA) theory.

The concept, which is presented in the following section, can generally be applied to any process that is governed by nucleation and/or growth and where the growing objects come into contact after some time. This contact avoids further growth and occurs, for example, if two growing grains touch during recrystallization or the diffusion fields of two growing precipitates start to overlap. In both cases, the transformation comes to a stop because there is no more bulk volume or diffusant available for further transformation. This process is generally known as *impingement*. If two or more grains or new phases come into “physical” contact, we speak of *hard* impingement. If contact occurs only indirectly, for example, the diffusion fields of growing precipitates overlap, the process is known as *soft* impingement.

6.1.1 The Extended Volume Approach—KJMA Kinetics

The extended volume approach describes the overall progress of a *general* phase transformation with impingement as a function of time. The term “general” emphasizes that this theory can be applied to problems of bulk phase transformation and precipitation in the same way as to phenomena such as recrystallization or solidification. The concepts behind the theory have been developed by A. N. Kolmogorov [Kol37], W. Johnson and R. F. Mehl [JM39], and M. Avrami [Avr39, Avr40, Avr41]. Thus, the theory is often referred to as *Kolmogorov–Johnson–Mehl–Avrami* or KJMA theory.

Consider a new phase β growing into a parent phase α . Let us assume that there are a few nuclei of β nucleated between time $t = 0$ and $t = \Delta t$, with a volume that is sufficiently small compared to the total volume of the parent phase [Figure 6-2(a)]. Within a subsequent time interval Δt , the existing nuclei grow at a rate \dot{G} and new nuclei are formed at a rate \dot{N} . The locations of the new nuclei are chosen at random and no account is taken whether the positions of the new nuclei belong to the already transformed volume or to the original, untransformed volume. The corresponding situation is depicted in Figure 6-2(b).

Figure 6-2(c) illustrates that, in continuously repeating the growth and nucleation steps, two critical situations can occur:

1. Since the size of the growing precipitates constantly increases, volumes of different precipitates can *overlap*.

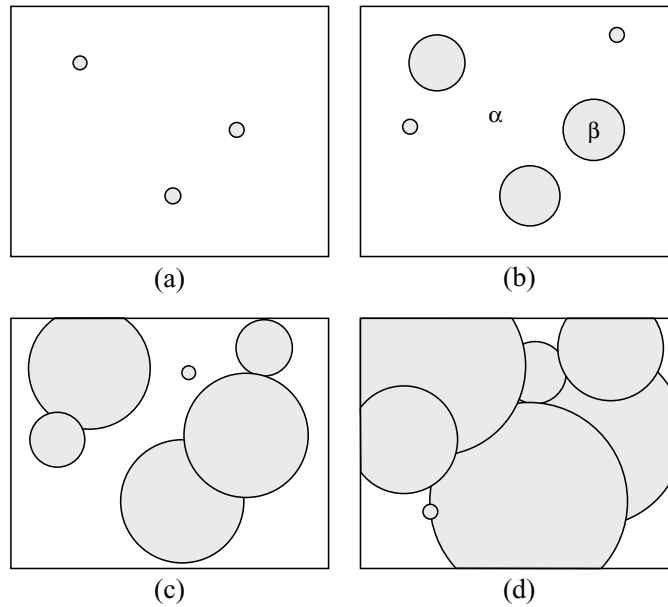


FIGURE 6-2 Schematic microstructure during phase transformations using the extended volume concept.

2. Since new nuclei are created at random locations, nuclei can be seeded within the already transformed volume.

To overcome these difficulties, the following assumptions are made: (i) first, let the new phase nucleate and grow, irrelevant if this process involves already transformed regions. Denote all volume that is created by nucleation and growth as the *extended volume*. (ii) Relate the extended volume to the actual transformed volume by the probability that a further nucleation and growth step occurs in untransformed volume.

When calculating the volume that is created over time by a single nucleation event, suppose that the transformation occurs at constant temperature and that the growth rate \dot{G} and nucleation rate \dot{N} are constant. The volume v^e of a single sphere, which has nucleated at a time τ , is then given with

$$v^e = \frac{4\pi}{3} \dot{G}^3 (t - \tau)^3 \quad (6.1)$$

The superscript “e” indicates that the volume of the sphere belongs to the extended volume, that is, the transformed volume neglecting impingement effects. If we consider the total number dN_τ of nucleation events between time τ and $\tau + d\tau$, we obtain

$$dN_\tau = \dot{N}_\tau V_{\text{tot}} d\tau \quad (6.2)$$

where V_{tot} is the total volume of the transforming sample. The total increase in extended volume within the time interval $d\tau$ is given as

$$dV^e = v^e \cdot dN_\tau = \frac{4\pi}{3} \dot{G}^3 (t - \tau)^3 \dot{N}_\tau V_{\text{tot}} d\tau \quad (6.3)$$

The total extended volume, which is created by all nucleation events, can be obtained by integration of equation (6.3), which, for constant \dot{N} and \dot{G} , yields

$$V^e = \frac{4\pi}{3} \dot{G}^3 \dot{N} V_{\text{tot}} \int_0^t (t - \tau)^3 d\tau = \frac{\pi}{3} \dot{G}^3 \dot{N} V_{\text{tot}} t^4 \quad (6.4)$$

and, with the extended volume fraction ξ^e , we have

$$\xi^e = \frac{V^e}{V_{\text{tot}}} = \frac{\pi}{3} \dot{G}^3 \dot{N} t^4 \quad (6.5)$$

In the next step, the extended volume fraction ξ^e is related to the actual transformed volume fraction ξ . This is accomplished by multiplying the change in extended volume dV^e by the probability to find untransformed regions. This has the effect of excluding all regions that have already transformed and which cannot contribute to the increase in real transformed volume. Thus, the increment in real volume dV is given with

$$dV = \left(1 - \frac{V}{V_{\text{tot}}}\right) dV^e \quad (6.6)$$

with the probability represented by the term in brackets. Equation (6.6) can easily be integrated after separation of variables, which leads to

$$\frac{V}{V_{\text{tot}}} = 1 - \exp\left(-\frac{V^e}{V_{\text{tot}}}\right) \quad (6.7)$$

and

$$\xi = 1 - \exp(-\xi^e) \quad (6.8)$$

With the extended volume fraction ξ^e given by equation (6.5), we finally obtain

$$\xi = 1 - \exp\left(-\frac{\pi}{3} \dot{G}^3 \dot{N} t^4\right) \quad (6.9)$$

Equation (6.9) describes the overall progress of a phase transformation taking into account nucleation, growth, and impingement. Although this particular form of equation is only valid for the assumptions made in the derivation, that is, constant \dot{N} and \dot{G} , it can be shown that equations derived for alternative nucleation and growth conditions have similar form. It is observed that the KJMA equation can be generally written as

$$\xi = 1 - \exp(-kt^n) \quad (6.10)$$

where k is often denoted as the *Avrami coefficient* and n as the *Avrami exponent*. In most cases, the value of n is between 1 and 4. The Avrami coefficient contains the nucleation and growth rates and is, therefore, very sensitive to variations in temperature. The Avrami exponent is related to the mechanism of phase transformation, for instance, to whether the nucleation rate remains constant or even increases throughout the progress of transformation or whether the nucleation rate goes to zero shortly after growth has started.

Figure 6-3 shows the general shape of the KJMA equation (6.10) for different values of k and n . The top diagram shows variations of the Avrami coefficient k , while the Avrami exponent n is held constant, whereas n is varied in the bottom diagram with k held constant.

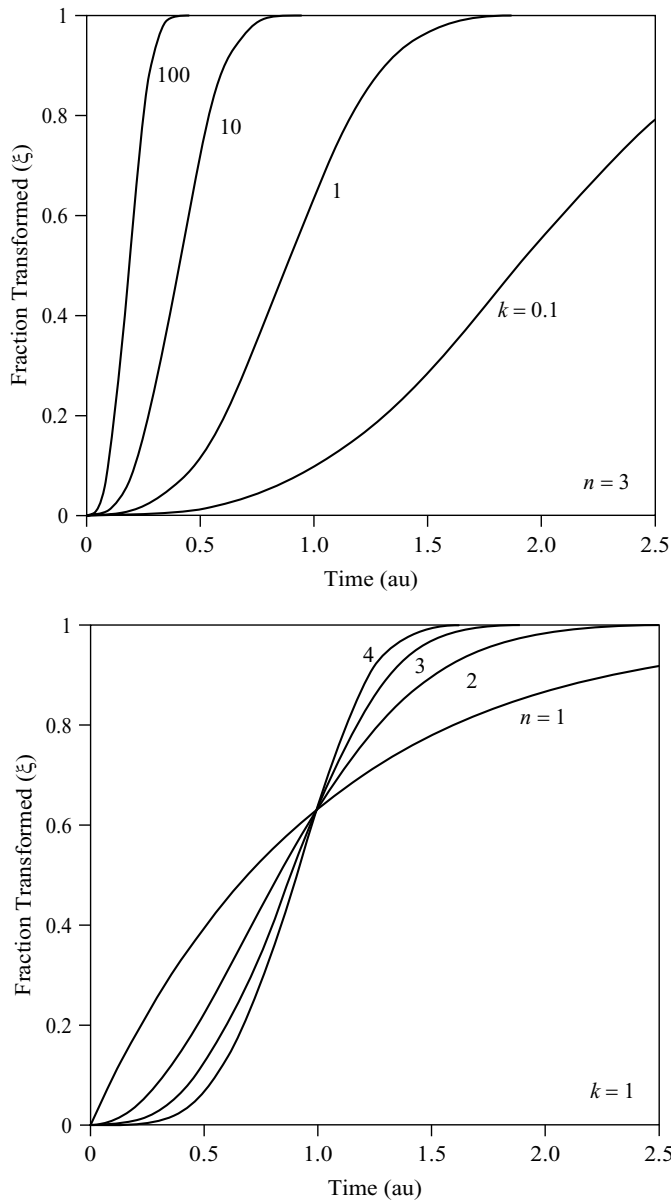


FIGURE 6-3 Transformation progress according to the KJMA equation (6.10), (top) variation of the Avrami coefficient k , (bottom) variation of the Avrami exponent n .

The fact that n is closely related to the mechanism of transformation, that is, the balance between nucleation and growth, suggests that the shape of the curve allows us to determine the governing mechanism of phase transformation. Unfortunately, these values are not unique in the sense that the same exponent n can be characteristic for multiple types of transformation. Nevertheless, some values for n are summarized in Table 6-1, which have originally been compiled in reference [Chr02].

TABLE 6-1 Interpretation of the Avrami Exponent in Equation (6.10)^a

<i>Value of n</i>	<i>Interpretation</i>
	<i>Polymorphic changes, discontinuous precipitation, eutectoid reactions, interface-controlled growth, etc.</i>
>4	Increasing nucleation rate
4	Constant nucleation rate
3–4	Decreasing nucleation rate
3	Zero nucleation rate
2	Grain edge nucleation
1	Grain boundary nucleation
	<i>Diffusion controlled growth</i>
>2.5	Small dimensions, increasing nucleation rate
2.5	Small dimensions, constant nucleation rate
1.5–2.5	Small dimensions, decreasing nucleation rate
1.5	Small dimensions, zero nucleation rate
1–1.5	Growth of particles of large volume
1	Needles and plates, large separation
1	Thickening of long cylinders
0.5	Thickening of large plates
2/3	Precipitation on dislocations (Cottrell atmosphere)

^a From J. W. Christian, *The Theory of Transformations in Metals and Alloys*, ref. [Chr02].

6.2 Solid-State Nucleation

Going back to the early 1900s, scientists have tried to understand and quantify the initial stages of the formation of new phases. In this section, the basic ideas of nucleation theory will be reviewed. Since *classical nucleation theory* (CNT) has been developed for homophase nucleation (homogeneous nucleation, due to compositional fluctuations and without change of the crystal structure) or heterophase nucleation in unary and binary systems only, in addition, an extension to CNT is presented for nucleation in multicomponent alloys.

6.2.1 Introduction

Nucleation is the process by which a new phase forms within an existing phase. Typical examples of phenomena that occur in the process of nucleation are

- Homogeneous condensation of droplets within supersaturated vapors. In fact, the early and classical experimental studies on nucleation have focused on condensation of water droplets in vapor chambers, where supersaturation is easily and controllably realized through quick expansion of pressurized and saturated vapor. Or simply think of the early morning fogs in the vicinity of lakes and rivers.
- Heterogeneous condensation of droplets on undercooled surfaces. If you wear eyeglasses and you enter a warm room in winter, you immediately recognize the nucleation of droplets on the cold glass. This type of nucleation differs from the previous one through the kind of nucleation site: In the first case, the droplets have formed randomly within the vapor, whereas in this case, special nucleation sites have been activated and taken advantage of.

- Solidification of liquids on cooling. When a liquid starts to solidify, new crystals are predominantly created on the walls of the liquid container. Growth of the crystals occurs in columnar or dendritic mode into the liquid core. Only in the later stages of solidification are crystals also nucleated within the liquid phase.
- Precipitation of second phases in multicomponent solid or liquid matter. Due to equilibrium partitioning of individual components in the different phases of the alloy, the new phases grow with a chemical composition which is in general different from that of the parent phase. Therefore, solid-state nucleation frequently involves the (long-range) transport of atoms by diffusion.

In many cases, nucleation is not an easy process and does not happen without cost or effort. Usually, formation of a new phase needs some activation such that the classical nucleation barrier is overcome. This process is commonly treated in terms of probabilities, which makes nucleation a stochastic process. Once the nucleus has reached overcritical size, it can grow in a deterministic manner. Concepts to describe the nucleation and growth process are discussed in the following sections.

6.2.2 Macroscopic Treatment of Nucleation—Classical Nucleation Theory

Consider a homogeneous binary alloy with components A and B. Let us assume that there is some driving force for formation of clusters of pure B atoms. Let the initial configuration be a homogeneous solution of B in A and let the Gibbs free energy of unit volume of atoms in this configuration be G_{AB} .

If, by compositional fluctuations in the matrix,¹ a cluster of pure B atoms forms in the alloy at some arbitrary location, the Gibbs free energy of unit volume of this cluster can be defined as G_{BB} . If it is further assumed that the reservoir of atoms in the initial configuration is sufficiently large such that the mean chemical composition of the atoms surrounding the cluster is unchanged by the nucleation process, the difference in bulk energy for unit volume of atoms transformed from the initial alloy into the cluster can be written as

$$\Delta G_{AB \rightarrow BB} = G_{BB} - G_{AB} = \Delta G_{\text{bulk}}^0 \quad (6.11)$$

When looking at a single cluster and assuming that the cluster has spherical shape with a radius ρ , the *bulk* energy difference ΔG_{bulk} between the initial configuration and the configuration after the cluster has formed is

$$\Delta G_{\text{bulk}} = \frac{4}{3}\pi\rho^3 \cdot \Delta G_{\text{bulk}}^0 \quad (6.12)$$

Since the cluster now has a distinct shape and chemical composition other than the composition of the matrix, an interfacial area can be defined. Generally, the atomic binding in the interface between the atoms in the cluster and the atoms in the matrix is weaker than the binding between the like atoms on both sides of the interface and, consequently, this new interfacial region must be taken into account in the analysis of the cluster formation energy. A detailed quantification of the binding energies across interfaces is given later in Section 6.2.5 on interfacial energies.

The contribution ΔG_{surf} of the interfacial region to the total free energy of cluster formation can be expressed in terms of the specific interfacial energy γ and the geometrical surface area with

$$\Delta G_{\text{surf}} = 4\pi\rho^2 \cdot \gamma \quad (6.13)$$

¹ Although a solution can be *homogeneous* on a macroscopic level, that is, the solution contains no gradients in concentration, there are always *local, microscopical* variations in chemical composition observable, which are caused by the random walk of vacancies.

The total energy change due to formation of this cluster is then

$$\Delta G = \Delta G_{\text{bulk}} + \Delta G_{\text{surf}} = \frac{4}{3}\pi\rho^3\Delta G_{\text{bulk}}^0 + 4\pi\rho^2\gamma \quad (6.14)$$

Equation (6.14) manifests the basic concept behind CNT, which treats the total free energy change during cluster formation as the sum of a term ΔG_{bulk} , which is proportional to the volume of the new cluster, and a term ΔG_{surf} , which is proportional to the surface area created during nucleation. Figure 6-4 displays these two terms as function of the cluster size together with the total free energy change.

According to equation (6.14) and Figure 6-4, the early stages of cluster formation are characterized by an increase in total free energy with increasing cluster size. This means that,

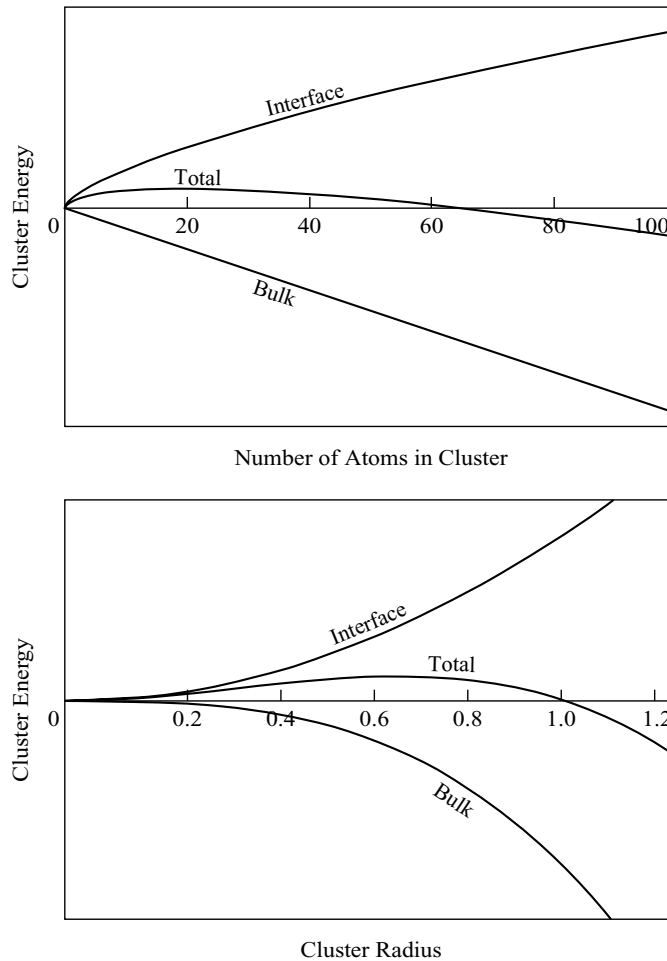


FIGURE 6-4 Bulk and surface free energy of a spherical nucleus as a function of the number of atoms N in the cluster (top) and the cluster radius R (bottom).

not until a critical cluster size is reached, energy must be invested for each individual atom that attaches to the cluster. Even though the like B atoms are attracted by each other, small clusters of B atoms are energetically unfavorable and they are always more likely to dissolve than to grow.

However, nucleation is a stochastic process and, with some probability, the random compositional fluctuations create clusters, which are large enough to grow. Once, a cluster has reached critical size, addition of extra atoms is a process where energy is gained rather than spent and cluster growth becomes more likely than cluster dissolution. At this point, the stochastic regime of precipitate nucleation switches over to the deterministic regime of precipitate growth. In a first approach, the particular size where this transition occurs is given by the maximum of the nucleation free energy ΔG . The radius of a sphere containing exactly the number of atoms at this point is called *critical nucleation radius* ρ^* and the value of the free energy at the maximum is the *nucleation barrier* or the *critical nucleation energy* G^* . The position of the maximum nucleation free energy can be found by setting the derivative of equation (6.14) with respect to ρ to zero, that is,

$$\frac{\partial \Delta G}{\partial \rho} = 4\pi\rho^2 \Delta G_{\text{bulk}}^0 + 8\pi\rho\gamma = 0 \quad (6.15)$$

The critical nucleation radius ρ^* follows with

$$\rho^* = -\frac{2\gamma}{\Delta G_{\text{bulk}}^0} = \frac{2\gamma}{D} \quad (6.16)$$

In equation (6.16), an effective driving force D has been introduced with $D = -\Delta G_{\text{bulk}}^0$. Note that D is closely related to the chemical driving force D_{ch} , which has been introduced in equation (2.79) of Section 2.2.6. In the absence of mechanical stresses or other driving forces, the relation $D = \Omega D_{\text{ch}}$ holds, where Ω is the molar volume.

Back substitution of equation (6.16) into (6.14) yields the critical nucleation energy G^* with

$$G^* = \frac{16\pi}{3} \frac{\gamma^3}{D^2} \quad (6.17)$$

For initiation of a single nucleation event, the critical nucleation energy G^* must be overcome and the probability P_{nucl} that this process occurs can be expressed as

$$P_{\text{nucl}} = \exp\left(-\frac{G^*}{kT}\right) \quad (6.18)$$

where k is the Boltzmann constant and T is the absolute temperature. From the probability of an individual nucleation event, the frequency of nucleation events in unit volume and unit time can be deduced. The respective quantity J is denoted as the *nucleation rate* and it quantifies the number of nuclei that are created in unit volume per unit time. The unity of J is [*events*/(m^3s)]. Under steady state conditions, the nucleation rate J_{SS} is proportional to the probability P_{nucl} of a single nucleation event multiplied by the total number of possible nucleation sites N_0 . To obtain the exact expression for the steady state nucleation rate, further thermodynamic and kinetic aspects have to be taken into consideration. These will not be elucidated here and the interested reader is referred to, for example, the textbook by Khashchiev [Kha00] or the review by Russell [Rus80].

The rigorous treatment of nucleation in the framework of CNT delivers that the steady state nucleation rate can be interpreted as the flux of clusters in cluster size space, which grow from critical to overcritical size and, in condensed systems, J_{SS} can be written as

$$J_{SS} = N_0 Z \beta^* \exp\left(-\frac{G^*}{kT}\right) \quad (6.19)$$

In equation (6.19), the additional quantities Z and β^* have been introduced. The Zeldovich factor Z is related to the fact that the critical size of a nucleus is not exactly given by the maximum of the cluster formation energy. An additional energy contribution from thermal activation kT has to be taken into account because the thermal vibrations destabilize the nucleus as compared to the unactivated state. Z is often of the order of 1/40 to 1/100 and thus decreases the effective nucleation rate.

The atomic attachment rate β^* takes into account the long-range diffusive transport of atoms, which is necessary for nucleus formation if the chemical composition of matrix and precipitate differs. Quantitative expressions for these quantities are given in Section 6.2.4.

6.2.3 Transient Nucleation

In the previous section, we have found that clusters are created by random compositional fluctuations and that the steady state nucleation rate J_{SS} is determined by the flux of clusters in cluster size space, which grow from critical to overcritical size. In the derivation of the steady state nucleation rate, it has been assumed—without explicitly mentioning it—that the distribution of clusters is in a stationary state, namely, the size distribution of clusters is time invariant (see ref. [Rus80]). This is rarely the case, however, in practical heat treatment situations at least in the initial stages.

Consider a *homogeneous* solution of B atoms in an A-rich matrix, which has been homogenized at a temperature above the solution limit of the B clusters.² After quenching from homogenization temperature into a supersaturated state, the sharp cluster size distribution, which initially consists of mainly monomers and dimers, becomes wider, because larger clusters are stabilized by the increasing influence of favorable B–B bonding over thermally induced mixing. Only after a characteristic time, which is determined by factors such as atomic mobility, driving force, and interfacial energy, a stable distribution of clusters can be established, which is denoted as the *equilibrium cluster distribution*. The characteristic period until the equilibrium cluster distribution is reached is denoted as the *incubation time* τ .

It is interesting to note that a time-invariant cluster distribution can only exist when no driving force for cluster formation is present, that is, clusters are thermodynamically unstable. If a positive driving force for precipitation exists, overcritical clusters will immediately grow in a deterministic manner and will thus escape the stochastic distribution of clusters produced by random compositional fluctuations. Figure 6-5 schematically shows cluster distributions for situations, where the largest clusters have undercritical or supercritical size, respectively. In the first case, the cluster distributions are stationary and time invariant (equilibrium cluster distribution). The shape of the distributions only depends on driving force and temperature. In the second case, precipitates are continuously nucleated and the shape of the size distribution depends on time.

² A homogeneous solution can be achieved by annealing for a sufficiently long time at a sufficiently high temperature above the solution temperature of the precipitate phase. In this case, the vast majority of B atoms is present in the form of monomers and dimers and only a negligible number of larger clusters exists. The supersaturated state is established by rapid quenching from homogenization temperature to reaction temperature.

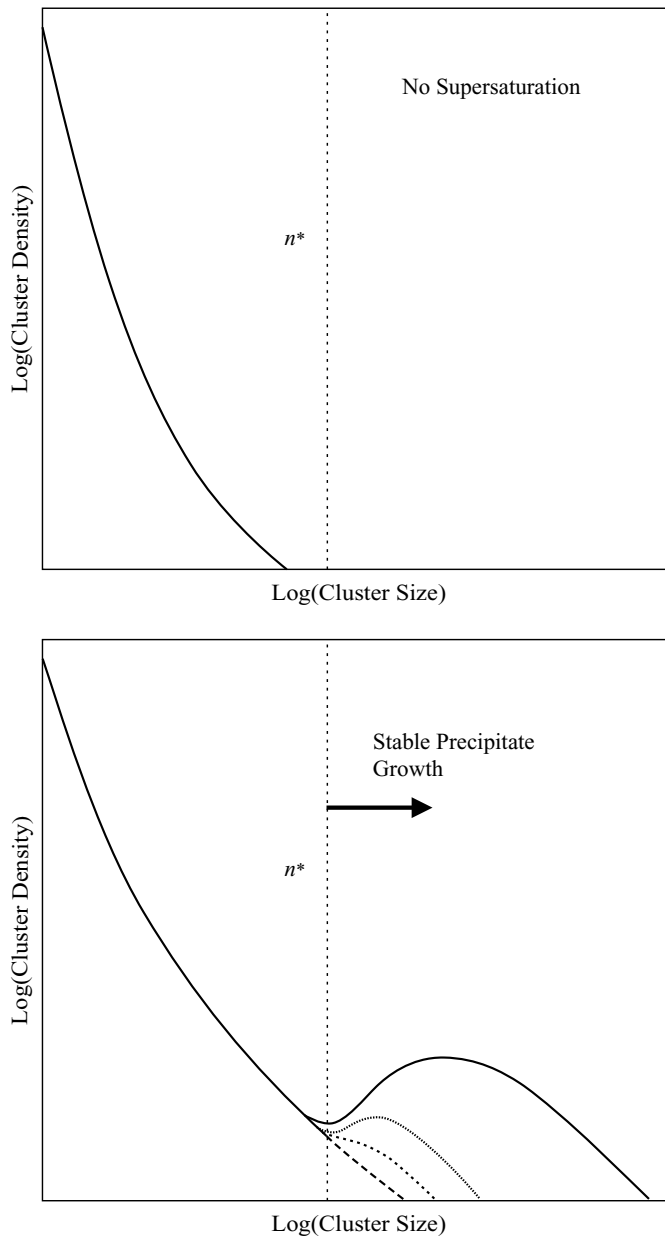


FIGURE 6-5 Typical equilibrium cluster distributions without driving force for precipitation, (top) and with driving force, (bottom). The top distributions are time invariant. Under steady state conditions, the bottom distributions will continuously create stable precipitates.

When taking the incubation time τ for nucleation into account in the expression for the nucleation rate, a most pragmatic approach is to multiply the steady state nucleation rate J_{SS} by a smooth function, which is zero at time $t = 0$, and which approaches unity at times $t > \tau$. In CNT, the traditional expression is

$$J = J_{SS} \cdot \exp\left(-\frac{\tau}{t}\right) \quad (6.20)$$

and we finally obtain

$$J = N_0 Z \beta^* \exp\left(-\frac{G^*}{kT}\right) \exp\left(-\frac{\tau}{t}\right) \quad (6.21)$$

The transient nucleation rate J describes the rate at which nuclei are created per unit volume and unit time taking into account the incubation time τ . It should be noted, finally, that the exponential function in equation (6.20) has received some criticism due to physical inconsistencies. Nonetheless, this approach is widely used because the error in practical calculation, which is introduced by this weakness, is small compared to the uncertainties of other input quantities, such as the interfacial energy.

6.2.4 Multicomponent Nucleation

The transient nucleation rate given in equation (6.21) can be rigorously derived for binary alloy systems. However, already in ternary systems, the applied methodology becomes involved and treatments of higher-order systems are more or less lacking. In a first approximation, equation (6.21) can nevertheless be applied to multicomponent systems, provided that extended expressions for some of the quantities that appear in this relation are used.

When investigating equation (6.21) closer, we find that some of the quantities are already applicable to multicomponent systems “as they are.” For instance, the number of potential nucleation sites N_0 is independent of the number of components and the Zeldovich factor Z as well as the critical nucleation energy G^* already contain full multicomponent thermodynamic information. The critical quantity, which contains kinetic quantities describing multicomponent diffusive fluxes, is the atomic attachment rate β^* . An approximate multicomponent expression has been derived in ref. [SFFK04] in the modeling of multicomponent multiphase precipitation kinetics based on the thermodynamic extremal principle. The corresponding expression is presented in Table 6-2 together with expressions for the other necessary quantities for evaluation of multicomponent nucleation rates.

Finally, an important note shall be placed on practical evaluation of multicomponent nucleation rates. It has not yet been emphasized that all quantities in Table 6-2 rely on the *a priori* knowledge of the chemical composition of the nucleus. The term “*a priori*” means that we have to input the nucleus composition in all formulas without really knowing what this composition should be.

Luckily, there are some concepts that help us in making reasonable guesses of what a “successful” and realistic nucleus composition might be. In a first step, it is assumed that the wide variety of possible compositions can be substituted by a single *characteristic* composition. This step is rationalized by the reasoning that, from the variety of different possible chemical compositions, precipitates which appear first and/or precipitates which appear in the highest number density will be the most successful ones in the direct growth competition of an entire precipitate population. In this sense it should be sufficient to consider this most successful representative precipitate composition only.

In the second step, a chemical composition is chosen for the representative nucleus, which most closely represents the situation under which the nucleus is formed. For instance, in a system with only substitutional elements, the composition which gives the highest chemical

TABLE 6-2 Expressions for Evaluation of Multicomponent Nucleation Kinetics Based on Equation (6.21)

Quantity		Value	Comment
Z (dim. less)	Zeldovich factor	$\left[\frac{-1}{2\pi kT} \frac{\partial^2 \Delta G}{\partial n^2} \right]^{\frac{1}{2}}$	$n \dots$ number of atoms in the nucleus
β^* s^{-1}	Atomic attachment rate	$\frac{4\pi\rho^{*2}}{a^4\Omega} \left[\sum_{i=1}^n \frac{(c_{ki} - c_{0i})^2}{c_{0i}D_{0i}} \right]^{-1}$	$\rho^* \dots$ crit. nucl. radius $a \dots$ atomic distance $\Omega \dots$ molar volume $c_i \dots$ concentrations $D_{0i} \dots$ diffusion coeff.
ΔG^* (J)	Critical nucleation energy	$\frac{16\pi}{3} \frac{\gamma_k^3}{F^2}$	$F \dots$ effective driving force $\gamma \dots$ interfacial energy
ρ^* (m)	Critical nucleation radius	$\frac{2\gamma_k}{F}$	
τ (s)	Incubation time	$\frac{1}{2\beta^* Z^2}$	

driving force³ could be a reasonable choice because maximum driving force D leads to (i) approximately maximum thermodynamic stability and often also to (ii) maximum nucleation rates (the nucleation barrier G^* is minimum in the exponential term of the nucleation rate J). However, the second statement is not always true. If substantial long-range diffusional transport of atoms toward the nucleus is necessary to grow the nucleus, this process can be very costly in terms of time. Much higher nucleation rates and, thus, a higher nucleus density could be achieved with compositions, which are somewhere in between the maximum driving force composition and a composition with minimum necessary solute transport. The parameter determining the amount of necessary diffusive transport is the atomic attachment rate β^* (see Table 6-2). This quantity is a maximum, if the chemical composition of the nucleus is most closely the composition of the matrix, namely, minimum transport of atoms is necessary to form a nucleus.

A typical example for this latter situation is given in the precipitation of carbides and nitrides in steels, where the precipitates are composed of slow diffusing substitutional elements and fast diffusing interstitial elements, such as carbon and nitrogen. Under specific conditions, the growth of carbides with a composition close to the matrix composition and only transport of fast diffusing carbon and nitrogen is more favorable than forming precipitates with high thermodynamic stability and high content of carbide forming elements, but slow nucleation kinetics due to slow diffusion of these elements. Figure 6-6 shows the theoretical nucleation rates for cementite precipitates $(FeCr)_3C$ in the ternary system Fe-3wt%Cr-C as evaluated with the CNT relations given in Table 6-2 for varying Cr content of the cementite nuclei. The different curves are related

³ For practical evaluation of the composition with highest driving force, see Section 2.2.6

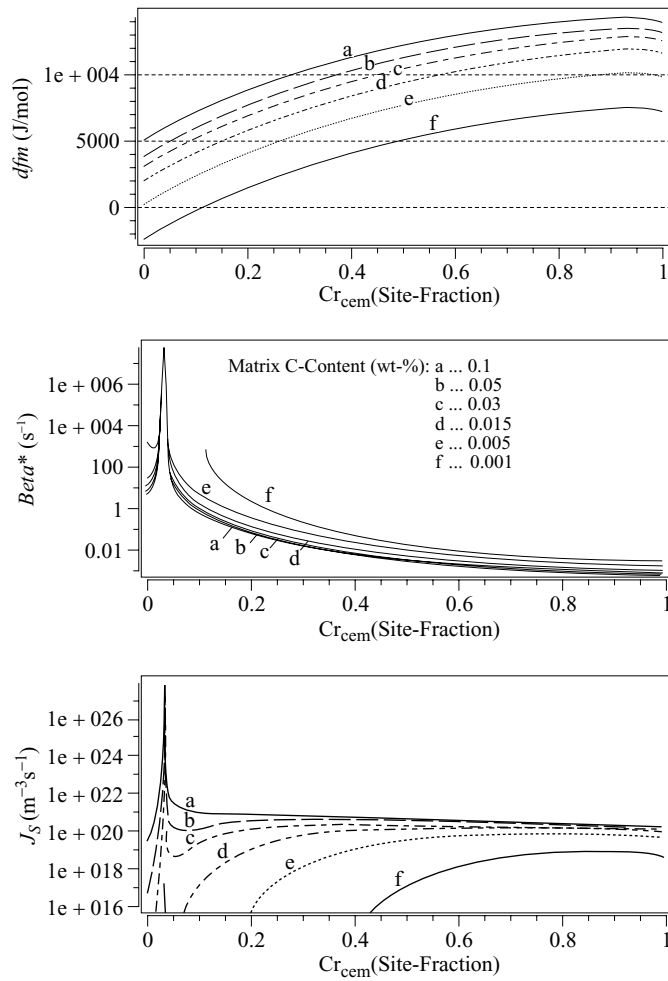


FIGURE 6-6 Chemical driving force dfm , atomic attachment rate β^* , and steady state nucleation rate J_S as a function of the Cr content of a cementite precipitate in the $Fe-Cr-C$ system (from ref. [KSF05b]).

to different carbon content of the supersaturated matrix, which is equivalent to different driving forces for precipitation.

The analysis demonstrates that, under situation of high supersaturation, the highest nucleation rates are achieved for the so-called *paraequilibrium* composition, which is the particular chemical composition where matrix and precipitate have the same amount of substitutional elements and only the amount of interstitial elements differs.⁴ At the paraequilibrium composition, β^* is a maximum, because only fast diffusing carbon atoms are needed to grow the precipitate,

⁴ The term “*paraequilibrium*” composition is related to a specific type of constrained equilibrium, in which equilibration of chemical potentials is only achieved for interstitial elements, whereas substitutional elements are not allowed to partition between the phases. The term “*ortho-equilibrium*” composition denotes the full, unconstrained thermodynamic equilibrium for substitutional and interstitial elements.

and the Fe to Cr ratio is identical in precipitate and matrix. With decreasing supersaturation, the chemical driving force decreases and, at some point, the nucleation rate for paracomposition cementite goes to zero, whereas the driving force for higher-chromium nuclei is still sufficient to support significant nucleation.

Finally a note is dropped on practical evaluation of the optimum nucleus composition. From a physical point of view, the particular composition, which yields the highest nucleation rate, is often a most reasonable choice. However, computation of this composition is not always easy because equation (6.19) must be scanned in the entire composition space. In practical simulation, orthoequilibrium and paraequilibrium composition are popular choices due to the fact that they are often available from thermodynamic equilibrium calculation without additional computational cost.

6.2.5 Treatment of Interfacial Energies

In the previous sections, we have introduced the interfacial energy γ as a convenient physical quantity, which describes the energy of the narrow region between precipitate and matrix. In reality, however, this quantity is most delicate and, only in rare cases, reliable values of γ are known. One of the reasons for this is the fact that γ cannot be measured directly by experimental means. Interfacial energies can only be obtained by indirect methods, that is by comparison of suitable experiments with the corresponding theoretical treatment, which includes the interfacial energy as a parameter. A most popular method in this respect is to compare experiments on phase transformation and precipitation kinetics to corresponding theoretical models and determine the interfacial energy by adjusting γ such that simulation and experiment are in accordance. Another problematic aspect in using interfacial energies in kinetic simulations is the fact that γ is (strongly) dependent on a number of parameters, such as crystallographic misorientation, elastic misfit strains, degree of coherency, and solute segregation. All this makes interfacial energies a never-ending story of scientific interest, research, and also misconception.

In this section, a popular approach is presented, which relates the interfacial energy of a coherent phase boundary to interatomic bonding and finally to the thermodynamic quantity enthalpy. This approach allows for an estimation of *effective* interfacial energies, which can be used in computer simulations as first estimates.⁵

The Nearest-Neighbor Broken-Bond Model

The theoretical foundation for the first approach to calculating interfacial energies from consideration of atomic bonding was laid by W. L. Bragg and E. J. Williams [BW34] in 1934. In this work, the concept of nearest-neighbor bonds was introduced and applied to estimate the total energy of a crystal based on the sum of binding energies of neighboring atoms. This idea was shortly after (1938) applied by R. Becker in his *nearest-neighbour broken-bond* model [Bec32]. Some years later, in 1955, D. Turnbull [Tur55] made the connection between the interfacial energy and the enthalpy of solution. This concept is briefly reviewed now.

Consider two blocks of material. Block 1 is composed of pure A atoms, whereas block 2 consists of pure B atoms. Divide each block into two sections, and interchange the half blocks (see Figure 6-7). The energy of the newly formed interfaces in blocks 3 and 4 can be calculated as the sum of the energies of the new bonds in blocks 3 and 4, minus the energy of the broken bonds in the original blocks 1 and 2.

⁵ Although derivation of the following expressions is demonstrated rigorously only for coherent interfaces, in many metallic systems, the values for γ obtained by this methodology can also be applied to incoherent interfaces.

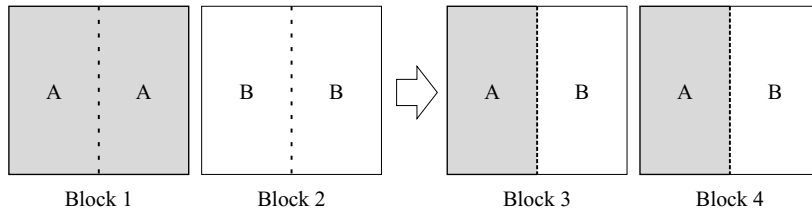


FIGURE 6-7 Calculation of interfacial energies.

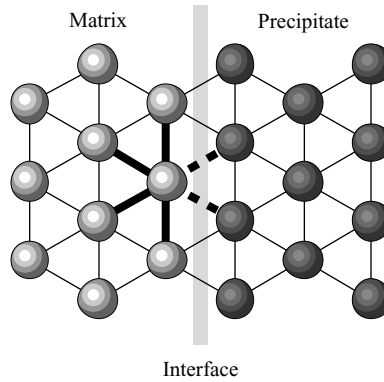


FIGURE 6-8 Two-dimensional coherent interface with nearest-neighbor broken bonds.

According to this thought experiment, the specific interfacial energy γ is evaluated as the difference in bond energies between the two separate blocks and the energy of the interchanged blocks per unit interfacial area. Thus, we can write

$$\gamma = E_{AB}^{\text{new}} - E_{AA}^{\text{broken}} - E_{BB}^{\text{broken}}, \quad (6.22)$$

where the energy E refers to unit area of interface. The energies in equation (6.22) are easily obtained by counting the broken bonds in the blocks. Figure 6-8 schematically illustrates broken bonds across a two-dimensional interface.

In a general crystal structure, let z_S be the number of bonds across the interface counted per atom and let n_S be the number of surface atoms per unit area within the surface plane. Accordingly, we have

$$\begin{aligned} E_{AA}^{\text{broken}} &= \frac{n_S z_S}{2} \cdot \epsilon_{AA} \\ E_{BB}^{\text{broken}} &= \frac{n_S z_S}{2} \cdot \epsilon_{BB} \\ E_{AB}^{\text{new}} &= n_S z_S \cdot \epsilon_{AB} \end{aligned} \quad (6.23)$$

The factor 1/2 for the like A–A and B–B bonds avoids counting bonds twice. This factor is missing in the expression for the AB bonds because we have two identical A–B interfaces (see Figure 6-7). For the interfacial energy we finally have

$$\gamma = n_S z_S \left[\epsilon_{AB} - \frac{1}{2} (\epsilon_{AA} + \epsilon_{BB}) \right] \quad (6.24)$$

According to equation (6.24), the coherent interfacial energy is determined by the the crystallographic neighborhood and the difference in bond energies between like and unlike atoms. Since ϵ_{AA} , ϵ_{BB} , and ϵ_{AB} are not always readily available, these quantities can conveniently be substituted by a quantity, which is nowadays easily accessible in the framework of computational thermodynamics: the enthalpy of solution.

Interatomic Bonding and Enthalpy

Recalling the formalism of the *regular solution* model of Section 2.2.3, the total bond energy of one mole of pure A atoms and one mole of pure B atoms is

$$\begin{aligned} E_{AA} &= \frac{Nz}{2} \cdot \epsilon_{AA} \\ E_{BB} &= \frac{Nz}{2} \cdot \epsilon_{BB} \end{aligned} \quad (6.25)$$

On mixing one mole A and one mole B atom, we have

$$E_{AB} = Nz \cdot \epsilon_{AB} \quad (6.26)$$

Consequently, the difference between the bond energies of the pure substances and the mixture is given with

$$\Delta H = zN \left[\epsilon_{AB} - \frac{1}{2} (\epsilon_{AA} + \epsilon_{BB}) \right] \quad (6.27)$$

with the coordination number z , which defines the number of nearest neighbors of an atom in the solution and the Avogadro number N . Comparison of equations (6.24) and (6.27) links the specific interfacial energy γ to the enthalpy of mixing ΔH by

$$\gamma = \frac{n_S z_S}{N z_L} \cdot \Delta H \quad (6.28)$$

ΔH is defined as the change in enthalpy on dissolving one mole of substance B in A. In the framework of computational thermodynamics, the solution enthalpy can easily be calculated for a given configuration. Let f be the phase fraction of a precipitate phase. The change in enthalpy on transferring one mole of atoms from the matrix phase into the precipitate is $\Delta H = \partial H / \partial f$ and the interfacial energy can finally be written as

$$\gamma = \frac{n_S z_S}{N z_L} \cdot \frac{\partial H}{\partial f} \quad (6.29)$$

It should again be emphasized that the interfacial energy as defined in equation (6.29) is valid only in a first approximation for real matrix–precipitate interfacial energies, because real interfaces are rarely coherent and elastic misfit and crystallographic misorientation are not explicitly considered in the nearest-neighbor broken-bond model. However, more recent computations have shown that the structural factor $\kappa = z_S / z_L$ in fcc, bcc, and hcp structures is frequently in the order of $0.28 < \kappa < 0.33$. With this approximation, equation (6.29) can deliver reasonable first estimates of interfacial energies even for general matrix–precipitate interfaces.

6.3 Diffusion-Controlled Precipitate Growth

Precipitation is a phenomenon where atoms agglomerate to form clusters of a new phase. The driving force for the process comes from favorable atomic bonding of the cluster atoms, and it is expressed in terms of the supersaturation of the original solution, that is, an undercooling below the solution temperature of the precipitate phase. A typical heat treatment for controlled precipitation consists of (i) solution annealing above solution temperature, (ii) rapid quenching to precipitation temperature, and (iii) annealing for a given period of time. The first step is performed to homogenize the material and to evenly distribute the solute atoms. In the third step, the precipitate-forming atoms cluster and grow.

Common types of precipitates in solid-state materials are carbides, nitrides, carbonitrides, and oxides as well as special precipitate-forming intermetallic phases. If the chemical composition of the matrix and the precipitate differs, the precipitation process involves the long-range transport of atoms by diffusion. In many cases, precipitate and matrix also differ in the type of crystal structure, and precipitation needs simultaneous short-range lattice rearrangement. However, diffusion is often the slower process and, therefore, the rate-controlling factor. If the growth rate of a precipitate is governed by the diffusive transport of atoms, the reaction is denoted as being *diffusion controlled*. If the growth rate of a precipitate is determined by the rearrangement process of atoms across the interface, the reaction is said to be *interface controlled*.

When looking at typical life cycles of precipitates, or entire precipitate populations, a traditional categorization is often made into three stages of precipitation, which are

- **Nucleation:** The initial stages in precipitate life. Stochastic process driven by microscopic thermal and compositional fluctuations.
- **Growth:** Controlled attachment of atoms to the precipitate. Deterministic process driven by chemical and/or mechanical driving forces.
- **Coarsening:** Dissolution of small precipitates in favor of larger ones. Driven by curvature-induced pressure, that is, the Gibbs–Thomson effect (compare Section 2.2.7).

Figure 6-9 shows a typical sequence of snapshots during the lifetime of a precipitate population. Assume that the open circles represent precipitate-forming atoms. Figure 6-9(a) displays a state, where the hypothetical system is held at very high temperature, such that the majority of atoms occurs as monomers. Formation of larger clusters is prohibited by the strong thermal activation of the atoms. In Figure 6-9(b), the temperature of the system has been decreased and the attractive bond energies between the precipitate-forming atoms leads to formation of small but unstable clusters. Nevertheless, the system is still above solution temperature of the precipitate phase and no larger clusters occur.

Figure 6-9(c) represents a situation, where a first stable nucleus of the precipitate phase has formed. The vicinity of the stable precipitate is depleted from solute atoms, because these have diffused toward the stable cluster. In the Figure 6-9(d), the simultaneous nucleation and growth of precipitates has proceeded. Temperature has been further decreased and strong clustering of even smaller precipitates is observed. At this stage, temperature is held constant and the precipitation reaction is continued in isothermal regime.

The Figures 6-9(e) and (f) show two snapshots from the last stage of precipitation, that is, coarsening or Ostwald ripening. Many of the smallest precipitates or clusters have already dissolved and the corresponding atoms have attached to the larger precipitates, which continue to grow at the expense of the smaller ones. The driving force for this process comes from the Gibbs–Thomson effect (compare Section 2.2.7), which says that the concentration of solute atoms around curved interfaces is higher the smaller the radius of the curved surface is.

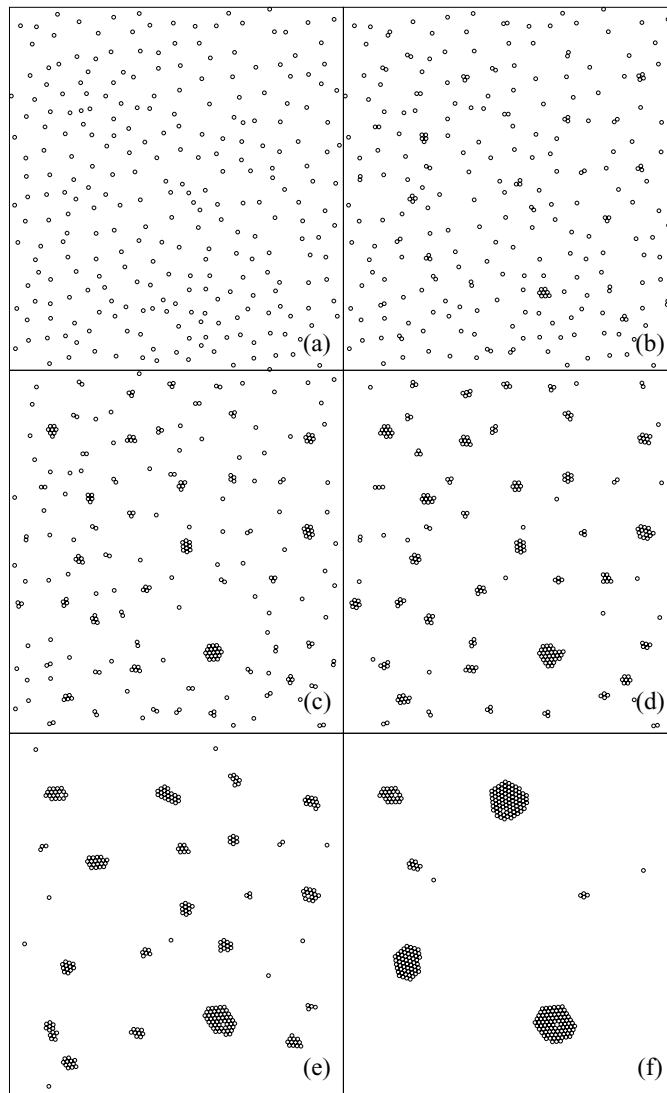


FIGURE 6-9 *Different stages in a precipitate life. See text for discussion.*

If the solute concentration around a small precipitate is higher than the concentration around a neighboring larger precipitate, a diffusive flux exists, which continuously transports atoms from the smaller to the larger precipitate. Consequently, the smaller precipitates dissolve and the larger continue to grow.

Although the individual stages in precipitate life are not always clearly separable, they have clear characteristics that lead to different theoretical treatments of the individual stages. Whereas nucleation is considered to be a *stochastic* process, that is, nuclei are born as a consequence of random thermal and compositional fluctuations, the growth and dissolution processes are described in a *deterministic* way. Precipitate nucleation has already been dealt with in Section 6.2.2; modeling of the growth of precipitates is the topic of this section.

The third process, coarsening, can be treated explicitly in an analytical manner or implicitly by numerical integration of the evolution laws for precipitate growth and dissolution. In the analytical approach to coarsening, the classical Lifschitz–Slyozow–Wagner (LSW) theory is most prominent. With this theory, evolution laws for the mean radius of a precipitate population and the precipitate number density is given. Since analytical treatments are discussed in detail in many textbooks, we will not repeat the corresponding theory here. And, when treating the growth and dissolution processes in a numerical framework on the computer, coarsening is implicitly taken into account. A corresponding numerical approach to the evolution of precipitate populations has been suggested by Kampmann and Wagner in 1984 [KW84] and the model is commonly denoted as the *numerical Kampmann–Wagner model*, which is presented later in this section.

6.3.1 Problem Definition

Consider an alloy system A–B with B atoms dissolved in an A-rich matrix. Assume that B-rich precipitates can form in the solution and that sufficient driving force is available for this process. When the precipitate grows, B atoms must be supplied by the surrounding matrix by a diffusional process. Figure 6-10 schematically shows the concentration profile around a growing precipitate and also indicates the movement of the phase interface due to the diffusional flux of B atoms toward and into the growing precipitate’s interface.

When describing the physical processes, which occur during phase transformation as depicted in Figure 6-10, for convenience, the problem can be separated into the three problems of

- Migration of the phase boundary
- Diffusion of atoms in the precipitate
- Diffusion of atoms in the matrix

The last two processes can be modeled straightforwardly based on Fick’s second law for multiple components (compare Section 5.3.5). For further treatment, with the simplification

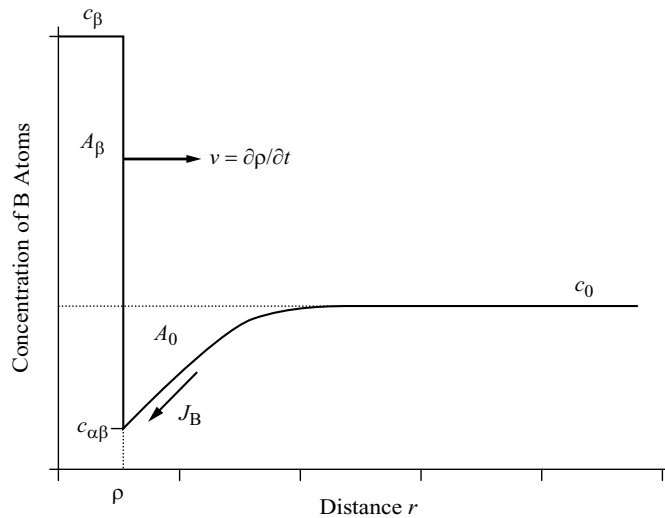


FIGURE 6-10 Concentration profile of B atoms around the moving interface of a growing stoichiometric precipitate.

of a composition independent diffusion coefficient D and for a one-dimensional axisymmetric geometry with the spatial coordinate r , the *diffusional flux* of atoms on both sides of the interface can be expressed as

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial r^2} \quad (6.30)$$

Next, we have to consider *mass conservation*. In precipitation modeling, this requirement is equivalent to the statement that, during precipitate growth, the number of atoms entering the precipitate in unit time and across unit interfacial area, that is, the flux of atoms into the precipitate–matrix interface, is compensated by the difference of interfacial composition on the two sides of the interface multiplied with interface velocity. With the variables depicted in Figure 6-10, mass conservation can be expressed as

$$(c_\beta - c_{\alpha\beta}) \frac{\partial \rho}{\partial t} = -J(\rho) \quad (6.31)$$

The flux $J(\rho)$ at the precipitate–matrix interface can further be expressed in terms of Fick’s first law and we have

$$(c_\beta - c_{\alpha\beta}) \frac{\partial \rho}{\partial t} = D \frac{\partial c}{\partial r}(\rho) \quad (6.32)$$

It is important to note that equation (6.32) holds for each geometry, whether it is axisymmetric linear, cylindrical, or spherical. Moreover, this equation uniquely defines the interface velocity $\partial \rho / \partial t$ by simple consideration of mass conservation. The only unknowns, which are, in principle, left undetermined, are the concentrations c_β and $c_{\alpha\beta}$ on the two sides of the interface, that is, the coupling conditions for the chemical potentials, and the concentration gradients $\partial c / \partial r$. To resolve this issue, two concepts have been applied successfully to this problem: (i) the local equilibrium hypothesis and (ii) the thermodynamic extremal principle. The first concept has been applied to phase transformation modeling for many decades; the latter approach has only more recently been “rediscovered” and applied to precipitate growth and interface kinetics. Both concepts are presented and discussed in the following sections.

In the traditional local equilibrium approach, advantage is taken of the observation that diffusion in an interface is often several orders of magnitude faster than in the bulk. Accordingly, one can assume that diffusion in the interface is always fast enough such that thermodynamic equilibrium is established immediately after two phases are brought into contact. This concept is known as the *local equilibrium hypothesis*. In local equilibrium, the chemical potentials at the two sides of the interface are identical.

For solution of the moving boundary problem based on the local equilibrium hypotheses, the unknown compositions at the two sides of the interface are to be determined. In a binary system A–B and with the phases α and β , with β representing the precipitate phase, the unknown compositions c_β and $c_{\alpha\beta}$ can most easily be read directly from the phase diagram. With Figure 6-11 and considering that the alloy has been solution treated in a state denoted by “A” and then quenched into the $\alpha + \beta$ two-phase region at point “B,” precipitates of β can nucleate in the supersaturated α matrix. In thermodynamic equilibrium, two phases will coexist with compositions that are determined by the horizontal tie-line through point “B” and with fractions determined by the lever rule.

With the concentrations given by the equilibrium tie-line, the kinetic problem is not yet completely determined. Equation (6.32) still contains the unknown concentration gradient $\partial c / \partial r(\rho)$ at the precipitate–matrix interface, which causes the diffusional fluxes into and out of the interface. In the following sections, three approaches will be presented that solve the problem with

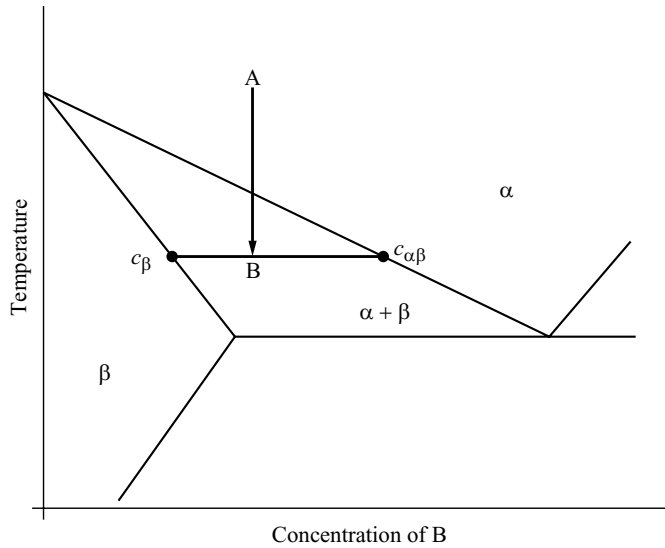


FIGURE 6-11 Binary phase diagram for a hypothetical alloy A–B. The alloy is solution treated in the α one-phase region at point A and then quenched into the $\alpha + \beta$ two-phase region at point B, where precipitates of the β phase can form. The local equilibrium concentrations at the phase interface are given by the horizontal tie-line between c_β and $c_{\alpha\beta}$.

more or less abstraction and, therefore, more or less mathematical complexity. All of these are, however, applicable to binary systems and/or diffusion of one component only. If the problem allows that the diffusional growth kinetics can be reduced to a single *representative* species, application to even complex systems is nevertheless possible (and has been reported many times in literature).

In a more general approach, when multicomponent diffusion effects should also be taken into account, the concentration profiles can be evaluated by the numerical solution of Fick's second law based on methods such as finite differences (compare Section 5.4) or finite elements (compare Chapter 9). This part of the problem can be considered as a standard problem based on the theoretical models described in Section 5.3.5 and current computational capabilities.⁶ However, an important difficulty arises, if more than two components participate in the multicomponent moving boundary problem: In two-component systems, the interfacial concentrations are uniquely determined by the horizontal tie-line in the phase diagram (see Figure 6-11). In higher-component systems, the tie-line is no longer necessarily horizontal, and the exact interfacial compositions must be evaluated as a coupled effect of multicomponent diffusion, interface movement, and mass conservation. For instance, in a system with n elements, the interface velocity is given by the system of equations

$$(c_{i,\beta} - c_{i,\alpha\beta}) \frac{\partial \rho}{\partial t} = \sum_{j=1}^n D_{ij} \frac{\partial c_j}{\partial r}(\rho), \quad i = 1, \dots, n \quad (6.33)$$

⁶ For a summary of the theory of precipitate growth and numerical techniques for solution of the moving boundary problem, the review by Van der Ven and Delacy [dVD96] is recommended.

The value of the interface velocity $\partial\rho/\partial t$ in equations (6.33) must, of course, be identical for all elements i and a valid set of interfacial concentrations must be sought under this constraint. A corresponding model is implemented, for instance, in the software package DICTRA [JAA90]. This approach can be considered as a rigorous solution of the moving boundary problem in multicomponent systems, and it is applied to a wide range of problems with good success. We must, nevertheless, be aware that limitations can apply in higher-order multicomponent systems due to numerical difficulties in evaluation of the set of equations (6.33).

Alternatively, the coupling conditions for the chemical potentials in the moving boundary problem can be derived from Onsager's principle of maximum entropy production, which is also known as the *thermodynamic extremal principle*. This principle is founded on the hypothesis that any non-equilibrated thermodynamic system evolves along the particular kinetic path where entropy production is a maximum. Application of this principle to precipitation kinetics in the framework of a mean-field description of the multicomponent multiphase multiparticle precipitation problem is presented in Section 6.4.2.

Subsequently, three models for evaluation of the growth kinetics of individual spherical precipitates are briefly reviewed.

6.3.2 Zener's Approach for Planar Interfaces

In 1949, Clarence Zener proposed a simple and elegant approximate solution [Zen49] to the diffusion-driven movement of planar, sharp interfaces. The basic assumptions that he makes are that (i) the interfacial compositions are determined by local thermodynamic equilibrium, (ii) planar symmetry is used, and (iii) the concentration profile in the surrounding matrix can be approximated by a linear profile. Figure 6-12 schematically displays this last assumption.

In Zener's treatment, a distance Z is defined such that the mass increase in B atoms in the precipitate is balanced by the decrease of mass in the matrix, that is, the two areas A_β and A_0 in the concentration profile in Figure 6-12 are identical. With the spatial coordinates ξ and x , the mass conservation law reads

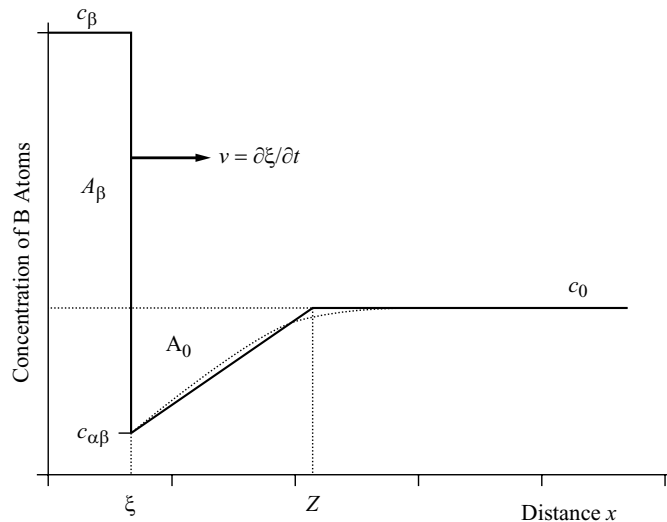


FIGURE 6-12 Schematic drawing of real and linearized concentration profile of B atoms around the moving interface of a growing precipitate.

$$(c_\beta - c_{\alpha\beta}) \frac{\partial \xi}{\partial t} = D \frac{\partial c}{\partial x}(\xi) \quad (6.34)$$

Accordingly, the velocity of the planar interface $\partial \xi / \partial t$ can be directly evaluated if the concentration gradient of B atoms is known. In Zener's approach, this gradient is approximated by the linearization of the profile according to the construction shown in Figure 6-12. With this assumption, we have

$$(c_\beta - c_{\alpha\beta}) \frac{\partial \xi}{\partial t} = D \frac{c_0 - c_{\alpha\beta}}{Z - \xi} \quad (6.35)$$

and with the identity of the areas A_0 and A_β ,

$$\xi(c_\beta - c_0) = \frac{1}{2}(Z - \xi)(c_0 - c_{\alpha\beta}) \quad (6.36)$$

we have

$$\frac{\partial \xi}{\partial t} = \frac{D}{2\xi} \frac{(c_0 - c_{\alpha\beta})^2}{(c_\beta - c_{\alpha\beta})(c_\beta - c_0)} \quad (6.37)$$

Equation (6.37) can be further simplified if the concentration of B atoms in the matrix is small compared to the concentration of B in the precipitate. In this case, $(c_\beta - c_{\alpha\beta}) \sim (c_\beta - c_0)$ and we obtain

$$\frac{\partial \xi}{\partial t} = \frac{D}{2\xi} \frac{(c_0 - c_{\alpha\beta})^2}{(c_\beta - c_{\alpha\beta})^2} \quad (6.38)$$

Equation 6.38 is the well-known parabolic growth equation of Zener, which provides a very simple relationship between the interface velocity and concentration in the case of diffusion-controlled growth in planar symmetry.

With the dimensionless supersaturation S

$$S = \frac{c_0 - c_{\alpha\beta}}{c_\beta - c_{\alpha\beta}} \quad (6.39)$$

we finally obtain

$$\frac{\partial \xi}{\partial t} = D \frac{S^2}{2\xi} \quad (6.40)$$

$$\xi \dot{\xi} = \frac{1}{2} D S^2 \quad (6.41)$$

$$\xi = \sqrt{\xi_0^2 + S^2 D t} \quad (6.42)$$

where ξ is the position of the interface and $\dot{\xi} = \partial \xi / \partial t$ is the interface velocity.

6.3.3 Quasi-static Approach for Spherical Precipitates

Consider a spherical precipitate, which is surrounded by an infinite diffusion field of initial composition c_0 . In the quasi-static approximation, the precipitate–matrix interface is assumed to be stationary and the precipitate itself is considered to be a point source and sink for solute atoms, which is located in the center of the concentration field.

Again, the concentration of B atoms at the interface can be determined from the local equilibrium hypothesis. In this quasi-stationary case, the solute profile is given by Fick's second law in spherical coordinates (see Section 5.3.1) with the boundary condition

$$\frac{\partial c}{\partial t} = D \frac{\partial}{\partial r} \left[r^2 \frac{\partial c}{\partial r} \right] = 0 \quad (6.43)$$

Equation (6.43) has the general solution

$$c = A + \frac{B}{r} \quad (6.44)$$

Based on this general solution, it is possible to analyze the flux of atoms at the interface. With the local equilibrium condition ($c(\rho) = c_{\alpha\beta}$), we can specify the constants in equation (6.44) with

$$c = c_0 - (c_0 - c_{\alpha\beta}) \frac{\rho}{r} \quad (6.45)$$

which provides the concentration gradient at $r = \rho$, that is, at the precipitate–matrix interface, and the flux into the interface as

$$\frac{\partial c}{\partial r} = (c_0 - c_{\alpha\beta}) \frac{\rho}{r^2} \quad (6.46)$$

and

$$J(\rho) = -D \frac{\partial c}{\partial r}(\rho) = -(c_0 - c_{\alpha\beta}) \frac{D}{\rho} \quad (6.47)$$

With the mass conservation equation, we can relate the diffusional flux and the interface velocity by

$$(c_{\beta} - c_{\alpha\beta}) \frac{\partial \rho}{\partial t} = -J(\rho) = (c_0 - c_{\alpha\beta}) \frac{D}{\rho} \quad (6.48)$$

Thus, we obtain the expression

$$\frac{\partial \rho}{\partial t} = \frac{D}{\rho} \frac{(c_0 - c_{\alpha\beta})}{(c_{\beta} - c_{\alpha\beta})} \quad (6.49)$$

Again, using the dimensionless supersaturation S , we finally arrive at

$$\frac{\partial \rho}{\partial t} = D \frac{S}{\rho} \quad (6.50)$$

$$\rho \dot{\rho} = DS \quad (6.51)$$

$$\rho = \sqrt{\rho_0^2 + 2SDt} \quad (6.52)$$

where ρ is the position of the interface, that is, the radius of the precipitate and $\dot{\rho} = \partial \rho / \partial t$ is the interface velocity, namely, the growth rate of the precipitate.

6.3.4 Moving Boundary Solution for Spherical Symmetry

In the quasi-stationary approximation, the assumption has been made that the movement of the interface is slow compared to the evolution of the concentration profile and that the interface can therefore be considered as stationary. If the supersaturation S increases, that is, the concentration of B atoms is high, substantial error can be introduced by this assumption.

When simultaneously taking into account the evolution of the diffusion field *and* the movement of the boundary, we can start with writing Fick's second law for the region outside the precipitate:

$$\frac{\partial c}{\partial t} = D \frac{\partial}{\partial r} \left[r^2 \frac{\partial c}{\partial r} \right], r > \rho \quad (6.53)$$

and the mass conservation law:

$$(c_\beta - c_{\alpha\beta}) \frac{\partial \rho}{\partial t} = D \frac{\partial c}{\partial r}(\rho) \quad (6.54)$$

To solve these two equations simultaneously, the substitution

$$\eta = \frac{r - \rho}{\sqrt{4Dt}} \quad (6.55)$$

is introduced. Thus, a coordinate transformation is performed into a moving coordinate system, with the position of the interface ρ implicitly given in the expression for the variable η . With equation (6.55), the diffusion equation (6.53) finally takes the (relatively) simple form

$$-2 \left[\eta + \frac{1}{\eta} \right] \frac{\partial c}{\partial \eta} = \frac{\partial^2 c}{\partial \eta^2} \quad (6.56)$$

Full derivation of this relation and solution is elaborated in, for example, Glicksman's book [Gli00] and it is clearly beyond the scope of this book. However, the results and comparison to the previous solutions will be given subsequently. According to Glicksman's treatment, the position of the interface can be expressed in terms of a constant K with

$$\rho = \sqrt{\rho_0^2 + 4K^2Dt} \quad (6.57)$$

where K can unfortunately only be given in implicit form as

$$2K^2 \cdot \left[1 - \sqrt{\pi} K e^{K^2} \operatorname{erfc}(K) \right] = S \quad (6.58)$$

With the additional substitution

$$\Phi = 2K^2 \quad (6.59)$$

equations (6.57) and (6.58) can be reformulated for comparison with the growth rate equations that have been derived for the previous approaches. They finally read

$$\rho = \rho_0 + \sqrt{2\Phi Dt} \quad (6.60)$$

and

$$\Phi \cdot \left[1 - \sqrt{\pi\Phi/2} \cdot e^{\Phi/2} \operatorname{erfc}(\sqrt{\Phi/2}) \right] = S \quad (6.61)$$

For small S , the parameter Φ converges toward S and the moving boundary solution approaches the quasi-stationary solution.

The precipitate growth models, which have been described in this chapter, are compared among each other in Section 6.5.

6.4 Multiparticle Precipitation Kinetics

In the previous sections, analytical models have been introduced, which solve the moving boundary problem for the growth of individual (spherical) precipitates at different levels of abstraction. In this section, first, the classical approach to numerically solving the multiparticle precipitation problem is reviewed, which makes use of classical nucleation theory and a parabolic growth law as described in the previous sections. This approach has been suggested by Kampmann and Wagner [KW84] in 1984 to describe precipitation of titanium in copper and γ' in Ni-Al. This methodology has later been utilized numerous times in application to different kinds of precipitation problems.

Subsequently, a model is presented for precipitation kinetics in multicomponent, multiphase, multiparticle systems. This latter model is particularly useful in complex systems with many components and many different phases. For solution of the problem, use is made of the thermodynamic extremal principle, which was formulated in 1931 [Ons31] by L. Onsager.

6.4.1 The Numerical Kampmann–Wagner Model

Consider a binary system with components A and B. Assume that clusters of B atoms can form in this system. Assume further that the nucleation kinetics of the B-rich clusters can be described by classical nucleation theory (CNT), which has been outlined in Section 6.2, and that the growth kinetics of individual precipitates can be described by the growth laws, which have been presented in Section 6.3.

Both CNT and the growth laws for the evolution of the precipitate radius represent ordinary differential equations. On integration of these equations in time, the evolution of all precipitates in the system is obtained. The following methodology has been suggested by Kampmann and Wagner [KW84] to solve this problem:

- Consider a system with unit volume. Divide time into discrete intervals Δt .
- Evaluate the nucleation rate [equation (6.21)] for this time period. If $J > 0$
 - Create $N_{m+1} = J(t)\Delta t$ precipitates. Collect all new N_{m+1} precipitates in a *precipitate class* with identical radius ρ_{m+1} and identical chemical composition. Let m be the number of existing precipitate classes in the class array.
 - Assign a radius ρ_{m+1} to the precipitate class, which is slightly larger than the critical nucleation radius ρ^* [equation(6.16)].
- Evolve the radius ρ_k of all existing precipitate classes according to $\Delta\rho_k = \dot{\rho}_k\Delta t$, with $k = 1, \dots, m$.
- If the radius ρ_k of a precipitate class drops below a certain limit, remove the precipitate class from the array and decrease m by 1.
- If a new class of precipitates has been nucleated, increase m by 1.
- Repeat these steps until the finish time is reached.

This algorithm is commonly known as the numerical Kampmann–Wagner (NKW) model. With this methodology, in every time interval, a new precipitate class is created (provided that

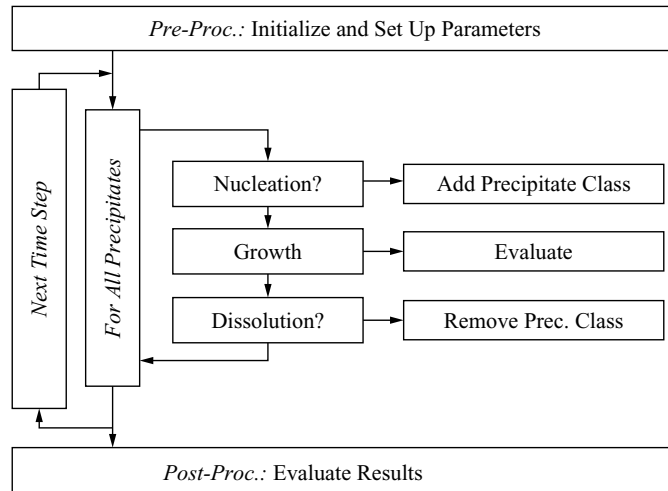


FIGURE 6-13 Typical scheme for numerical integration of the evolution equations for precipitate nucleation and growth according to the numerical Kampmann–Wagner model.

CNT yields a positive nonzero nucleation rate) and the growth kinetics of the existing precipitate classes are evaluated. A flowchart of this procedure is shown in Figure 6-13.

An important feature of the numerical Kampmann–Wagner procedure is that, in following the evolution of numerous precipitate classes simultaneously, information on the evolution of the size distribution of the precipitates is obtained. Figure 6-14 shows typical precipitate size distributions during an isothermal precipitation reaction from a homogeneous supersaturated solution.⁷ Note that the plots of the mean radius and the number density display the logarithm of time on the x -axis and the logarithm of the arithmetic *mean* radius and the logarithmic number density on the y -axis.

In the initial stages of the precipitation process, the nucleation rate increases steadily due to the transient effect of incubation time. The number of precipitates continuously increases as well as the mean radius of the precipitates. If supersaturation decreases because the matrix is slowly depleted from solute atoms (or if all possible nucleation sites have been used) nucleation of new precipitates comes to a stop. At this point, the nucleation rate drops to zero and the mean radius exhibits a linear increase due to the fact that only growth of a constant number of precipitates occurs. Since the growth kinetics are determined by a parabolic law (which is one of the ingredients of the NKW model), the slope of the increase of the mean radius in Figure 6-14 is constant and equal to $1/2$.

If the matrix content of solute atoms approaches the equilibrium value, supersaturation approaches one and precipitate growth comes to a stop. At this point, the curve of the mean radius exhibits a plateau. Within several orders of magnitude of time, the mean precipitate radius and the number density remain constant. The major observable change in the precipitate population before coarsening starts is a transition from a left-shifted size distribution to a right-shifted size distribution (points A→B→C in Figure 6-14).

⁷ The precipitate size distributions and the curves for the mean radius and number density in Figure 6-14 have been computed with the software package MatCalc [KB01] in application of the numerical Kampmann–Wagner algorithm for cementite (Fe_3C) precipitation in the Fe–C system.

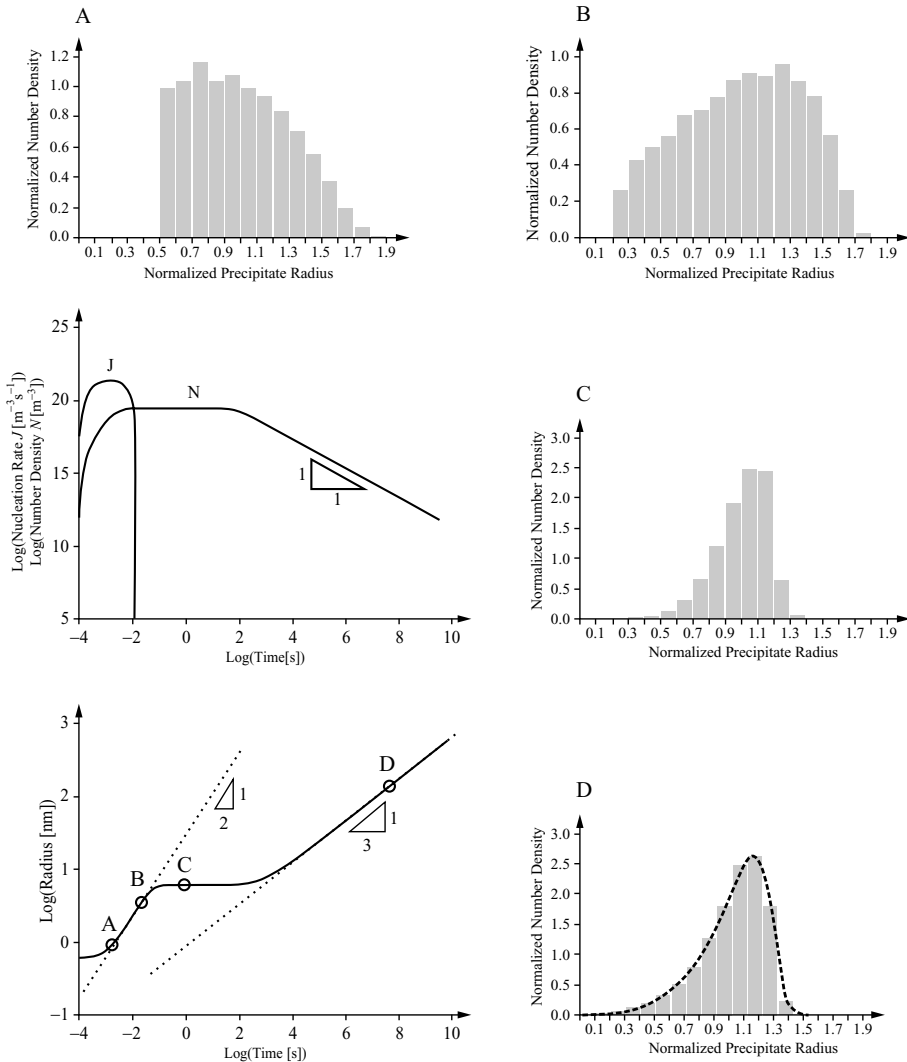


FIGURE 6-14 Evolution of the mean radius and number density of a precipitate population together with representative normalized size distributions $\rho_k/\rho_{\text{mean}}$ in the course of a typical isothermal precipitation reaction (for discussion see text).

In the final stage of the precipitation reaction, the smallest precipitates dissolve in favor of the larger ones (Gibbs–Thomson effect, compare Section 2.2.7). Consequently, the number density of precipitates slowly decreases. In the double logarithmic plot of Figure 6-14, this decrease is linear with time and the slope of the decrease is equal to one. Simultaneously, the mean radius of the precipitate distribution increases continuously with a slope of 1/3. This is consistent with the theoretical value for the coarsening exponent as obtained from the Lifshitz–Slyozov–Wagner (LSW) theory of Ostwald ripening.

In the asymptotic limit of coarsening, the precipitate distribution approaches a stationary state, where the shape of the normalized size distribution is time invariant. In the representative

size distribution for the coarsening regime (point D in Figure 6-14), the distribution obtained with the NKM model is compared to the theoretical distribution of the LSW theory (solid line).

6.4.2 The SFFK Model—A Mean-Field Approach for Complex Systems

In the previous sections, theoretical models for nucleation and growth of individual precipitates have been introduced, and a numerical procedure for simulation of the evolution of entire precipitate populations has been discussed. Traditionally, these approaches are applied to systems with a low number of components due to the fact that the classical concepts for precipitate nucleation and growth have primarily been developed for single-component diffusion. It has also been emphasized that a rigorous solution of the moving boundary problem in multicomponent systems leads to substantial numerical difficulties due to increasing problem complexity with increasing number of components. The major source for these problems stems from the local equilibrium coupling conditions for the chemical potentials across the interface (see Section 6.3.1 for details).

In this section, an alternative treatment of precipitate growth is presented, which is particularly suitable for application to higher-component systems due to a high degree of model abstraction. Nevertheless, the model captures the essential physics of the phase transformation process, in particular effects of multicomponent thermodynamics and diffusion. Details on the model for the general and popular case of the multiple sublattice model (see Section 2.2.8) are published in refs. [SFFK04, KSFF04, KSF05a]. Here, a simplified treatment for only substitutional elements on a single sublattice will be given for the benefit of clarity.

Problem Definition

Consider a chemically homogeneous alloy of unit volume and allow an arbitrary number of spherical particles to precipitate inside this volume. Assume that these precipitates are randomly distributed and that each precipitate is in contact with the surrounding (homogeneous!) matrix. Figure 6-15 presents a sketch of this approach by showing a unit cube of matter containing spherical precipitates of various sizes, chemical composition, and phase type.

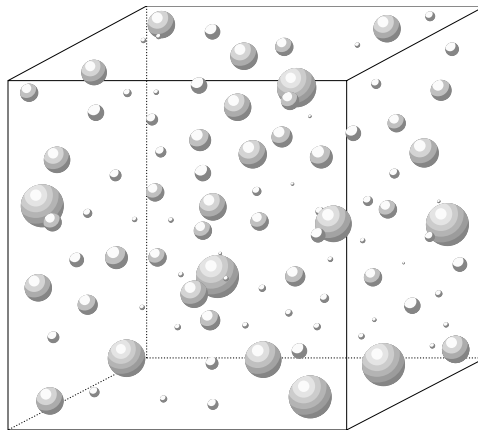


FIGURE 6-15 Geometrical arrangement of randomly distributed precipitates in an otherwise chemically homogeneous alloy.

In a system with n components and m precipitates, the total Gibbs energy of the ensemble can be expressed as

$$G = \sum_{i=1}^n N_{0i} \mu_{0i} + \sum_{k=1}^m \frac{4\pi\rho_k^3}{3} \left(\lambda_k + \sum_{i=1}^n c_{ki} \mu_{ki} \right) + \sum_{k=1}^m 4\pi\rho_k^2 \gamma_k \quad (6.62)$$

where N_{0i} is the number of moles of component i in the matrix and μ_{0i} is the corresponding chemical potential. The subscript “0” is related to quantities of the matrix, whereas the subscript k denotes quantities of the precipitate with the index k . ρ_k is the radius of the precipitate, c_{ki} is the mean chemical composition, λ_k is the bulk mechanical energy, and μ_{ki} the chemical potential of component i . γ_k is the specific interfacial energy of the precipitate–matrix interface.

It is important to recognize that the representation of the total free energy of the system according to the SFFK model involves only *mean quantities* for the composition of the matrix as well as the precipitates. There are no concentration profiles involved in the formulation, which makes the SFFK model a classical mean-field approach. Figure 6-16 sketches the approximated mean concentrations around a single precipitate and indicates the flux of atoms toward the precipitate in the growth stage.

In the expression for the total free energy of the system, equation (6.62), each of the three terms can be related to a particular energy contribution. Accordingly,

- The first term describes the Gibbs energy of the matrix by summing up the product of all moles of components N_{0i} with the corresponding chemical potentials μ_{0i} [compare equation (2.20) in Section 2.1.2].
- The second term is related to the sum of the bulk free energies of all m precipitates, where the specific quantities for the mechanical free energy contribution λ_k and the chemical free energy $\sum_{i=1}^n c_{ki} \mu_{ki}$ are multiplied with the precipitate volume $4/3\pi\rho_k^3$. The summation over k in the second term is performed over all m precipitates.
- The third term takes into account the energy contribution of the precipitate–matrix interfaces. The specific interfacial energy γ_k is multiplied by the precipitate surface area $4\pi\rho_k^2$ and summation is performed over all precipitates k .

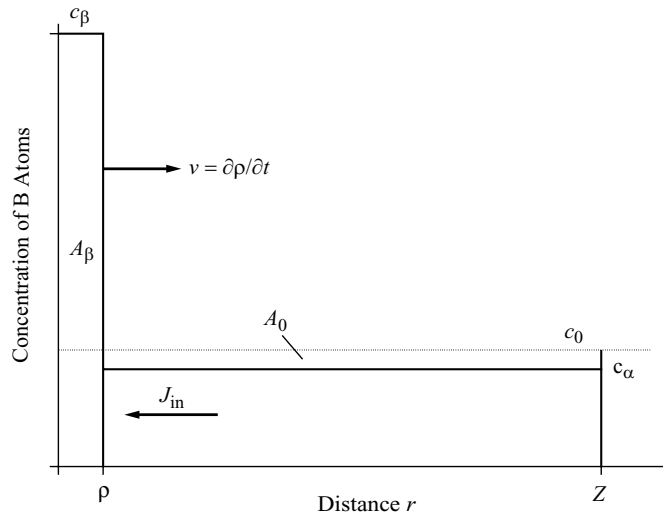


FIGURE 6-16 Mean-field concentrations around a single precipitate in the SFFK approach.

Equation (6.62) describes the total free energy of the system in the present (or actual) state, which is defined by the *independent state parameters* ρ_k and c_{ki} . The other (dependent) parameters can be determined from the mass conservation law for each component i

$$N_i = N_{0i} + \sum_{k=1}^m \frac{4\pi\rho_k^3}{3} c_{ki} \quad (6.63)$$

and the global mass conservation

$$\sum_{i=1}^n N_i = N \quad (6.64)$$

If the system is not in equilibrium (which is necessarily the case if more than one precipitate coexist!), driving force exists for variation of some of the independent state parameters ρ_k and c_{ki} such that the total free energy of the system can be decreased. In other words: The radius and/or the chemical composition of the precipitates in the system will evolve. Goal of the next subsection is to identify the corresponding evolution equations and find the expressions for the rate of change of these quantities as a function of the system state.

Gibbs Energy Dissipation

If a thermodynamic system evolves toward a more stable thermodynamic state, the difference in free energy between the initial and the final state is dissipated. The classical dissipation products in phase transformation reactions are transformation heat (which is transported away) or entropy. In the SFFK model, three dissipation mechanisms are assumed to be operative. These are

- Dissipation by interface movement (\sim friction)
- Dissipation by diffusion inside the precipitate
- Dissipation by diffusion inside the matrix

The first mechanism, that is, the Gibbs energy dissipation due to **interface movement** is founded in the fact that a certain driving pressure is necessary to make an interface migrate. The interface opposes this driving pressure with a force against the pressure, which is comparable in its character to a friction force. This resistance against the driving pressure dissipates energy and the total rate of dissipation due to interface migration can be written as

$$Q_1 = \sum_{k=1}^m \frac{4\pi\rho_k^2}{M_k} \dot{\rho}_k^2 \quad (6.65)$$

with M_k being the interface mobility.

The rate of Gibbs energy dissipation due to **diffusion inside the precipitate** is given by the standard expression

$$Q_2 = \sum_{k=1}^m \sum_{i=1}^n \int_0^{\rho_k} \frac{RT}{c_{ki}D_{ki}} 4\pi r^2 j_{ki}^2 dr \quad (6.66)$$

R is the universal gas constant and j_{ki} is the flux of component i in the precipitate k . If it is assumed that the atoms are homogeneously deposited in or removed from the precipitate, the radial flux is given with

$$j_{ki} = -\frac{r\dot{c}_{ki}}{3}, 0 \leq r \leq \rho \quad (6.67)$$

Substitution of equation (6.66) into (6.67) and integration yields

$$Q_2 = \sum_{k=1}^m \sum_{i=1}^n \frac{4\pi RT \rho_k^5}{45 c_{ki} D_{ki}} \dot{c}_{ki}^2 \quad (6.68)$$

The third contribution is more difficult to obtain and can only be evaluated in an approximate manner. If it is assumed that the distance between the individual precipitates is sufficiently large such that the diffusion profiles of the individual precipitates do not overlap, the **diffusive flux outside the precipitate** can be expressed as

$$Q_3 = \sum_{k=1}^m \sum_{i=1}^n \int_{\rho_k}^Z \frac{RT}{c_{0i} D_{0i}} 4\pi r^2 J_{ki}^2 dr \quad (6.69)$$

where Z is a characteristic length given by the mean distance between two precipitates. The flux J_{ki} can be obtained from the mass conservation law across the interface similar to the treatments presented in Section 6.3. Accordingly, we have

$$(J_{ki} - j_{ki}) = \dot{\rho}_k (c_{0i} - c_{ki}) \quad (6.70)$$

Insertion of equation (6.70) into (6.69) under the assumption $Z \gg \rho_k$ yields the approximate solution

$$Q_3 \approx \sum_{k=1}^m \sum_{i=1}^n \frac{4\pi RT \rho_k^3}{c_{0i} D_{0i}} (\dot{\rho}_k (c_{0i} - c_{ki}) + \rho_k \dot{c}_{ki}/3)^2 \quad (6.71)$$

The total rate of dissipation is finally given as the sum of the individual contributions with $Q = Q_1 + Q_2 + Q_3$.

So far, we have formulated the total Gibbs free energy of a thermodynamic system with spherical precipitates and expressions for the dissipation of the free energy when evolving the system. In order to connect the rate of total free energy change with the free energy dissipation rate, the thermodynamic extremal principle can be used as a handy tool. This principle is introduced in the following section.

The Principle of Maximum Entropy Production

In 1929 and, in extended form, in 1931, Lars Onsager (1903–1976), a Norwegian chemical engineer, published his famous reciprocal relations [Ons31], which define basic symmetries between generalized thermodynamic forces and generalized fluxes. For development of these fundamental relations, Onsager received the Nobel Prize for Chemistry in 1968. In the same paper (and, ironically, in a rather short paragraph), Onsager suggested that a thermodynamic system will evolve toward equilibrium along the one path, which produces maximum entropy. This suggestion is nowadays commonly known as Onsager's thermodynamic extremal principle.

The *thermodynamic extremal principle* or the *principle of maximum entropy production* is not a fundamental law of nature; instead, it is much more of a law of experience. Or it could be a consequence of open-minded physical reasoning. Scientists have experienced that systems, such as the ones that are treated in this context, always (or at least in the vast majority of all experienced cases) behave according to this principle. Therefore, it can be considered as a useful rule and, in a formalistic context, also as a useful and handy mathematical tool. In fact, the thermodynamic extremal principle has been successfully applied to a variety of physical problems, such as cavity nucleation and growth, sintering, creep in superalloy single crystals, grain growth, Ostwald ripening, diffusion, and diffusional phase transformation. In all these

cases, application of the principle offered either new results or results being consistent with existing knowledge, but derived in a most convenient and consistent way.

Let q_i ($i = 1, \dots, K$) be the suitable independent state parameters of a closed system under constant temperature and external pressure. Then, under reasonable assumptions on the geometry of the system and/or coupling of processes, etc., the total Gibbs energy of the system G can be expressed by means of the state parameters q_i ($G = G(q_1, q_2, \dots, q_K)$), and the rate of the total Gibbs energy dissipation Q can be expressed by means of q_i and \dot{q}_i ($Q = Q(q_1, q_2, \dots, q_K, \dot{q}_1, \dot{q}_2, \dots, \dot{q}_K)$). In the case that Q is a positive definite quadratic form of the rates \dot{q}_i [the kinetic parameters, compare equations (6.65), (6.66), (6.69)], the evolution of the system is given by the set of linear equations with respect to \dot{q}_i as

$$\frac{\partial G}{\partial q_i} = -\frac{1}{2} \frac{\partial Q}{\partial \dot{q}_i} \quad (i = 1, \dots, K) \quad (6.72)$$

For a detailed discussion of the theory behind the thermodynamic extremal principle and application to problems in materials science modeling, the interested reader is referred to ref. [STF05].

Evolution Equations

When applying the thermodynamic extremal principle to the precipitation system defined previously in equations (6.62), (6.65), (6.68), and (6.71), the following set of equations has to be evaluated:

$$\frac{\partial G}{\partial \rho_k} = -\frac{1}{2} \frac{\partial Q}{\partial \dot{\rho}_k} \quad (k = 1, \dots, m) \quad (6.73)$$

$$\frac{\partial G}{\partial c_{ki}} = -\frac{1}{2} \frac{\partial Q}{\partial \dot{c}_{ki}} \quad (k = 1, \dots, m; i = 1, \dots, n) \quad (6.74)$$

The matrix of the set of linear equations is, fortunately, not dense, and it can be decomposed for individual values of k into m sets of linear equations of dimension $n + 1$.

Let us denote for a fixed k : $y_i \equiv \dot{c}_{ki}$, $i = 1, \dots, n$, $y_{n+1} \equiv \dot{\rho}_k$. Then the set of linear equations can be written as

$$\sum_{j=1}^{n+1} A_{ij} y_j = B_i \quad (j = 1, \dots, n + 1) \quad (6.75)$$

It is important to recognize that application of the thermodynamic extremal principle leads to linear sets of evolution equations for each individual precipitate, which provide the growth rate $\dot{\rho}_k$ and the rate of change of chemical composition \dot{c}_{ki} on basis of the independent state variables of the precipitation system. For a single sublattice, the coefficients in equation (6.75) are given with

$$A_{n+1n+1} = \frac{1}{M_k} + RT \rho_k \sum_{i=1}^n \frac{(c_{ki} - c_{0i})^2}{c_{0i} D_{0i}} \quad (6.76)$$

$$A_{1i} = A_{i1} = \frac{RT \rho_k^2}{3} \frac{c_{ki} - c_{0i}}{c_{0i} D_{0i}}, \quad (i = 1, \dots, n) \quad (6.77)$$

$$A_{ij} = \frac{RT \rho_k^3}{45} \left(\frac{c_{ki} - c_{0i}}{c_{0i} D_{0i}} \right) \delta_{ij} \quad (i = 1, \dots, n, j = 1, \dots, n) \quad (6.78)$$

The symbol δ_{ij} is the Kronecker delta, which is zero if $i \neq j$ and one if $i = j$. The right-hand side of equation (6.75) is given by

$$B_i = -\frac{\rho_k}{3}(\mu_{ki} - \mu_{0i}) \quad (i = 1, \dots, n) \quad (6.79)$$

$$B_{n+1} = -\frac{2\gamma}{\rho_k} - \lambda_k - \sum_{i=1}^n c_{ki}(\mu_{ki} - \mu_{0i}) \quad (6.80)$$

Detailed expressions for the coefficients of the matrix A_{ij} and the vector B_i for the case of interstitial–substitutional alloys is described in ref. [SFFK04]. A full treatment in the framework of the multiple sublattice model (see Section 2.2.8) is demonstrated in ref. [KSF05a].

Growth Rate for a Stoichiometric Precipitate

For a comparison of the SFFK growth kinetics with the growth equations of Section 6.3, we derive the growth equation for a single stoichiometric precipitate in a binary system. In this case, the precipitate radius ρ_k remains as the only independent state parameter because the precipitate composition is constant. The system of equations (6.75) then reduces to a single equation with the coefficients

$$A\dot{\rho} = B \quad (6.81)$$

For infinite interfacial mobility M_k and neglecting the effect of interface curvature, the coefficients are given as

$$A = RT\rho \sum_{i=1}^n \frac{(c_i^\beta - c_i^0)^2}{c_i^0 D_i^0} \quad (6.82)$$

and

$$B = \sum_{i=1}^n c_i^\beta (\mu_i^\beta - \mu_i^0) \quad (6.83)$$

This term B is equivalent to the chemical driving force for precipitation and we can substitute the expression

$$B = \frac{F}{\Omega} = -\frac{1}{\Omega} \left(\sum_{i=1}^n \text{eq} X_i^{\text{0eq}} \mu_i^0 - X_i^0 \mu_i^0 \right) \quad (6.84)$$

If component B occurs in dilute solution, the chemical potential terms belonging to the majority component A in equation (6.84) can be neglected, since, in dilute solution, we have

$$\text{eq} \mu_A^0 \approx \mu_A^0 \quad (6.85)$$

With the well-known relation for the chemical potential of an ideal solution

$$\mu = \mu_0 + RT \ln(X) \quad (6.86)$$

and insertion into equation (6.84), we obtain

$$B = -\frac{RT}{\Omega} X_B^\beta \left(\ln \frac{\text{eq} X_B^0}{X_B^0} \right) = -RT c_B^\beta \ln \frac{\text{eq} c_B^0}{c_B^0} \quad (6.87)$$

The last step is done in order to make the growth rates comparable with the previous analytical models, which are all expressed in terms of the concentrations c . The subscript “B” is dropped in the following equations and the variable nomenclature of Section 6.3 is used. For the growth rate, we obtain

$$\dot{\rho} = \frac{B}{A} = \frac{-RTc_{\beta} \ln \frac{c_{\alpha\beta}}{c_0}}{RT\rho \frac{(c_{\beta}-c_0)^2}{c_0 D_0}} \quad (6.88)$$

and

$$\rho \dot{\rho} = \frac{-Dc_0 \ln \frac{c_{\alpha\beta}}{c_0}}{(c_{\beta} - c_0)} \quad (6.89)$$

On integration, we finally have

$$\rho = \sqrt{\rho_0^2 - 2Dt \cdot \frac{-c_0 \ln \frac{c_{\alpha\beta}}{c_0}}{(c_{\beta} - c_0)}} \quad (6.90)$$

6.5 Comparing the Growth Kinetics of Different Models

Based on the different analytical models, which have been derived previously, the growth kinetics for the precipitates can be evaluated as a function of the dimensionless supersaturation S , which has been defined as

$$S = \frac{c_0 - c_{\alpha\beta}}{c_{\beta} - c_{\alpha\beta}} \quad (6.91)$$

Figure 6-17 shows the relation between the supersaturation S as defined in equation (6.39) and the *relative supersaturation* $c_{\alpha\beta}/c_0$, which is a characteristic quantity for the SFFK model. Figure 6-18 compares the different growth rates as a function of the supersaturation S . The

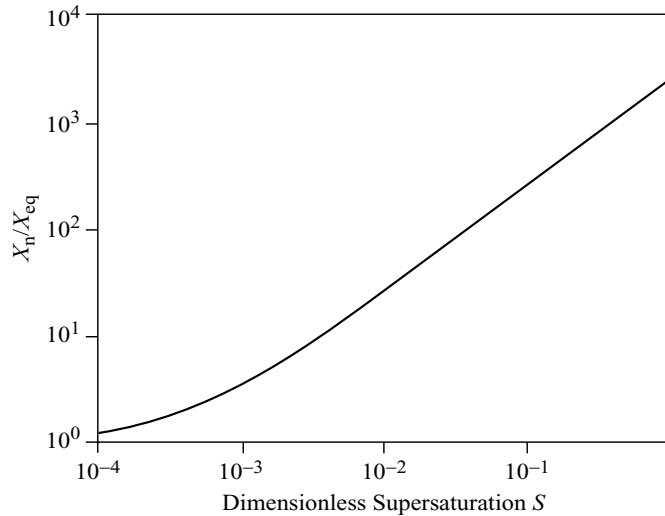


FIGURE 6-17 Relation between the supersaturation S and the relative supersaturation $c_{\alpha\beta}/c_0$.

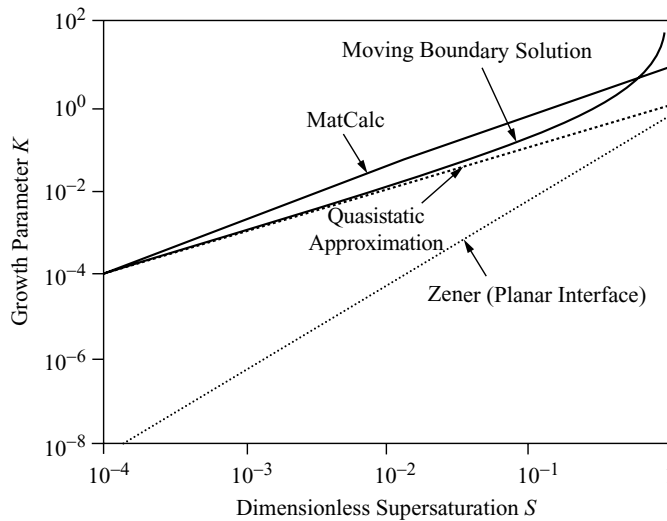


FIGURE 6-18 Comparison of the growth equations for the growth of precipitates. Note that the Zener solution has been derived for planar interface and therefore compares only indirectly to the other two solutions.

curve for the Zener planar interface movement is only drawn for comparison, and it must be held in mind that this solution is valid for planar interfaces, whereas the other three solutions are valid for spherical symmetry.

For low supersaturation, all models for spherical symmetry are in good accordance. Particularly the quasi-static approach exhibits good agreement with the exact moving boundary solution as long as S is not too high. Substantial differences only occur if S becomes larger. In view of the fact that the SFFK model is a mean-field model with considerable degree of abstraction, that is, no detailed concentration profiles, the agreement is reasonable.

Bibliography

- [Aar99] H. I. Aaronson. *Lectures on the Theory of Phase Transformations*. TMS, PA, 2 Ed., 1999.
- [Avr39] M. Avrami. Kinetics of phase change. i: General theory. *J. Chem. Phys.*, pp. 1103–1112, 1939.
- [Avr40] M. Avrami. Kinetics of phase change. ii: Transformation-time relations for random distribution of nuclei. *J. Chem. Phys.*, 8:212–224, 1940.
- [Avr41] M. Avrami. Kinetics of phase change. iii: Granulation, phase change, and microstructure kinetics of phase change. *J. Chem. Phys.*, 9:177–184, 1941.
- [Bec32] R. Becker. Die Keimbildung bei der Ausscheidung in metallischen Mischkristallen. *Ann. Phys.*, 32:128–140, 1932.
- [BW34] W. L. Bragg and E. J. Williams. The effect of thermal agitation on atomic arrangement in alloys. *Proc. R. Soc. London*, 145:699–730, 1934.
- [Chr02] J. W. Christian. *The Theory of Transformations in Metals and Alloys, Part I and II*. Pergamon, Oxford, 3rd Ed., 2002.
- [dVD96] A. Van der Ven and L. Delaey. Models for precipitate growth during the $\gamma \rightarrow \alpha + \gamma$ transformation in Fe–C and Fe–C–M alloys. *Prog. Mater. Sci.*, 40:181–264, 1996.
- [Gli00] M. E. Glicksman. *Diffusion in Solids*. Wiley, New York, 2000.

- [Hil98] M. Hillert. *Phase Equilibria, Phase Diagrams and Phase Transformations—Their Thermodynamic Basis*. Cambridge University Press, Cambridge, 1998.
- [JAA90] B. Jönsson J. O. Andersson, L. Höglund, and J. Agren. *Computer Simulation of Multicomponent Diffusional Transformations in Steel*, pp. 153–163, 1990.
- [JM39] W. A. Johnson and R. F. Mehl. Reaction kinetics in processes of nucleation and growth. *Transactions of American Institute of Mining and Metallurgical Engineers (Trans. AIME)*, 135:416–458, 1939.
- [KB01] E. Kozeschnik and B. Buchmayr. *MatCalc A Simulation Tool for Multicomponent Thermodynamics, Diffusion and Phase Transformations*, Volume 5, pp. 349–361. Institute of Materials, London, Book 734, 2001.
- [Kha00] D. Khashchiv. *Nucleation—Basic Theory with Applications*. Butterworth–Heinemann, Oxford, 2000.
- [Kol37] A. N. Kolmogorov. Statistical theory of crystallization of metals. (in Russian). *Izvestia Akademia Nauk SSSR Ser. Mathematica (Izv. Akad. Nauk SSSR, Ser. Mat; Bull. Acad. Sci. USSR. Ser. Math)*, 1:355–359, 1937.
- [Kos01] G. Kostorz, ed. *Phase Transformations in Materials*. Wiley-VCH Verlag GmbH, Weinheim, 2001.
- [KSF05a] E. Kozeschnik, J. Svoboda, and F. D. Fischer. Modified evolution equations for the precipitation kinetics of complex phases in multicomponent systems. *CALPHAD*, 28(4):379–382, 2005.
- [KSF05b] E. Kozeschnik, J. Svoboda, and F. D. Fischer. On the role of chemical composition in multicomponent nucleation. In *Proc. Int. Conference Solid-Solid Phase Transformations in Inorganic Materials, PTM 2005, Pointe Hilton Squaw Peak Resort, Phoenix, AZ, U.S.A., 29.5.–3.6.2005*, pp. 301–310, 2005.
- [KSFF04] E. Kozeschnik, J. Svoboda, P. Fratzl, and F. D. Fischer. Modelling of kinetics in multi-component multi-phase systems with spherical precipitates II.—numerical solution and application. *Mater. Sci. Eng. A*, 385(1–2):157–165, 2004.
- [KW84] R. Kampmann and R. Wagner. Kinetics of precipitation in metastable binary alloys—theory and applications to Cu-1.9 at % Ti and Ni-14 at % AC. *Acta Scripta Metall.*, pp. 91–103, 1984. Series, Decomposition of alloys: the early stages.
- [Ons31] L. Onsager. *Reciprocal Relations in Irreversible Processes*, Vol. 37, pp. 405–426. (1938); Vol. 38, pp. 2265–2279 (1931).
- [PE04] D. A. Porter and K. E. Easterling. *Phase Transformations in Metals and Alloys*. CRC Press, Boca Raton, FL, 2 Ed., 2004.
- [Rus80] K. Russell. Nucleation in solids: The induction and steady state effects. *Adv. Colloid Interf. Sci.*, 13:205–318, 1980.
- [SFFK04] J. Svoboda, F. D. Fischer, P. Fratzl, and E. Kozeschnik. Modelling of kinetics in multicomponent multi-phase systems with spherical precipitates I.—theory. *Mater. Sci. Eng. A*, 385(1-2):166–174, 2004.
- [STF05] J. Svoboda, I. Turek, and F. D. Fischer. Application of the thermodynamic extremal principle to modeling of thermodynamic processes in material sciences. *Phil. Mag.*, 85(31):3699–3707, 2005.
- [Tur55] D. Turnbull. Impurities and imperfections. *American Society of Metals*, pp. 121–144, 1955.
- [Zen49] C. Zener. Theory of growth of spherical precipitates from solid solution. *J. Appl. Phys.*, 20:950–953, 1949.

This page intentionally left blank

7 Phase-Field Modeling

—*Britta Nestler*

The following sections are devoted to introducing the phase-field modeling technique, numerical methods, and simulation applications to microstructure evolution and pattern formation in materials science. Model formulations and computations of pure substances and of multicomponent alloys are discussed. A thermodynamically consistent class of nonisothermal phase-field models for crystal growth and solidification in complex alloy systems is presented. Expressions for the different energy density contributions are proposed and explicit examples are given. Multicomponent diffusion in the bulk phases including interdiffusion coefficients as well as diffusion in the interfacial regions are formulated. Anisotropy of both, the surface energies and the kinetic coefficients, is incorporated in the model formulation. The relation of the diffuse interface models to classical sharp interface models by formally matched asymptotic expansions is summarized.

In Section 7.1, a motivation to develop phase-field models and a short historical background serve as an introduction to the topic, followed by a derivation of a first phase-field model for pure substances, that is, for solid–liquid phase systems in Section 7.2. On the basis of this model, we perform an extensive numerical case study to evaluate the individual terms in the phase-field equation in Section 7.3. The finite difference discretization methods, an implementation of the numerical algorithm, and an example of a concrete C++ program together with a visualization in MatLab is given. In Section 7.4, the extension of the fundamental phase-field model to describe phase transitions in multicomponent systems with multiple phases and grains is described. A 3D parallel simulator based on a finite difference discretization is introduced illustrating the capability of the model to simultaneously describe the diffusion processes of multiple components, the phase transitions between multiple phases, and the development of the temperature field. The numerical solving method contains adaptive strategies and multigrid methods for optimization of memory usage and computing time. As an alternative numerical method, we also comment on an adaptive finite element solver for the set of evolution equations. Applying the computational methods, we exemplarily show various simulated microstructure formations in complex multicomponent alloy systems occurring on different time and length scales. In particular, we present 2D and 3D simulation results of dendritic, eutectic, and peritectic solidification in binary and ternary alloys. Another field of application is the modeling of competing polycrystalline grain structure formation, grain growth, and coarsening.

7.1 A Short Overview

Materials science plays a tremendous role in modern engineering and technology, since it is the basis of the entire microelectronics and foundry industry, as well as many other industries. The manufacture of almost every man-made object and material involves phase transformations and solidification at some stage. Metallic alloys are the most widely used group of materials in industrial applications. During the manufacture of castings, solidification of metallic melts occurs involving many different phases and, hence, various kinds of phase transitions [KF92]. The solidification is accompanied by a variety of different pattern formations and complex microstructure evolutions. Depending on the process conditions and on the material parameters, different growth morphologies can be observed, significantly determining the material properties and the quality of the castings. For improving the properties of materials in industrial production, the detailed understanding of the dynamical evolution of grain and phase boundaries is of great importance. Since numerical simulations provide valuable information of the microstructure formation and give access for predicting characteristics of the morphologies, it is a key to understanding and controlling the processes and to sustaining continuous progress in the field of optimizing and developing materials.

The solidification process involves growth phenomena on different length and time scales. For theoretical investigations of microstructure formation it is essential to take these multi-scale effects as well as their interaction into consideration. The experimental photographs in Figure 7-1 give an illustration of the complex network of different length scales that exist in solidification microstructures of alloys.

The first image [Figure 7-1(a)] shows a polycrystalline Al–Si grain structure after an electrolytical etching preparation. The grain structure contains grain boundary triple junctions which themselves have their own physical behavior. The coarsening by grain boundary motion takes place on a long timescale. If the magnification is enlarged, a dendritic substructure in the interior of each grain can be resolved. Each orientational variant of the polycrystalline structure consists of a dendritic array in which all dendrites of a specific grain have the same crystallographic orientation. The second image in Figure 7-1(b) displays fragments of dendritic arms as a 2D cross section of a 3D experimental structure with an interdendritic eutectic structure at a higher resolution, where eutectic lamellae have grown between the primary dendritic phase. In such a eutectic phase transformation, two distinct solid phases S_1 and S_2 grow into an undercooled melt if the temperature is below the critical eutectic temperature. Within the interdendritic eutectic lamellae, a phase boundary triple junction of the two solid phases and the liquid

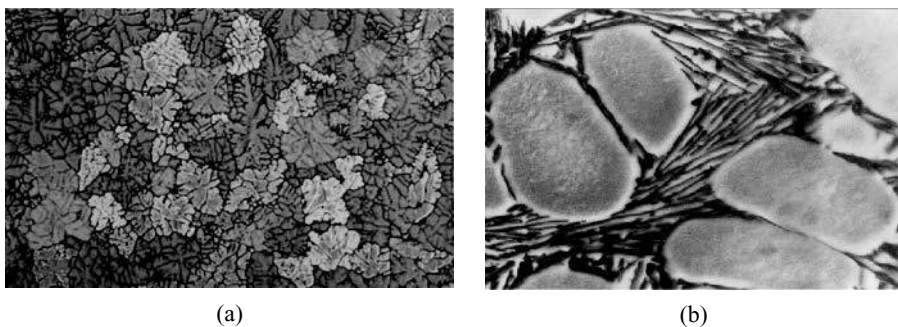


FIGURE 7-1 *Experimental micrographs of Al–Si alloy samples, (a) Grain structure with different crystal orientations and (b) network of primary Al dendrites with an interdendritic eutectic microstructure of two distinguished solid phases in the regions between the primary phase dendrites.*

occurs. The dendrites and the eutectic lamellae grow into the melt on a micrometer scale and during a short time period. Once the dendrites and the eutectic lamellae impinge one another, grain boundaries are formed.

Traditionally, grain boundary motion as well as phase transitions have been described mathematically by moving free boundary problems in which the interface is represented by an evolving surface of zero thickness on which boundary conditions are imposed to describe the physical mechanisms occurring there (see, e.g., Luckhaus and Visintin [LV83], Luckhaus [Luc91], and for an overview we refer to the book of Visintin [Vis96]). In the bulk phases, partial differential equations, for example, describing mass and heat diffusion, are solved. These equations are coupled by boundary conditions on the interface, such as the Stefan condition demanding energy balance and the Gibbs–Thomson equation. Across the sharp interface, certain quantities such as the heat flux, the concentration, or the energy may suffer jump discontinuities. Within the classical mathematical setting of free boundary problems, only results with simple geometries or for small times have been rigorously derived mathematically. In practical computations, this formulation leads to difficulties when the interface develops a complicated geometry or when topology changes occur (compare the computations of Schmidt [Sch96]). Such situations are commonly encountered in growth structures of metallic alloys as can be seen in Figure 7-1.

Since the 1990s, phase-field models have attracted considerable interest as a means of describing phase transitions for a wide range of different systems and for a variety of different influences such as fluid flow, stress, and strain. In particular, they offer a formulation suitable for numerical simulations of the temporal evolution of complex interfacial shapes associated with realistic features of solidification processes. In a phase-field model, a continuous order parameter describes the state of the system in space and time. The transition between regions of different states is smooth and the boundaries between two distinct states are represented by diffuse interfaces. With this diffuse interface formulation, a phase-field model requires much less restrictions on the topology of the grain and phase boundaries.

The phase-field methodology is based on the construction of a Cahn–Hilliard or Ginzburg–Landau energy or entropy functional. By variational derivatives, a set of partial differential equations for the appropriate thermodynamic quantities (e.g., temperature, concentrations) with an additional reaction–diffusion equation for the phase-field variable, often called the phase-field equation, can be derived from the functional. The derivation of the governing equations, although originally ad hoc [Lan86], was subsequently placed in the more rigorous framework of irreversible thermodynamics [PF90, WSW⁺93].

The relationship of the phase-field formulation and the corresponding free boundary problem (or sharp interface description) may be established by taking the sharp interface limit of the phase-field model, whereby the interface thickness tends to zero and is replaced by interfacial boundary conditions. This was first achieved by Caginalp [Cag89], who showed with the help of formally matched asymptotic expansions that the limiting free boundary problem is dependent on the particular distinguished limit that is employed. Later rigorous proofs have been given by Stoth [Sto96] and Soner [Son95]. The sharp interface limit in the presence of surface energy anisotropy has been established by Wheeler and McFadden [WM96]. In further progress, Karma and Rappel [KR96, KR98] (see also [Kar01, MWA00]) have developed a new framework, the so-called thin interface asymptotics, which is more appropriate to the simulation of dendritic growth at small undercoolings by the phase-field model. This analysis has been extended by Almgren [Alm99]. There, the Gibbs–Thomson equation is approximated to a higher order, and the temperature profile in the interfacial region is recovered with a higher accuracy when compared to the classical asymptotics. Further numerical simulations (see refs. [PGD99, PK00, KLP00]) confirm the superiority of this approach in the case of small undercoolings.

Phase-field models have been developed to describe both the solidification of pure materials [Lan86, CF] and binary alloys [LBT92, WBM92, WBM93, CX93, WB94]. In the case of pure materials, phase-field models have been used extensively to simulate numerically dendritic growth into an undercooled liquid [Kob91, Kob93, Kob94, WMS93, WS96, PGD98a]. These computations exhibit a wide range of realistic phenomena associated with dendritic growth, including side arm production and coarsening. The simulations have also been used as a means of assessing theories of dendritic growth. Successively more extensive and accurate computations have been conducted at lower undercoolings closer to those encountered in experiments of dendritic growth [KR96, WS96, PGD98a]. Essential to these computations is the inclusion of surface energy anisotropy, which may be done in a variety of ways [CF, Kob93]. Wheeler and McFadden [WM96, WM97] showed that these anisotropic formulations may be cast in the setting of a generalized stress tensor formulation, first introduced by Hoffman and Cahn [HC72, CH74] for the description of sharp interfaces with anisotropic surface energy. Furthermore, effort has been made to include fluid motion in the liquid phase by coupling a momentum equation to the phase-field and temperature equations [TA98, STSS97]. Anderson *et al.* [AMW98] have used the framework of irreversible thermodynamics to derive a phase-field model in which the solid is modeled as a very viscous liquid. Systems with three phases as well as grain structures with an ensemble of grains of different crystallographic orientations have also been modeled by the phase-field method using a vector valued phase field [CY94, Che95, SPN⁺96, NW, KWC98, GNS98, GNS99a, GNS99b, GN00]. In a system of multiple grains, each component of the vector-valued order parameter characterizes the orientation of a specific crystal. The influence of anisotropy shows the formation of facets in preferred crystallographic directions.

7.2 Phase-Field Model for Pure Substances

For modeling crystal growth from an undercooled pure substance, the system of variables consists of one pure and constant component ($c = 1$), of the inner energy e , and of an order parameter $\phi(\vec{x}, t)$, called the phase-field variable. The value of $\phi(\vec{x}, t)$ characterizes the phase state of the system and its volume fraction in space \vec{x} of the considered domain Ω and at time t . In contrast to classical sharp interface models, the interfaces are represented by thin diffuse regions in which $\phi(\vec{x}, t)$ smoothly varies between the values of ϕ associated with the adjoining bulk phases. For a solid–liquid phase system, a phase-field model may be scaled such that $\phi(\vec{x}, t) = 1$ characterizes the region Ω_S of the solid phase and $\phi(\vec{x}, t) = 0$ the region Ω_L of the liquid phase. The diffuse boundary layer, where $0 < \phi(\vec{x}, t) < 1$, and the profile across the interface are schematically drawn in Figure 7-2.

To ensure consistency with classical irreversible thermodynamics, the model formulation is based on an entropy functional

$$\mathcal{S}(e, \phi) = \int_{\Omega} \left(s(e, \phi) - (\epsilon a(\nabla \phi) + \frac{1}{\epsilon} w(\phi)) \right) dx \quad (7.1)$$

Equation (7.1) is an integral over different entropy density contributions. The bulk entropy density s depends on the phase-field variable ϕ and on the inner energy density e . The contributions $a(\nabla \phi)$ and $w(\phi)$ of the entropy functional reflect the thermodynamics of the interfaces and ϵ is a small length scale parameter related to the thickness of the diffuse interface.

The set of governing equations for the energy conservation and for the non-conserved phase-field variable can be derived from equation (7.1) by taking the functional derivatives $\delta \mathcal{S} / \delta e$ and $\delta \mathcal{S} / \delta \phi$ in the following form:

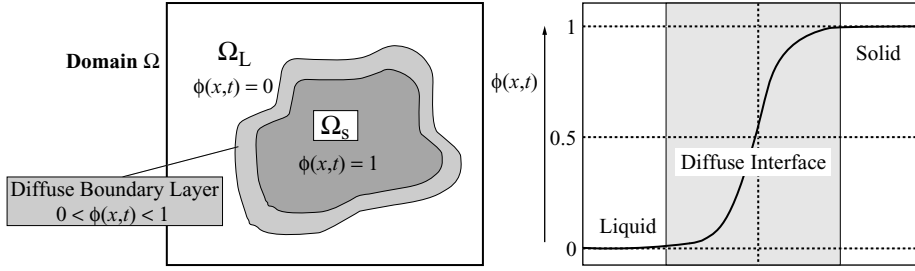


FIGURE 7-2 (Left image): Schematic drawing of a solid–liquid phase system with a bulk solid region Ω_S in dark gray with $\phi(\vec{x}, t) = 1$ surrounded by a bulk liquid Ω_L in white with $\phi(\vec{x}, t) = 0$ and a diffuse interface layer with $0 < \phi(\vec{x}, t) < 1$ in light gray. (right image): Diffuse interface profile of the phase-field variable varying smoothly from zero to one.

$$\frac{\partial e}{\partial t} = -\nabla \cdot \left\{ L_{00}(T, \phi) \nabla \frac{\delta \mathcal{S}}{\delta e} \right\} \quad \text{energy conservation} \quad (7.2)$$

$$\tau \epsilon \frac{\partial \phi}{\partial t} = \frac{\delta \mathcal{S}}{\delta \phi} \quad \text{phase-field equation} \quad (7.3)$$

where τ is a kinetic mobility and T is the temperature. In the case of anisotropic kinetics, τ is a function of $\nabla \phi$ and, hence, depends on the orientation of the phase boundary. $\nabla \cdot \{L_{00}(T, \phi) \nabla (\delta \mathcal{S} / \delta e)\}$ denotes a divergence operator of a flux with the mobility coefficient $L_{00}(T, \phi)$ related to the heat conductivity $\kappa(\phi)$.

For simplicity, we assume κ to be constant $\kappa(\phi) = \kappa$ and write $L_{00} = \kappa T^2$. We make the ansatz $e = -Lh(\phi) + c_v T$ with a latent heat L , a constant specific heat c_v , and a superposition function $h(\phi)$ connecting the two different phase states. $h(\phi)$ can be chosen as a polynomial function fulfilling $h(1) = 1$ and $h(0) = 0$, for example,

$$h(\phi) = \phi, \quad \text{or} \quad (7.4)$$

$$h(\phi) = \phi^2(3 - 2\phi) \quad \text{or} \quad (7.5)$$

$$h(\phi) = \phi^3(6\phi^2 - 15\phi + 10) \quad (7.6)$$

The preceding three choices for the function $h(\phi)$ are displayed in Figure 7-3.

Applying thermodynamical relations, we obtain

$$\frac{\delta \mathcal{S}}{\delta e} = \frac{1}{T}$$

for pure substances and, from equation (7.2), we derive the governing equation for the temperature field $T(\vec{x}, t)$:

$$\frac{\partial T}{\partial t} = k \nabla^2 T + T_Q \frac{\partial h(\phi)}{\partial t} \quad (7.7)$$

where $k = \kappa / c_v$ is the thermal diffusivity and $T_Q = L / c_v$ is the adiabatic temperature.

According to the classical Lagrangian formalism, the variational derivative $\delta \mathcal{S} / \delta \phi$ is given by

$$\frac{\delta \mathcal{S}(e, \phi)}{\delta \phi} = \frac{\partial \mathcal{S}}{\partial \phi} - \nabla \cdot \frac{\partial \mathcal{S}}{\partial (\nabla \phi)}$$

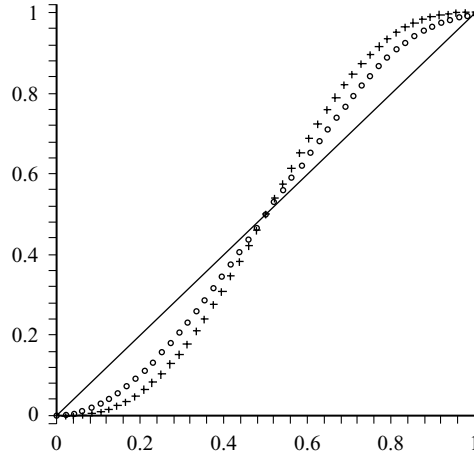


FIGURE 7-3 Plot of the three choices for the function $h(\phi)$ according to equation (7.4) (solid line), equation (7.5) (open circles), and equation (7.6) (crosses).

Together with the thermodynamic relation $e = f + Ts$ and its derivative $\partial s / \partial \phi = -1/T \partial f / \partial \phi$, the phase-field equation can completely be expressed in terms of the bulk free energy density $f(T, \phi)$ instead of the bulk entropy density $s(e, \phi)$. Equation (7.3) for the phase-field variable reads

$$\tau \varepsilon \frac{\partial \phi}{\partial t} = \varepsilon \nabla \cdot a_{,\nabla \phi}(\nabla \phi) - \frac{1}{\varepsilon} w_{,\phi}(\phi) - \frac{f_{,\phi}(T, \phi)}{T} \quad (7.8)$$

where $a_{,\nabla \phi}$, $w_{,\phi}$, and $f_{,\phi}$ denote the partial derivative with respect to $\nabla \phi$ and ϕ , respectively.

Examples of the functions $f(T, \phi)$, $w(\phi)$, and $a(\nabla \phi)$ are

$$\begin{aligned} f(T, \phi) &= L \frac{T - T_M}{T_M} \phi^2 (3 - 2\phi) && \text{bulk free energy density} \\ w(\phi) &= \gamma \phi^2 (1 - \phi)^2 && \text{double well potential} \\ a(\nabla \phi) &= \gamma a_c^2 (\nabla \phi) |\nabla \phi|^2 && \text{gradient entropy density} \end{aligned}$$

where T_M is the melting temperature and γ defines the surface entropy density of the solid–liquid interface. Figure 7-4 illustrates the sum of the functions $w(\phi)$ and $f(T, \phi)$ for the system at two situations: at melting temperature $T = T_M$ and for an undercooling $T < T_M$. The double well function $w(\phi)$ is a potential with two minima corresponding to the two bulk phases solid and liquid. At equilibrium $T = T_M$, both minima are at the same height. Under this condition, a solid–liquid interface without curvature would be stable. If the system temperature is below the melting temperature $T < T_M$, the minimum of the solid phase is lowered driving a phase transition from liquid to solid.

7.2.1 Anisotropy Formulation

The anisotropy of the surface entropy is realized by the factor $a_c(\nabla \phi)$. In two dimensions, an example for the function $a_c(\nabla \phi)$ reads

$$a_c(\nabla \phi) = 1 + \delta_c \sin(M \theta) \quad (7.9)$$

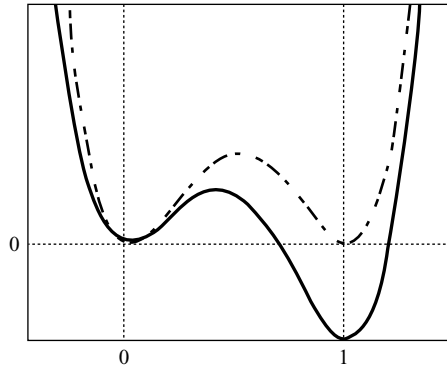


FIGURE 7-4 Plot of $(w(\phi) + f(T, \phi))$ for $T = T_M$ (dash-dotted line) and for $T < T_M$ (solid line).

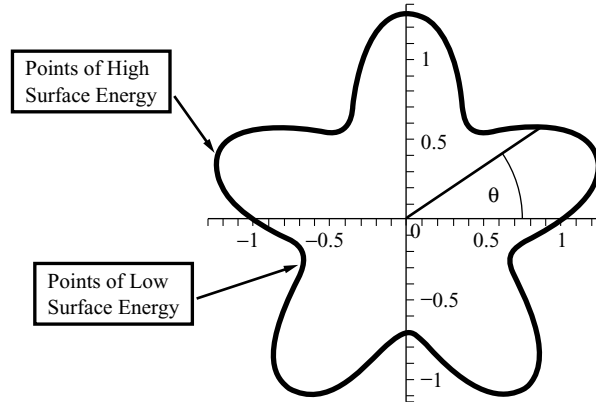


FIGURE 7-5 Two-dimensional contour plot of the smooth anisotropy function $a_c(\nabla\phi) = 1 + 0.3 \sin(5\theta)$ with points of high and points of low surface energy. The corresponding crystal forms its facets into the directions of the five minima of the curve.

where δ_c is the magnitude of the capillary anisotropy and M defines the number of preferred growth directions of the crystal. $\theta = \arccos(\nabla\phi/|\nabla\phi| \cdot \vec{e}_1)$ is the angle between the direction of the interface given by the normal $\nabla\phi/|\nabla\phi|$ and a reference direction, for example, the x -axis \vec{e}_1 . A plot of the anisotropy function $a_c(\nabla\phi)$ for $\delta = 0.3$ and $M = 5$ is shown in Figure 7-5. The shape of the corresponding crystal can be determined by a Wulff construction. For this, a sequence of straight lines is drawn from the origin to the contour line of the entropy function. At each intersection, the perpendicular lines are constructed intersecting with one another. The shape of the crystal results as the interior region of all intersection points. In the case of the five-fold anisotropy function $a_c(\nabla\phi) = 1 + 0.3 \sin(5\theta)$, the five preferred growth directions occur into the directions of the five minima of the surface energy plot.

In the following 3D expression, consistency with an underlying cubic symmetry of the material is assumed:

$$a_c(\nabla\phi) = 1 - \delta_c \left(3 - 4 \frac{1}{|\nabla\phi|^4} \sum_i \left(\frac{\partial\phi}{\partial x_i} \right)^4 \right) \quad (7.10)$$

where $\partial/\partial x_i$ is the partial derivative with respect to the Cartesian coordinate axis x_i , $i = 1, 2, 3$.

7.2.2 Material and Model Parameters

As a first application, we show the simulation of dendritic growth in a pure Ni substance. To perform the simulations in a real system, a number of experimentally measured material and model data are needed as input quantities for the phase-field model. Thermophysical properties for pure nickel have been obtained by Barth *et al.* [BJWH93] using calorimetric methods in the metastable regime of an undercooled melt. The results have been tested by Eckler and Schwarz [Eck92] in a number of verifications of the sharp interface models in comparison with the experimental data. The values of surface energy, atomic attachment kinetics, and their anisotropies are taken from data of atomistic simulations by Hoyt *et al.* [HSAF99, HAK01], which have been linked with the phase-field simulations for analysis of dendritic growth in a wide range of undercoolings [HAK03]. It is remarkable to note that the values for the atomic kinetics given by atomistic simulations are approximately four to five times lower than those predicted by the collision-limited theory of interface advancing [CT82], which can be rather well compared with the values found from previous molecular dynamic simulation data of Broughton *et al.* [BGJ82]. The material parameters used for the simulations of pure nickel solidification are given in Table 7-1.

From these data, a set of parameters for the phase-field model can be computed such as the adiabatic temperature $T_Q = \Delta H/c_v = 418$ K, the microscopic capillary length $d_0 = \sigma_0 T_M / (\Delta H T_Q) = 1.659 \times 10^{-10}$ m, the averaged kinetic coefficient $\beta_0 = (\mu_{100}^{-1} + \mu_{110}^{-1}) / 2T_Q = 5.3 \times 10^{-3}$ s/m, and the strength of the kinetic anisotropy $\epsilon_k = (\mu_{100} - \mu_{110}) / (\mu_{100} + \mu_{110}) = 0.13$.

7.2.3 Application to Dendritic Growth

The solidification of pure nickel dendrites and morphology transformations can be simulated by numerically solving the system of governing equations for the evolution of the temperature and of the phase field [equations (7.7) and (7.8)]. The numerical methods are described in Section 7.4. The formation of dendritic structures in materials depends sensitively on the effect of both surface energy and kinetic anisotropy of the solid-liquid interface. Phase-field simulations of 2D and 3D structures crystallized from an undercooled nickel melt are shown in Figures 7-6 and 7-7.

TABLE 7-1 Material Parameters Used for the Phase-Field Simulations of Dendritic Growth from a Pure Nickel Melt

Parameter	Symbol	Dimension	Ni data	Ref.
Melting temperature	T_M	K	1728	
Latent heat	L	J/m ³	8.113×10^9	[BJWH93]
Specific heat	c_v	J/(m ³ K)	1.939×10^7	[BJWH93]
Thermal diffusivity	k	m ² /s	1.2×10^{-5}	[Eck92, Sch98]
Interfacial free energy	σ_0	J/m ²	0.326	[HAK01]
Strength of interfacial energy	δ_c		0.018	[HAK01]
Growth kinetics in $\langle 100 \rangle$ —crystallographic direction	μ_{100}	m/(sK)	0.52	[HSAF99]
Growth kinetics in $\langle 110 \rangle$ —crystallographic direction	μ_{110}	m/(sK)	0.40	[HSAF99]



FIGURE 7-6 Contour lines of the interfacial position at $\phi = 0.5$ for two neighboring Ni-Cu dendrites at different time steps.

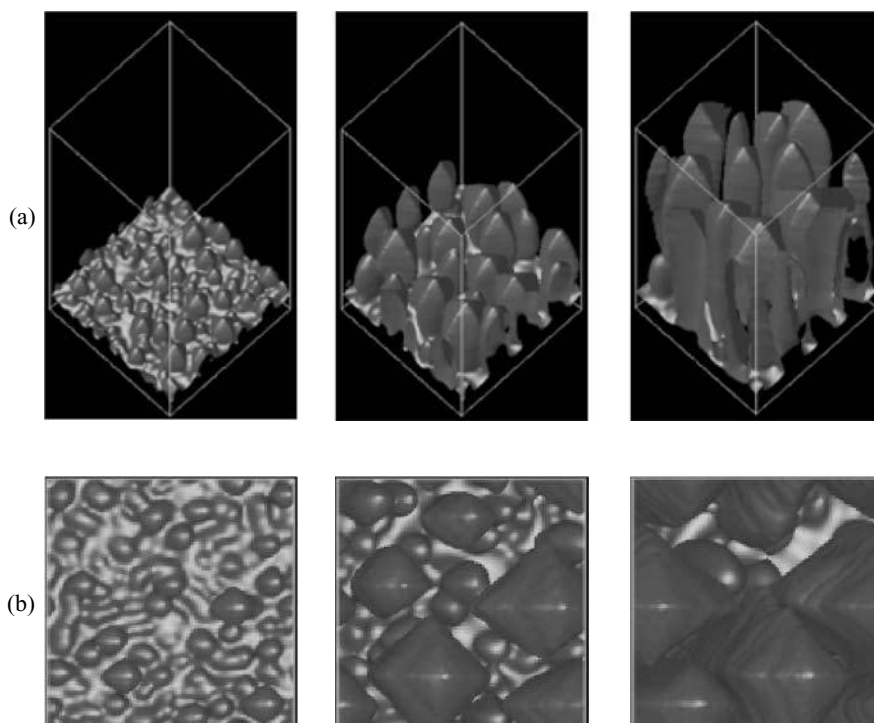


FIGURE 7-7 Three time steps of a Ni-Cu dendritic array formation, (a) 3D view, (b) top view.

The steady state growth dynamics (tip velocity) of pure nickel dendrites in two dimensions and three dimensions is compared with analytical predictions [Bre90] and with recent experimental measurements [FPG⁺04] for different undercoolings in Figure 7-8. For undercoolings $\Delta = (T - T_M)/T_Q$ in the range $0.4 \leq \Delta \leq 0.6$, the simulated tip velocities v match well with the Brener theory in 2D and 3D. The leveling off at higher undercoolings, $\Delta > 0.6$, can

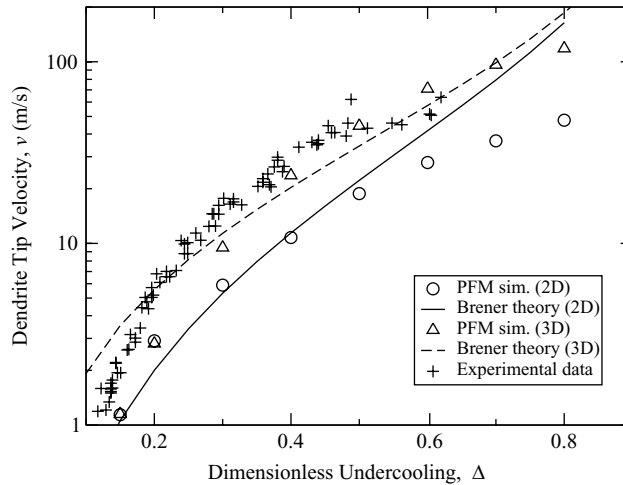


FIGURE 7-8 Tip velocity of nickel dendrites plotted against the dimensionless undercooling Δ for the material parameters given in Table 7-1. The data of 2D and 3D phase-field simulations (circles and triangles) are shown in comparison with the theoretical predictions by Brener [Bre90] (dashed line and solid line). The more recently measured experimental data of pure nickel solidification [FPG⁺04]) confirm the simulation results in the undercooling range $0.4 \leq \Delta \leq 0.6$.

be explained by the divergence of the Brener's theory as Δ tends to one. For 3D dendrites, the experimentally measured data also agree well with the phase-field simulations over the considered undercooling interval $0.4 \leq \Delta \leq 0.6$. For small undercoolings $0.15 \leq \Delta < 0.40$, the disagreement between the experimental data and the phase-field model predictions is attributed to the influence of the forced convective flow in the droplets [FPG⁺04] and to tiny amounts of impurities in the “nominally pure” nickel samples during the experimental procedure of measurements. The convective flow in the droplets enhances the growth velocity in the range of small undercoolings, so that the tip velocity of the dendrites is comparable to the velocity of the liquid flow. The method of numerical simulations can be used to systematically investigate the fundamental influence of the anisotropy, of the undercooling, and of the process conditions on the crystal morphologies.

7.3 Case Study

The set of partial differential equations [equations (7.7) and (7.8)] can be solved using different numerical methods such as, for example, finite differences or finite elements. The efficiency of these algorithms can be optimized applying modern techniques of high performance computing such as adaptive grid methods, multigrid concepts, and parallelization. In the following sections, the system of equations is reduced to a pure solid–liquid phase-field model with a constant temperature (isothermal situation) consisting of only one phase-field equation. The most straightforward discretization of finite differences with an explicit time marching scheme is introduced by writing the phase-field equation in discrete form and by defining suitable boundary conditions. Applying the finite difference method, a case study is presented analyzing the effect of the different terms in the phase-field equation.

7.3.1 Phase-Field Equation

To describe the phase transition in a pure solid–liquid system, the model consists of one phase-field variable ϕ , defining the solid phase with $\phi(\vec{x}, t) = 1$ and the liquid phase with $\phi(\vec{x}, t) = 0$. For further simplification, we consider a system of constant temperature $T = T_{\text{const}}$ and we can hence neglect the dependence of the functional on the variable T . In this case, the Ginzburg–Landau entropy functional is of the form

$$\mathcal{S}(\phi) = \int_{\Omega} s(\phi) - \left(\epsilon a(\nabla\phi) + \frac{1}{\epsilon} w(\phi) \right) dx$$

depending only on the order parameter ϕ . For the system of two phases, the gradient entropy density reads

$$a(\nabla\phi) = \gamma |\nabla\phi|^2 \quad (7.11)$$

where γ is an isotropic surface entropy density. For the potential entropy density, we consider a double well potential

$$w(\phi) = \gamma \phi^2 (1 - \phi)^2 \quad (7.12)$$

As described in Section 7.2 by the equations (7.7) and (7.8), the bulk entropy density $s(\phi)$ can be expressed in terms of the free energy density $f(\phi)$. We choose the third-order polynomial form

$$f(\phi) = m \phi^2 (3 - 2\phi) \quad (7.13)$$

where m is a constant bulk energy density related to the driving force of the process, for example, to the isothermal undercooling ΔT , for example, $m = m(\Delta T)$.

The governing equation for the phase state can be derived by taking the variational derivative of the entropy functional with respect to ϕ ,

$$\tau \epsilon \frac{\partial \phi}{\partial t} = \frac{\delta \mathcal{S}(\phi)}{\delta \phi} = \frac{\partial \mathcal{S}}{\partial \phi} - \nabla \cdot \frac{\partial \mathcal{S}}{\partial (\nabla \phi)} \quad (7.14)$$

Computing the derivatives $a_{,\nabla\phi}(\nabla\phi)$, $w_{,\phi}(\phi)$, and $f_{,\phi}(\phi)$ for the expressions in equations (7.11)–(7.13), gives the phase-field equation

$$\tau \epsilon \partial_t \phi = \epsilon (2\gamma) \Delta \phi - \frac{1}{\epsilon} 18\gamma (2\phi^3 - 3\phi^2 + \phi) - 6m\phi(1 - \phi) \quad (7.15)$$

The equation is completed by a boundary condition at the domain boundary $\delta\Omega$, for example, the natural (or Neumann) boundary condition

$$\nabla\phi \cdot n_{\partial\Omega} = 0 \quad (7.16)$$

Other possible boundary conditions will be discussed later.

7.3.2 Finite Difference Discretization

In the following paragraphs, we formulate the finite difference discretization to numerically solve the equations (7.15) and (7.16) on a regular rectangular 2D computational domain and apply it to some fundamental simulation studies. In numerical analysis, “discretization” refers to passing from a continuous problem to one considered at only a finite number of points. In particular, discretization is used in the numerical solution of a differential equation by reducing

the differential equation to a system of algebraic equations. These determine the values of the solution at only a finite number of grid points of the domain.

In two dimensions, we initially restrict ourselves to a rectangular region of size

$$\Omega = [0, a] \times [0, b] \subset \mathbb{R}^2$$

In this region, a numerical grid is introduced, on which the phase-field equation is solved. The grid is divided into N_x cells of equal size in the x -direction and N_y cells in the y -direction resulting in grid lines spaced at a distance:

$$\delta x = \frac{a}{N_x} \quad \text{and} \quad \delta y = \frac{b}{N_y}$$

The differential equation to be solved is now only considered at the intersection points of the grid lines

$$x_{i,j} = (i\delta x, j\delta y), \quad i = 0, \dots, N_x, \quad j = 0, \dots, N_y$$

The Laplace operator

$$\Delta\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2}$$

is discretized at the grid point $x_{i,j}$ by:

$$(\Delta\phi)_{i,j} = \frac{\phi(x_{i+1,j}) - 2\phi(x_{i,j}) + \phi(x_{i-1,j}))}{\delta x^2} + \frac{\phi(x_{i,j+1}) - 2\phi(x_{i,j}) + \phi(x_{i,j-1}))}{\delta y^2}$$

In the following, we will use a short notation

$$(\Delta\phi)_{i,j} = \frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{\delta x^2} + \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{\delta y^2} \quad (7.17)$$

To discretize the time derivative $\frac{\partial\phi}{\partial t}$, the time interval $[0, t_{\text{end}}]$ is subdivided into discrete times $t_n = n\delta t$ with $n = 0, \dots, N_t$ and $\delta t = t_{\text{end}}/N_t$. The value of ϕ is considered only at times t_n . We use the Euler method to compute the time derivative at time t_{n+1} which employs first-order difference quotients:

$$\left(\frac{\partial\phi}{\partial t}\right)^n = \frac{\phi^{n+1} - \phi^n}{\delta t} \quad (7.18)$$

The superscript n denotes the time level.

If all remaining terms in the differential equation, in particular on the right-hand side, are evaluated at time t_n , the method is called “explicit.” In this case, the solution values at time t_{n+1} are computed solely from those at time t_n . “Implicit methods” evaluate the spatial derivatives at time t_{n+1} and permit the use of much larger time steps while still maintaining stability. However, the implicit methods require the solution of a linear or even nonlinear system of equations in each time step.

Building together the space and time discretizations in equations (7.17) and (7.18), the following discrete, explicit finite difference algorithm of the phase-field equation is obtained:

$$\begin{aligned} \phi_{i,j}^{n+1} = \phi_{i,j}^n + \frac{\delta t}{\tau} \left\{ 2\gamma \left(\frac{\phi_{i+1,j}^n - 2\phi_{i,j}^n + \phi_{i-1,j}^n}{\delta x^2} + \frac{\phi_{i,j+1}^n - 2\phi_{i,j}^n + \phi_{i,j-1}^n}{\delta y^2} \right) \right. \\ \left. - A \frac{1}{\epsilon^2} 18\gamma \left(2(\phi_{i,j}^n)^3 - 3(\phi_{i,j}^n)^2 + \phi_{i,j}^n \right) - B \frac{1}{\epsilon} 6m \left(\phi_{i,j}^n (1 - \phi_{i,j}^n) \right) \right\} \end{aligned} \quad (7.19)$$

Here, we introduced the factors $A, B \in \{0, 1\}$ in order to switch on or off the corresponding terms or the phase-field equation in our later case study.

According to equation (7.19), a simulation is started at $t = 0$ with given initial values for the phase-field variable $\phi_{i,j}^0$ at each grid point (i, j) . The time evolution is incremented by δt in each step of an outer loop until the final time t_{end} is reached. At time step n , the values of the phase field $\phi_{i,j}$, $i = 0, \dots, Nx$, and $j = 0, \dots, Ny$ are stored and those at time step $n + 1$ are computed.

We remark that for more general cases of anisotropic gradient entropy densities of the form

$$a(\nabla\phi) = \gamma a_c(\nabla\phi) |\nabla\phi|^2$$

it is more convenient to discretize the divergence of the variational derivative

$$\nabla \cdot \frac{\partial \mathcal{S}}{\partial(\nabla\phi)} = \nabla \cdot (a_{,\nabla\phi}(\nabla\phi))$$

by using one-sided (forward and backward) differences, for example,

$$\nabla^l \cdot (a_{,\nabla\phi}(\nabla^r\phi))$$

The discrete expressions are

$$(\nabla^r\phi)_{i,j} = \left(\frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x}, \frac{\phi_{i,j+1} - \phi_{i,j}}{\Delta y} \right)$$

and

$$\begin{aligned} \left(\nabla^l \cdot (a_{,\nabla\phi}(\nabla^r\phi)) \right)_{i,j} = \frac{(a_{,\nabla\phi}(\nabla^r\phi))_{i,j} - (a_{,\nabla\phi}(\nabla^r\phi))_{i-1,j}}{\Delta x} \\ + \frac{(a_{,\nabla\phi}(\nabla^r\phi))_{i,j} - (a_{,\nabla\phi}(\nabla^r\phi))_{i,j-1}}{\Delta y} \end{aligned}$$

It is easy to show that in the case of isotropic surface entropy densities with $a_c(\nabla\phi) = 1$ and $a(\nabla\phi) = \gamma |\nabla\phi|^2$, we recover the discretization of the Laplace operator of equation (7.17)

$$(\Delta\phi)_{i,j} = (\nabla^l \cdot (\nabla^r\phi))_{i,j}$$

7.3.3 Boundary Values

We will state three examples of possible domain boundary treatments: 1. Neumann; 2. periodic; and 3. Dirichlet boundary conditions. The discretization [equation (7.19)] of the phase-field equation for ϕ involves the values of ϕ at the boundary grid points:

$$\begin{aligned} \phi_{0,j}, \quad \phi_{Nx,j} \quad \text{with} \quad j = 0, \dots, Ny \\ \phi_{i,0}, \quad \phi_{i,Ny} \quad \text{with} \quad i = 0, \dots, Nx \end{aligned}$$

These values are obtained from a discretization of the boundary conditions of the continuous problem.

1. **Neumann Condition:** The component of $\nabla\phi$ normal to the domain boundary should vanish as in Eq. (7.16). In our rectangular domain, this can be realized by copying the ϕ value of the neighboring (interior) cell to the boundary cell:

$$\begin{aligned}\phi_{0,j} &= \phi_{1,j}, & \phi_{Nx,j} &= \phi_{Nx-1,j} & \text{with } & j = 0, \dots, Ny \\ \phi_{i,0} &= \phi_{i,1}, & \phi_{i,Ny} &= \phi_{i,Ny-1} & \text{with } & i = 0, \dots, Nx\end{aligned}$$

2. **Periodic Condition:** A periodic boundary condition mimics an infinite domain size with a periodicity in the structure. The values of the boundary are set to the value of the neighboring cell from the opposite side of the domain:

$$\begin{aligned}\phi_{0,j} &= \phi_{Nx-1,j}, & \phi_{Nx,j} &= \phi_{1,j} & \text{with } & j = 0, \dots, Ny \\ \phi_{i,0} &= \phi_{i,Ny-1}, & \phi_{i,Ny} &= \phi_{i,1} & \text{with } & i = 0, \dots, Nx\end{aligned}$$

3. **Dirichlet Condition:** The values of ϕ at the domain boundary are set to a fixed initial value:

$$\begin{aligned}\phi_{0,j} &= \phi_W, & \phi_{Nx,j} &= \phi_E & \text{with } & j = 0, \dots, Ny \\ \phi_{i,0} &= \phi_S, & \phi_{i,Ny} &= \phi_N & \text{with } & i = 0, \dots, Nx\end{aligned}$$

where $\phi_W, \phi_E, \phi_S, \phi_N$ are constant data defined in an initialization file.

At the end of each time step, the Neumann as well as the periodic boundary conditions have to be set using the just computed values of the interior of the domain. The preceding type of Dirichlet condition only needs to be initialized, since the value stays constant throughout the complete computation. Figure 7-9 illustrates the boundary conditions, the grid, and the computational domain.

7.3.4 Stability Condition

To ensure the stability of the explicit numerical method and to avoid generating oscillations, the following condition for the time step δt depending on the spatial discretizations δx and δy must be fulfilled:

$$\delta t < \frac{1}{4\gamma} \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right)^{-1} \quad (7.20)$$

7.3.5 Structure of the Code

An implementation concept to solve the discrete form of the phase-field equation [equation (7.19)] could be structured as shown in Figure 7-10 in two parts: A C++ program code with the solving routine and a MatLab application for visualization of the simulation results. Starting with a parameter file “params.cfg,” the program reads the configuration file, sets the internal variables, and parses the parameters to the initialization. In this part of the program, the memory is allocated in accordance with the domain size, the MatLab output file is opened, and the initial data are filled. The main process follows where the computation of each grid point and the time iteration takes place. To illustrate the evolution of the phase field in time and space, the C++ program produces a MatLab script file, for example, “data_file.m” as output file. This file contains a number of successive ϕ matrices at preselected time steps. Applying further self-written

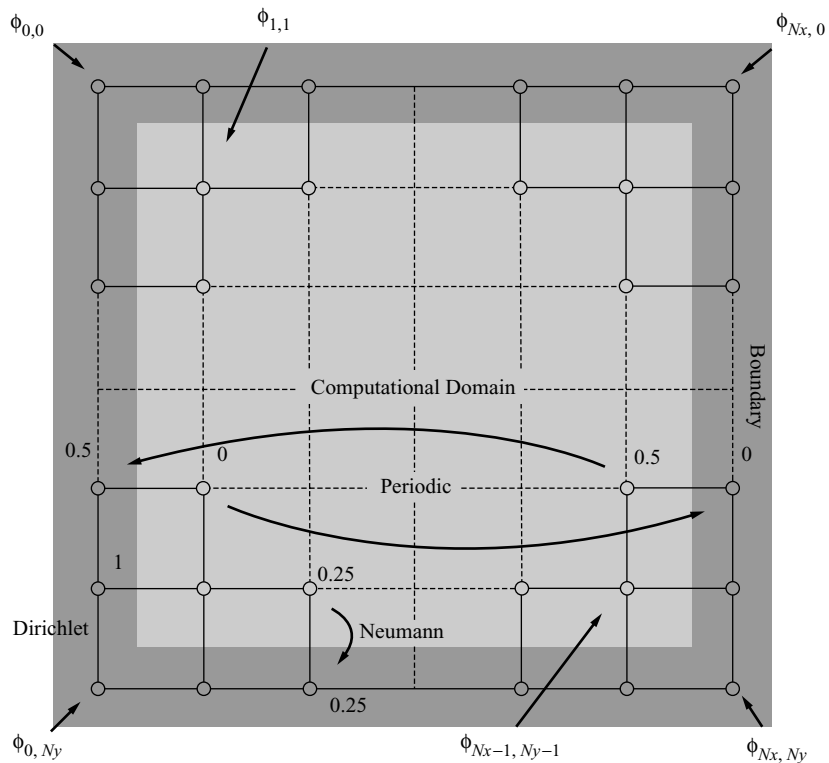


FIGURE 7-9 Schematic drawing of the Neumann, periodic, and Dirichlet boundary conditions within the numerical grid.

MatLab scripts “show_ij.m,” “show_xy.m,” or “show_3d.m” for visualization of the ϕ matrices shows images of the phase field as 1D curves, 2D colored pictures, or as a 3D surface plot.

For visualization, MatLab needs to be started and the “Current Directory” should be set to the folder where the MatLab files are in. Typing of the three commands show_ij, show_xy, and show3d will start the execution of the views.

7.3.6 Main Computation

The main iteration takes place in the method `void computation()` which contains one loop of the time iteration and two loops over the interior of the 2D domain (without boundary). In this routine the gradient of the phase-field variable $\nabla\phi$ is determined with forward differences. Example code lines are given in the Listing 7-1.

The structure of a C++ program is shown in Figure 7-11.

LISTING 7-1 Example Code of the Main Time and Spatial Loops

```

1
2 /* Computation of the domain matrix and storing of frames */
3
4 void computation() {
5
6     for (n=0; n<max_n; n++) { // Loop over max_n time steps
7         BoundaryCondition(oldMatrix);

```



```

8
9 // Computation of grad phi with forward differences
10 // Loop over the 2D domain
11 for (j=0; j<max_j-1; j++) {
12     for (i=0; i<max_i-1; i++) {
13         dphi[0][i][j] = oldMatrix [i+1][j] - oldMatrix [i][j];
14         dphi[0][i][j] = dphi[0][i][j] / delta_x;
15         dphi[1][i][j] = oldMatrix [i][j+1] - oldMatrix [i][j];
16         dphi[1][i][j] = dphi[1][i][j] / delta_y;
17     }
18 }
19
20 // Computation of the divergence and of the rhs of the
21 // phase-field equation
22 for (j=1; j<max_j-1; j++) {
23     for (i=1; i<max_i-1; i++) {
24         newMatrix[i][j] = rhsPhasefield(oldMatrix, i, j);
25     }
26 }
27
28 // A temporary pointer is needed to exchange
29 // the old matrix with the newMatrix
30 tempMatrix = oldMatrix;
31 oldMatrix = newMatrix;
32 newMatrix = tempMatrix;
33 }
34 }

```

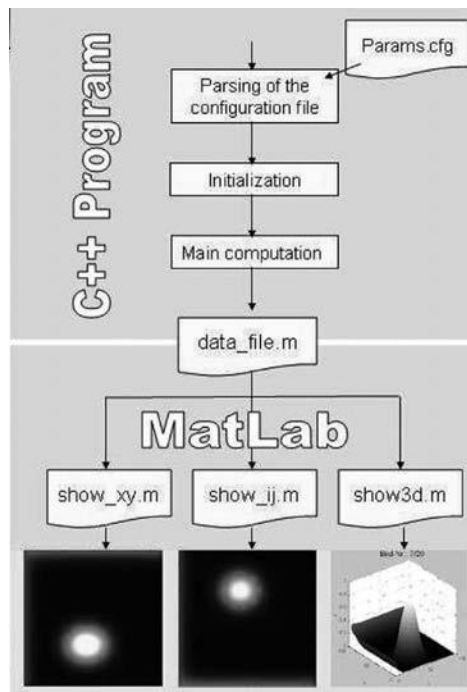


FIGURE 7-10 A possible program structure of a phase-field simulation code.

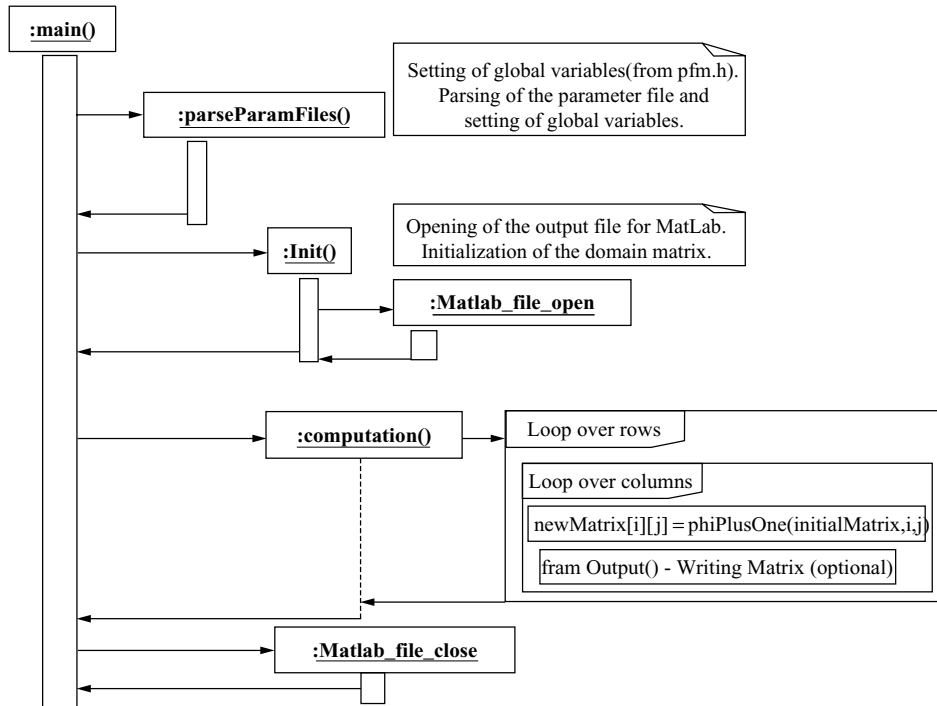


FIGURE 7-11 Possible structure of a phase-field program.

The function `rhsPhasefield(initialMatrix, i, j)` uses the old values of the phase-field variable at time step n to compute the right-hand side of the discretized phase-field equation (equation (7.19)) and to determine new values of the phase-field variable at time step $n + 1$, because of the time marching scheme is explicit. The first argument of the method `double rhsPhasefield(double** argPhi, int i, int j)` contains the values of the phase field at each grid point at time step n . Here, the pointer of the complete matrix $\phi_{i,j}^n$ is passed. The integers i and j correspond to the indices of the spatial loops. The implementation of the function contains the calculation of the divergence with backward differences and the summation of all terms on the right-hand side of equation (7.19). The code lines are displayed in the Listing 7-2.

LISTING 7-2 Function to Compute the Divergence and the Right-hand Side of the Phase-Field Equation

```

1
2   /* Function computing the new phi matrix */
3
4   double rhsPhasefield(double** argPhi, int i, int j) {
5
6   double newPhasefield;
7   double divergence;
8   double potentialEnergy;
9   double drivingForce;
10
11  // Computation of the divergence with backward differences
12  divergence = (dphi[0][i][j] - dphi[0][i-1][j])/delta_x

```

```

13     + (dphi[1][i][j] - dphi[1][i][j-1])/delta_y ;
14
15     potentialEnergy = A*w(argPhi[i][j]);
16     drivingForce = B*f(argPhi[i][j]);
17
18     // Summation of terms for the update of the phi matrix
19     newPhasefield = argPhi[i][j] + delta_t * ( 2*gamma*
20         divergence - 1/(epsilon*epsilon)*potentialEnergy
21         - 1/epsilon*drivingForce);
22
23     return newPhasefield;
24 }

```

7.3.7 Parameter File

In the following Listing 7-3, we give a detailed description of the parameter file in ASCII format for the configuration of a simulation and explain the meaning of the variables' names. Lines starting with the symbol *sharp* are signed as comments and will not be parsed to the main program.

LISTING 7-3 Description of Input Data for a Phase-Field Simulation

```

1
2 # # # # # # # #
3 # Parameter file - 'params.cfg'
4 # # # # # # # #
5 #
6 # Boundary conditions:
7 # boundary_type_* defines the type of the boundary condition.
8 # For each boundary a special type can be defined.
9 # Possible values:
10 # 0 = Dirichlet
11 # 1 = Neumann
12 # 2 = Periodic
13 # If the Dirichlet boundary condition is chosen, values
14 # at the boundaries need to be set by rand_wert_*.
15
16 boundary_type_N=0
17 boundary_type_E=0
18 boundary_type_S=0
19 boundary_type_W=0
20 boundary_value_N=1
21 boundary_value_E=1
22 boundary_value_S=1
23 boundary_value_W=1
24
25 # Initial filling:
26 # filling_* is an option to fill a square region.
27
28 filling_x1=40
29 filling_y1=40
30 filling_x2=60
31 filling_y2=60
32
33 # value_interiorbody initializes the filled region
34 # with a defined value. The remaining region is set
35 # to value_exteriorbody.
36
37 value_interiorbody=1

```

```

38 value_exteriorbody=0
39
40 # Parameters of the phase-field equation:
41 # epsilon is the thickness of the diffuse interface.
42
43 epsilon=1.0
44
45 # gamma is the surface entropy density of the interface.
46
47 gamma=1.0
48
49 # eta is the driving force.
50
51 eta=1.0
52
53 # delta_t is the time step
54
55 delta_t=0.001
56
57 # delta_x, delta_y are the cell spacings in x- and y-directions.
58
59 delta_x=0.1
60 delta_y=0.1
61
62 # A, B are coefficients to switch on/off terms of the rhs.
63
64 A=1
65 B=1
66
67 # Computational parameters -
68 # max_n determines the number of computed frames.
69
70 max_n=1000
71
72 # Nx, Ny are the numbers of cells in x- and y-direction.
73
74 Nx=100
75 Ny=100
76
77 # frames defines the distance between frames to be stored.
78
79 frames=50

```

7.3.8 MatLab Visualization

To illustrate the evolution of the phase-field in time and space, the C++ program may produce a MatLab file “data_file.m” as output file. This file contains a number of successive ϕ matrices at preselected time steps. Applying further MatLab script files “show_ij.m,” “show_xy.m,” or “show_3d.m” allows the graphical illustration of the phase field either as 2D colored images or as a 3D surface plot. The Listings 7-4 and 7-5 now give examples of the MatLab codes.

LISTING 7-4 Structure of a MatLab Output File

```

1  %%%
2  % data_file.m - MatLab output file with the matrices
3  % of the stored frames
4  %%%

```

```

5 % Deleting of the old data in the matrix A
6 A = [];
7
8 % Setting the x- and y-dimension of the computational domain
9 max_j = 6; max_i = 4;
10
11 % Pause in seconds between the frames used for the movie
12 p = 0.500000;
13
14 % Total Number of frames
15 n = 5;
16
17 % Frames 1-5 are stored in the matrix A
18 A(:, :, 1) = [0.000000,0.000000,0.000000,0.000000,0.000000,0.000000;
19 1.000000,1.000000,1.000000,0.000000,0.000000,0.000000;
20 1.000000,1.000000,1.000000,0.000000,0.000000,0.000000;
21 1.000000,1.000000,1.000000,0.000000,0.000000,0.000000];
22 ....
23 A(:, :, 5) = [0.211965,0.266338,0.211965,0.114580,0.033851,0.012800;
24 0.397253,0.475835,0.397253,0.211965,0.077528,0.027870;
25 0.475835,0.583743,0.475835,0.266338,0.092771,0.040670;
26 0.397253,0.475835,0.397253,0.211965,0.077528,0.027870];

```

LISTING 7-5 Matlab Script File for Visualization of the Simulation Data as 2D Images

```

1 %%%
2 % show_xy.m - MatLab Visualization of a colored 2D image
3 %%%
4 % Calling of the data file
5 data_file;
6
7 % Loop over n frames illustrating the temporal evolution
8 % of the phase field
9 for k = 1:1:n
10 % imagesc() illustrates the 2D matrices
11 imagesc(A(:, :, k), [0 1]);
12 % axis image keeps the proportions of the graphical axis
13 axis image;
14 axis xy;
15 % Labelling of the axes
16 xlabel('x');
17 ylabel('y');
18 % colormap() defines the colour table
19 colormap (hot);
20 % pause() sets a pause between the frames to slow down the movie
21 pause (p);
22 end;

```

7.3.9 Examples

The aim of this section is to study the behavior of equation (7.19) numerically for some special cases and to experience the influence of the boundary conditions, of the parameters, and of individual terms in the phase-field equation.

For the subsequent case study, the following parameters occurring in equation (7.19) are taken:

$$\tau = 1.0, \epsilon = 1.0, \gamma = 1.0 \text{ and } Nx = 100, Ny = 10, Nt = 1000$$

By choosing appropriate values for A and B given subsequently, the certain terms in the phase-field equation are switched on and off, respectively. The time step δt can be set according to the stability condition in equation (7.20).

Case 1: Energy Equation—Dirichlet and Neumann Boundary Conditions

In this first application, we consider the case in which the phase-field equation reduces to a heat transport equation of the type $\partial_t u(\vec{x}, t) = D \Delta u(\vec{x}, t)$ by setting the control coefficients $A = 0$ and $B = 0$. The influence of different boundary conditions is displayed in Figure 7-12. The diagrams show the profile of the phase-field variable $\phi(\vec{x}, t)$ starting from $\phi(\vec{x}, t) = 0$ at time $t = 0$ everywhere in the domain.

Case 2: Phase-Field Equation

To investigate the phase-field equation, we switch on the potential entropy contribution $w_{,\phi}(\phi)$ and the bulk driving force $f_{,\phi}(\phi)$ by setting the coefficients $A = 1$ and $B = 1$. The gray scales indicate $\phi = 1$ in white, $\phi = 0$ in black, and the diffuse interface region in varying colors.

1. **Diffuse Interface Thickness:** A planar solid–liquid front is placed in the center of the domain at $Nx/2$ with a sharp interface profile, with zero driving force $m = 0$ and with Neumann boundary conditions on each side. The effect of different values of the small length scale parameter: $\epsilon = 1$ and $\epsilon = 10$ responsible for the thickness of the diffuse interface is shown in Figure 7-13.

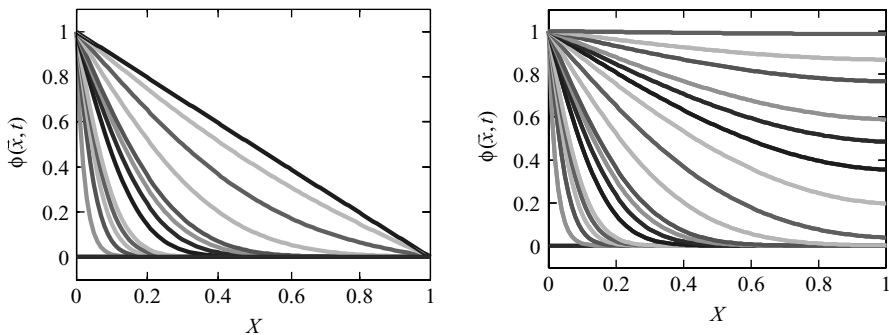


FIGURE 7-12 Left diagram (pure Dirichlet condition): The left (west) boundary is set to $\phi_W = 1$ and the right (east) boundary to $\phi_E = 0$. A linear profile is established; Right diagram (combination of the Dirichlet and the Neumann boundary condition): The left (west) boundary is set constant to $\phi_W = 1$ and the right (east) boundary is isolated (Neumann). The complete domain “warms up” approaching the constant value $\phi = 1$.

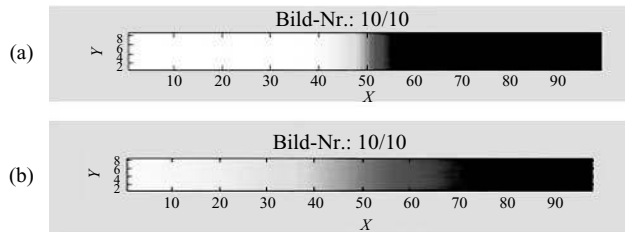


FIGURE 7-13 Diffuse interface profile for (a) $\epsilon = 1$ and (b) $\epsilon = 10$.

2. **Driving Force:** As a next configuration, the three simulations shown in Figure 7-14(b)–(d) were performed with $\epsilon = 1$ with different values of the driving force and with the initial configuration of Figure 7-14(a). For $m = 0$, the initial planar front remains stable, for $m = -1$ the solid phase (light color) grows, whereas for $m = 1$ the solid phase shrinks.

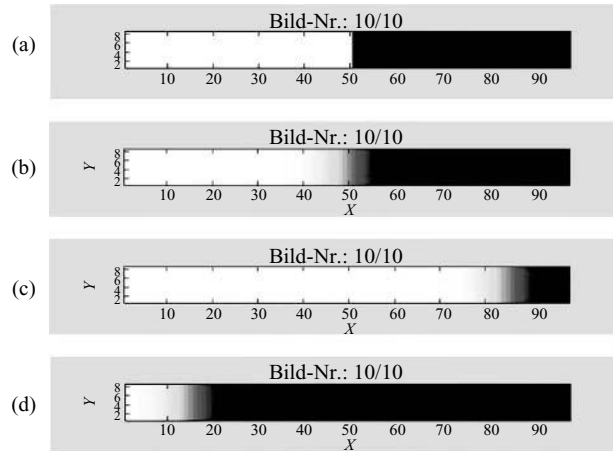


FIGURE 7-14 (a) Initial solid–liquid front with a sharp interface and $\epsilon = 1$, (b) diffuse interface profile for zero driving force ($m = 0$), (c), and (d) moving solid–liquid interface for $m = -1$ and $m = 1$, respectively.

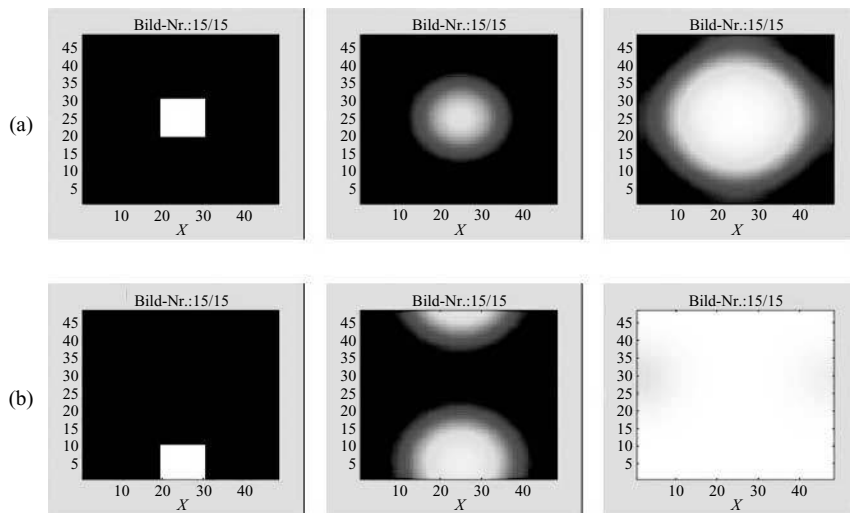


FIGURE 7-15 Two-dimensional phase-field simulation with (a) a solid nucleus in the center of the domain and Neumann boundary conditions and with (b) a solid nucleus at the bottom of the domain and periodic boundary conditions.

Case 3: 2D Phase-Field Simulations of a Growing Nucleus

For the simulation in Figure 7-15, a solid nucleus is set in a 2D domain of $N_x \times N_y = 50 \times 50$ grid points, with $A = 1$, $B = 1$ and with a driving force $m = -2.5$. Due to periodic boundary conditions in Figure 7-15(b) the particle grows across the lower boundary and appears at the top boundary.

7.4 Model for Multiple Components and Phases

7.4.1 Model Formulation

Based on the phase-field model for pure substances, a more general approach can be derived in a thermodynamically consistent way allowing for an arbitrary number of phases (or grains) and components [GNS04]. The model that will be described in the following sections uses N -order parameters to describe either different phases in alloy systems or different orientational variants in polycrystals. The formulation can be defined solely via the bulk free energies of the individual phases, the surface energy densities (surface entropy densities, respectively) of the interfaces, the diffusion and mobility coefficients. Thus, the full set of phase-field evolution equations is defined by quantities which can be measured. Since the bulk free energies determine the phase diagrams (see, e.g., Chalmers [Cha77], Haasen [Haa94]), the phase-field model can be used to describe phase transitions, in principal, for arbitrary phase diagrams. The phase-field model for a general class of multicomponent and multiphase (or polycrystalline) alloy systems is formulated consisting of K components and N different phases (or grains) in a domain $\Omega \subset \mathbb{R}^3$. The domain Ω is separated in phase regions $\Omega_1, \dots, \Omega_N$ occupied by the N phases as schematically illustrated in the left image of Figure 7-16. The middle and right images show examples of an Al-Si grain structure with grains of different crystallographic orientations and of a real multiphase structure with primary dendrites and an interdendritic eutectic substructure.

The concentrations of the components are represented by a vector $\mathbf{c}(\vec{x}, t) = (c_1(\vec{x}, t), \dots, c_K(\vec{x}, t))$. Similarly, the phase fractions are described by a vector-valued-order parameter $\phi(\vec{x}, t) = (\phi_1(\vec{x}, t), \dots, \phi_N(\vec{x}, t))$. The variable $\phi_\alpha(\vec{x}, t)$ denotes the local fraction of phase α . The phase-field model is based on an entropy functional of the form

$$S(e, \mathbf{c}, \phi) = \int_{\Omega} \left(s(e, \mathbf{c}, \phi) - (\epsilon \alpha(\phi, \nabla \phi) + \frac{1}{\epsilon} w(\phi)) \right) dx \quad (7.21)$$

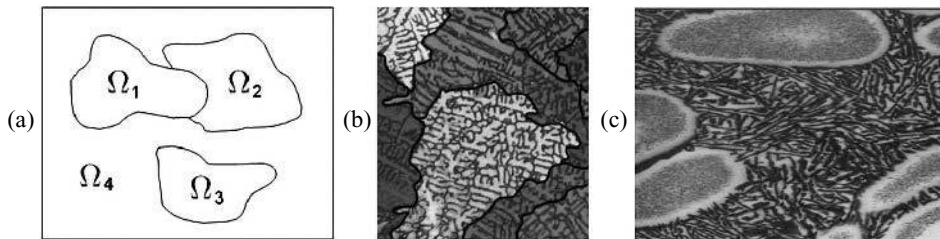


FIGURE 7-16 (a) Schematic drawing of a domain separation by four different phase regions, (b) polycrystalline grain structure, and (c) multiphase solidification microstructure with dendrites and an interdendritic eutectic structure.

We assume that the bulk entropy density s depends on the internal energy density e , the concentrations c , and the phase-field variable ϕ . We require that the concentrations of the components and of the phase-field variables fulfill the constraints

$$\sum_{i=1}^K c_i = 1 \quad \text{and} \quad \sum_{\alpha=1}^N \phi_\alpha = 1 \quad (7.22)$$

7.4.2 Entropy Density Contributions

It will be convenient to use the free energy as a thermodynamical potential. We therefore postulate the Gibbs relation

$$df = -sdT + \sum_i \mu_i dc_i + \sum_\alpha r_\alpha d\phi_\alpha$$

Here, T is the temperature, $\mu_i = f_{,c_i}$ are the chemical potentials, and $r_\alpha = f_{,\phi_\alpha}$ are potentials due to the appearance of different phases. We set

$$e = f + sT \quad (7.23)$$

and hence

$$\begin{aligned} de &= Tds + \sum_i \mu_i dc_i + \sum_\alpha r_\alpha d\phi_\alpha \\ ds &= \frac{1}{T} de - \sum_i \frac{\mu_i}{T} dc_i - \sum_\alpha \frac{r_\alpha}{T} d\phi_\alpha \end{aligned}$$

If we interpret s as a function of (e, c, ϕ) , then we have

$$s_{,e} = \frac{1}{T}, \quad s_{,c_i} = \frac{-\mu_i}{T}, \quad s_{,\phi_\alpha} = \frac{-r_\alpha}{T} \quad (7.24)$$

We note that given the free energy densities of the pure phases, we obtain the total free energy f as a suitable interpolation of the free energies f_α of the individual phases in the system. By inserting the free energy f into the phase-field method enables to model systems with a very general class of phase diagrams. In the way it is formulated, the model can describe systems with concave entropies $s_\alpha(e, c)$ in the pure phases. This corresponds to free energies $f_\alpha(T, c)$ which are convex in c and concave in T . In the case where $f(T, c)$ is not convex in the variable c , the free energy needs to contain gradients of the concentrations (as in the Cahn–Hilliard model).

Choosing the liquid phase to be the last component ϕ_N of the phase-field vector ϕ , an ideal solution formulation of the bulk free energy density reads

$$f_{\text{id}}(T, c, \phi) := \sum_{\alpha=1}^N \sum_{i=1}^K \left(c_i L_i^\alpha \frac{T - T_i^\alpha}{T_i^\alpha} h(\phi_\alpha) \right) + \sum_{i=1}^K \left(\frac{R_g}{v_m} T c_i \ln(c_i) \right) - c_v T \ln\left(\frac{T}{T_M}\right)$$

with $L_i^N = 0$ and L_i^α , $i = 1, \dots, K$, $\alpha = 1, \dots, N - 1$, being the latent heat per unit volume of the phase transition from phase α to the liquid phase and of pure component i . Furthermore, T_i^α , $i = 1, \dots, K$, $\alpha = 1, \dots, N - 1$ is the melting temperature of the i th component in phase α , T_M is a reference temperature. c_v , the specific heat and v_m , the molar volume are assumed to be constant, R_g is the gas constant. With a suitable choice of the function $h(\phi)$ as introduced in equations (7.4) – (7.6) satisfying $h(0) = 0$ and $h(1) = 1$, for example, $h(\phi_\alpha) = \phi_\alpha$ or $h(\phi_\alpha) = \phi_\alpha^2(3 - 2\phi_\alpha)$, the free energy density f is an interpolation of the individual free energy densities f_α . We can calculate

$$s = -f_{,T} = - \sum_{\alpha=1}^N \sum_{i=1}^K \left(c_i \frac{L_i^\alpha}{T_i^\alpha} h(\phi_\alpha) \right) - \sum_{i=1}^K \left(\frac{R}{v_m} c_i \ln(c_i) \right) + c_v \ln(T)$$

so that

$$e = f + Ts = - \sum_{\alpha=1}^N \sum_{i=1}^K (c_i L_i^\alpha h(\phi_\alpha)) + c_v T$$

We note that if $L_i^\alpha = L^\alpha$ for all components i , then e does not depend on c . The chemical potentials $\mu_i(T, \mathbf{c}, \phi)$ are given as the derivative of the free energy density $f(T, \mathbf{c}, \phi)$ with respect to c_i leading to

$$\mu_i(T, \mathbf{c}, \phi) = \sum_{\alpha=1}^N \left(L_i^\alpha \frac{T - T_i^\alpha}{T_i^\alpha} h(\phi_\alpha) \right) + \frac{R}{v_m} T (\ln(c_i) + 1) \quad (7.25)$$

A more general expression for alloys is the Redlich–Kister–Muggianu model of subregular solution

$$f_{\text{sr}} = f_{\text{id}} + \sum_{i=1}^K \sum_{j=1}^K c_i c_j \sum_{\nu=0}^M M_{ij}^{(\nu)} (c_i - c_j)^\nu$$

with binary interaction coefficients $M_{ij}^{(\nu)}$ depending on the parameter ν . For $M=0$, the Redlich–Kister–Muggianu ansatz takes the form of a regular solution model. In most applications, in particular to metallic systems, M takes a maximum value of two. A ternary term $\sim c_i c_j c_k$ can be added to describe the excess free enthalpy.

The thermodynamics of the interfaces gives additional contributions to the entropy given by a Ginzburg–Landau functional of the form

$$- \int_{\Omega} \left(\epsilon a(\phi, \nabla \phi) + \frac{1}{\epsilon} w(\phi) \right) dx$$

Here, $a(\phi, \nabla \phi)$ is the gradient entropy density which is assumed to be homogeneous of degree two in the second variable; namely, $a(\phi, \eta \nabla \phi) = \eta^2 a(\phi, \nabla \phi)$, $\forall \eta \in \mathbb{R}^+$. The simplest form of the gradient entropy density is

$$a(\phi, \nabla \phi) = \sum_{\alpha=1}^N |\nabla \phi_\alpha|^2$$

However, it has been shown [SPN⁺96, GNS98, GNS99b] that gradient entropies of the form

$$a(\phi, \nabla \phi) = \sum_{\alpha < \beta} A_{\alpha\beta} (\phi_\alpha \nabla \phi_\beta - \phi_\beta \nabla \phi_\alpha) \quad (7.26)$$

where $A_{\alpha\beta}$ are convex functions that are homogeneous of degree two, are more convenient with respect to the calibration of parameters in the phase-field model to the surface terms in the sharp interface model. A choice that leads to anisotropic surface terms is

$$a(\phi, \nabla \phi) = \sum_{\alpha < \beta} \gamma_{\alpha\beta} (a_{\alpha\beta}(q_{\alpha\beta}))^2 |q_{\alpha\beta}|^2 \quad (7.27)$$

where $\gamma_{\alpha\beta}$ represents the surface entropy density of the α - β interface and $q_{\alpha\beta} = (\phi_\alpha \nabla \phi_\beta - \phi_\beta \nabla \phi_\alpha)$ is a generalized gradient vector oriented in the direction of the normal to an α - β interface. The formulation using the generalized gradient vectors $q_{\alpha\beta}$ allows to distinguish the physics of each phase (or grain) boundary by providing enough degrees of freedom. Anisotropy of the surface entropy density is modeled by the factor $(a_{\alpha\beta}(q_{\alpha\beta}))^2$ depending on the orientation of the interface. Isotropic phase boundaries are realized by $a_{\alpha\beta}(q_{\alpha\beta}) = 1$. Weakly anisotropic crystals with an underlying cubic symmetry can be modeled in 3D by a straightforward extension of the expression in equation (7.10):

$$a_{\alpha\beta}(q_{\alpha\beta}) = 1 - \delta_{\alpha\beta} \left(3 - 4 \frac{|q_{\alpha\beta}|_4^4}{|q_{\alpha\beta}|^4} \right) \quad (7.28)$$

with $\delta_{\alpha\beta}$ being the strength of the anisotropy of the α - β interface. The norms are given by $|q_{\alpha\beta}|_4^4 = \sum_{i=1}^3 (q_i^4)$ and $|q_{\alpha\beta}|^4 = \left(\sum_{i=1}^3 (q_i^2) \right)^2$ with $q_i = (\phi_\alpha \frac{\partial}{\partial x_i} \phi_\beta - \phi_\beta \frac{\partial}{\partial x_i} \phi_\alpha)$.

For a strongly anisotropic crystal of faceted type, we define

$$a_{\alpha\beta}(q_{\alpha\beta}) = \max_{1 \leq k \leq n_{\alpha\beta}} \left\{ \frac{q_{\alpha\beta}}{|q_{\alpha\beta}|} \cdot \eta_{\alpha\beta}^k \right\} \quad (7.29)$$

where $\eta_{\alpha\beta}^k, k = 1, \dots, n_{\alpha\beta}$ are the $n_{\alpha\beta}$ corners of the Wulff shape of the α - β transition leading to flat crystal faces with sharp edges. These evolve in the direction of the cusps. In principle, equation (7.29) allows to model arbitrary crystal shapes with $n_{\alpha\beta}$ corners. For a comparison, we display in Figure 7-17(a) and (b) the smooth and faceted formulation of the function $a_{\alpha\beta}(q_{\alpha\beta})$ for a cubic crystal symmetry. In Figure 7-17(c) the simulation results of two crystals with 45° rotated orientation growing from adjacent nuclei are shown. Each grain develops its minimum energy surfaces in contact with the melt and at their interface.

The interfacial entropy density contribution $w(\phi)$ is a nonconvex function with N global minima corresponding to the N phases in the system. As an extension of the standard double well potential in equation (7.9) and Figure 7-4, one may take the standard multiwell potential

$$w_{\text{st}}(\phi) = 9 \sum_{\alpha < \beta} \gamma_{\alpha\beta} \phi_\alpha^2 \phi_\beta^2 \quad (7.30)$$

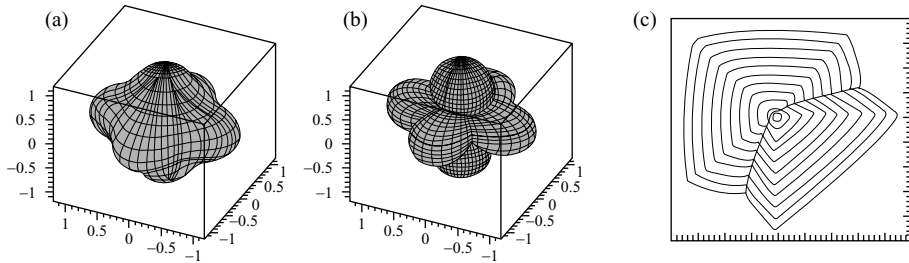


FIGURE 7-17 Three-dimensional surface plot of, (a) a smooth and, (b) a faceted cubic anisotropy, (c) contour plots of two adjacent growing, 45° misoriented cubic crystals applying the smooth anisotropy formulation in equation (7.28) with $\delta = 0.2$.

or a higher-order variant

$$\tilde{w}_{\text{st}}(\phi) = w_{\text{st}}(\phi) + \sum_{\alpha < \beta < \delta} \gamma_{\alpha\beta\delta} \phi_{\alpha}^2 \phi_{\beta}^2 \phi_{\delta}^2$$

For practical computations the multiobstacle potential yields good calibration properties. The expression is

$$w_{\text{ob}}(\phi) = \frac{16}{\pi^2} \sum_{\alpha < \beta} \gamma_{\alpha\beta} \phi_{\alpha} \phi_{\beta} \quad (7.31)$$

with a higher-order variant

$$\tilde{w}_{\text{ob}}(\phi) = w_{\text{ob}}(\phi) + \sum_{\alpha < \beta < \delta} \gamma_{\alpha\beta\delta} \phi_{\alpha} \phi_{\beta} \phi_{\delta}$$

where w_{ob} and \tilde{w}_{ob} are defined to be infinity whenever ϕ is not on the Gibbs simplex. In Figures 7-18 and 7-19, we show a plot of both the multiwell and the multiobstacle potential for the case of three phases ($N = 3$).

We refer to refs. [GNS99a] and [GNS99b] for a further discussion of the properties of the surface terms w_{st} , \tilde{w}_{st} , w_{ob} , and \tilde{w}_{ob} . We assume for simplicity that a and w and, hence, the interfacial contributions to the entropy, do not depend on (T, c) .

7.4.3 Evolution Equations

The energy and mass balance equations can be derived from the energy flux J_0 and from the fluxes of the components J_1, \dots, J_K by

$$\frac{\partial e}{\partial t} = -\nabla \cdot J_0 \quad \text{energy balance} \quad (7.32)$$

$$\frac{\partial c_i}{\partial t} = -\nabla \cdot J_i \quad \text{mass balances} \quad (7.33)$$

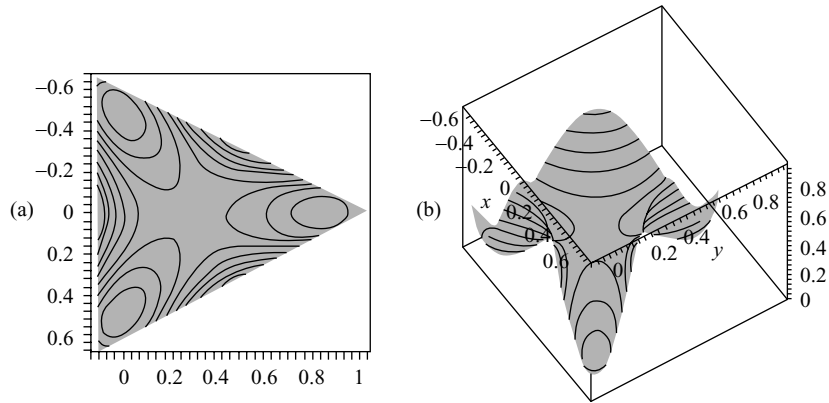


FIGURE 7-18 Plot of the multiwell potential $w_{\text{st}}(\phi)$ for $N = 3$ and equal surface entropy densities $\gamma_{\alpha\beta}$.

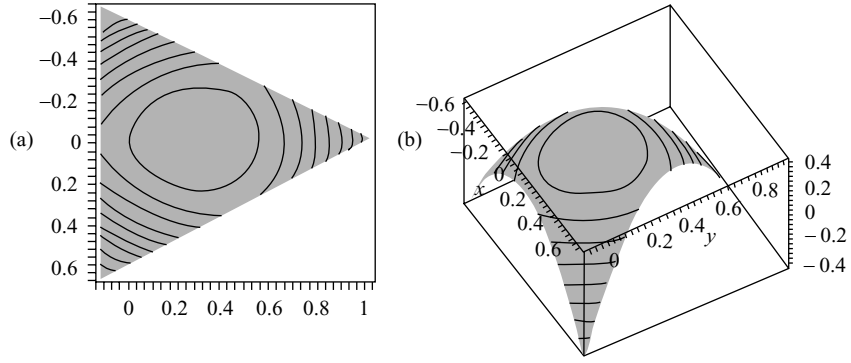


FIGURE 7-19 Plot of the multiobstacle potential $w_{\text{ob}}(\phi)$ for $N = 3$ and equal surface entropy densities $\gamma_{\alpha\beta}$.

and are coupled to a set of phase-field equations

$$\tau \varepsilon \frac{\partial \phi_\alpha}{\partial t} = \frac{\delta S}{\delta \phi_\alpha} - \lambda \quad \text{phase-field equation} \quad (7.34)$$

in such a way that the second law of thermodynamics is fulfilled in an appropriate local version. In order to derive the expressions for the fluxes J_0, \dots, J_K , we use the generalized thermodynamic potentials

$$\frac{\delta S}{\delta e} = \frac{1}{T} \quad \text{and} \quad \frac{\delta S}{\delta c_i} = \left(\frac{-\mu_i}{T} \right)$$

which will drive the evolution. Now we appeal to nonequilibrium thermodynamics and postulate that the fluxes are linear functions of the thermodynamic driving forces $\nabla \frac{\delta S}{\delta e}, \nabla \frac{\delta S}{\delta c_1}, \dots, \nabla \frac{\delta S}{\delta c_K}$ to obtain

$$\begin{aligned} J_0 &= L_{00}(T, c, \phi) \nabla \frac{\delta S}{\delta e} + \sum_{j=1}^N L_{0j}(T, c, \phi) \nabla \frac{\delta S}{\delta c_j} \\ &= L_{00}(T, c, \phi) \nabla \frac{1}{T} + \sum_{j=1}^N L_{0j}(T, c, \phi) \nabla \frac{-\mu_j}{T} \\ J_i &= L_{i0}(T, c, \phi) \nabla \frac{\delta S}{\delta e} + \sum_{j=1}^N L_{ij}(T, c, \phi) \nabla \frac{\delta S}{\delta c_j} \\ &= L_{i0}(T, c, \phi) \nabla \frac{1}{T} + \sum_{j=1}^N L_{ij}(T, c, \phi) \nabla \frac{-\mu_j}{T} \end{aligned}$$

with mobility coefficients $(L_{ij})_{i,j=0,\dots,K}$. To fulfill the constraint $\sum_{i=1}^K c_i = 1$ in equation (7.22) during the evolution, we assume

$$\sum_{i=1}^K L_{ij} = 0, \quad j = 0, \dots, K$$

which implies $\sum_{i=1}^K J_i = 0$, and, hence, $\partial_t(\sum_{i=1}^K c_i) = \nabla \cdot (\sum_{i=1}^K J_i) = 0$. We further assume that L is symmetric (Onsager relations). In addition, L is assumed to be positive semidefinite; that is,

$$\sum_{i,j=0}^K L_{ij} \xi_i \xi_j \geq 0 \quad \forall \xi = (\xi_0, \dots, \xi_K) \in \mathbb{R}^{K+1}$$

This condition ensures that an entropy inequality is satisfied. Cross effects between mass and energy diffusion are included in the model. One can neglect them by setting $L_{i0} = 0$ and $L_{0j} = 0$ for all $i, j \in \{1, \dots, K\}$. In general, the mobility coefficients $(L_{ij})_{i,j=0,\dots,K}$ are allowed to depend on T , \mathbf{c} , and ϕ . Given some heat and mass diffusion coefficients, $k = k(T, \mathbf{c}, \phi)$ and $D_i = D_i(T, \mathbf{c}, \phi)$, the L_{ij} read

$$L_{ji} = L_{ij} = \frac{v_m}{R_g} D_i c_i \left(\delta_{ij} - \frac{D_j c_j}{\sum_{k=1}^K D_k c_k} \right) \quad (7.35)$$

for $i, j = 1, \dots, K$ and then recursively

$$L_{0j} = -\frac{v_m}{R_g} \sum_{\alpha=1}^N \sum_{i=1}^K L_{ji} h(\phi_\alpha) L_i^\alpha \quad (7.36)$$

$$L_{00} = kT^2 + \frac{v_m}{R_g} \sum_{\alpha,\beta}^{N,N} \sum_{i,j}^{K,K} h(\phi_\alpha) L_i^\alpha L_{ji} h(\phi_\beta) L_j^\beta \quad (7.37)$$

where δ_{ij} denotes the Kronecker delta and L_i^α are the latent heats of fusion. The formulation in equations (7.35)–(7.37) takes bulk diffusion effects including interdiffusion coefficients into account. The dependence of the mass and heat diffusion coefficients on ϕ can be realized by, for example, linear expansions. To also consider enhanced diffusion in the interfacial region of phase or grain boundaries, additional terms proportional to $\phi_\alpha \phi_\beta$ with interfacial diffusion coefficients $D_i^{\alpha\beta}(T, \mathbf{c}, q_{\alpha\beta})$ need to be added. Altogether, we suggest for mass and heat diffusion

$$D_i = \sum_{\alpha=1}^N D_i^\alpha(T, \mathbf{c}) \phi_\alpha + \frac{1}{\epsilon} \sum_{\alpha < \beta} D_i^{\alpha\beta}(T, \mathbf{c}, q_{\alpha\beta}) \phi_\alpha \phi_\beta \quad (7.38)$$

$$k = \sum_{\alpha=1}^N k^\alpha(T, \mathbf{c}) \phi_\alpha \quad (7.39)$$

that is, in particular that the diffusion coefficients $D_i^{\alpha\beta}(T, \mathbf{c}, q_{\alpha\beta})$ can be anisotropic.

For the nonconserved phase-field variables ϕ_1, \dots, ϕ_N , we assume that the evolution is such that the system locally tends to maximize entropy conserving concentration and energy at the same time. Therefore, we postulate

$$\begin{aligned} \tau \epsilon \partial_t \phi_\alpha &= \epsilon \left(\nabla \cdot a_{,\nabla \phi_\alpha}(\phi, \nabla \phi) - a_{,\phi_\alpha}(\phi, \nabla \phi) \right) \\ &\quad - \frac{1}{\epsilon} w_{,\phi_\alpha}(\phi) - \frac{f_{,\phi_\alpha}(T, \mathbf{c}, \phi)}{T} - \lambda, \end{aligned} \quad (7.40)$$

where, as already introduced in equation (7.8), we denote with $a_{,\phi_\alpha}$, $w_{,\phi_\alpha}$, $f_{,\phi_\alpha}$, and $a_{,\nabla\phi_\alpha}$ the derivative with respect to the variables corresponding to ϕ_α and $\nabla\phi_\alpha$, respectively. For material systems with anisotropic kinetics, the kinetic coefficient τ may depend on the generalized gradient vectors $q_{\alpha\beta}$ in a similar way as the gradient energies $a(\phi, \nabla\phi)$ in equation (7.27). The quantity $\tau = \tau(\phi, \nabla\phi)$ in equation (7.40) models an anisotropic kinetic coefficient of the form

$$\tau(\phi, \nabla\phi) = \tau_0 + \sum_{\alpha < \beta} B_{\alpha\beta}(q_{\alpha\beta}) \quad (7.41)$$

with $B_{\alpha\beta}(q_{\alpha\beta}) = 0$ if $q_{\alpha\beta} = 0$. Possible choices are

$$B_{\alpha\beta} = \tau_{\alpha\beta}^0 \left(1 + \zeta_{\alpha\beta} \left(3 \pm 4 \frac{|q_{\alpha\beta}|^{\frac{4}{3}}}{|q_{\alpha\beta}|^4} \right) \right) - \tau_0 \quad \text{or} \quad (7.42)$$

$$B_{\alpha\beta} = \tau_{\alpha\beta}^0 \max_{1 \leq k \leq r_{\alpha\beta}} \left\{ \frac{q_{\alpha\beta}}{|q_{\alpha\beta}|} \cdot \xi_{\alpha\beta}^k \right\} - \tau_0, \quad (7.43)$$

if $q_{\alpha\beta} \neq 0$ for weakly cubic [equation (7.42)] or strongly faceted [equation (7.43)] kinetic anisotropies with $r_{\alpha\beta}$ corners $\xi_{\alpha\beta}^k \cdot \zeta_{\alpha\beta}$ determines the strength of the kinetic anisotropy similar to $\delta_{\alpha\beta}$ in equation (7.28) for the surface energy anisotropy. Systems with isotropic kinetics are realized by setting $\zeta_{\alpha\beta} = 0$. The parameter λ is an appropriate Lagrange multiplier such that the constraint $\sum_{\alpha=1}^N \phi_\alpha = 1$ in equation (7.22) is satisfied; that is,

$$\lambda = \frac{1}{N} \sum_{\alpha=1}^N \left[\epsilon \left(\nabla \cdot a_{,\nabla\phi_\alpha}(\phi, \nabla\phi) - a_{,\phi_\alpha}(\phi, \nabla\phi) \right) - \frac{1}{\epsilon} w_{,\phi_\alpha}(\phi) - \frac{f_{,\phi_\alpha}(T, \mathbf{c}, \phi)}{T} \right]$$

It has been shown in ref. [GNS04] by the inequality

$$\begin{aligned} \partial_t(\text{entropy}) &= \partial_t \left(s(e, \mathbf{c}, \phi) - \epsilon a(\phi, \nabla\phi) - \frac{1}{\epsilon} w(\phi) \right) \\ &\geq -\nabla \cdot \left(\sum_{i=0}^K \frac{-\mu_i}{T} J_i - \epsilon \sum_{\alpha=1}^N a_{,\nabla\phi_\alpha} \partial_t \phi_\alpha \right) \end{aligned}$$

that the derivation of the phase-field equation ensures a positive local entropy production.

7.4.4 Nondimensionalization

To nondimensionalize the system with respect to time, space, temperature, and inner energy density, we assume

$$t[\text{s}] = \tau^* \cdot \tilde{t}, \quad x[\text{m}] = \ell \cdot \tilde{x}, \quad T[\text{K}] = \vartheta \cdot \tilde{T}, \quad e \left[\frac{\text{J}}{\text{m}^3} \right] = (c_v \vartheta) \cdot \tilde{e}$$

where \tilde{t} , \tilde{x} , \tilde{T} , \tilde{e} are dimensionless and $\tau^*[\text{s}]$, $\ell[\text{m}]$, $\vartheta[\text{K}]$, $c_v[\text{J}/\text{m}^3\text{K}]$ are reference quantities. The notation $\tilde{\cdot}$ indicates dimensionless quantities. Introducing dimensionless latent heats \tilde{L}_i^α and a rescaled gas constant \tilde{R}_v , the chemical potential μ_j [equation (7.25)] follows as

$$\mu_j \left[\frac{\text{J}}{\text{m}^3} \right] = (c_v \vartheta) \cdot \tilde{\mu}_j \quad \text{with} \quad L_i^\alpha \left[\frac{\text{JK}}{\text{m}^3\text{K}} \right] = (c_v \vartheta) \cdot \tilde{L}_i^\alpha, \quad R \left[\frac{\text{J}}{\text{molK}} \right] = v_m c_v \tilde{R}_v$$

With the dimensionless mobility coefficients [equations (7.35)–(7.37)] $\tilde{L}_{00}, \tilde{L}_{0j}, \tilde{L}_{i0}, \tilde{L}_{ij}$ of the form

$$\begin{aligned} L_{00} \left[\frac{\text{JK}}{\text{sm}} \right] &= \frac{\ell^2 c_v \vartheta^2}{\tau^*} \tilde{L}_{00}, & L_{0j} \left[\frac{\text{m}^2 \text{K}}{\text{s}} \right] &= \frac{\ell^2 \vartheta}{\tau^*} \tilde{L}_{0j}, \\ L_{i0} \left[\frac{\text{m}^2 \text{K}}{\text{s}} \right] &= \frac{\ell^2 \vartheta}{\tau^*} \tilde{L}_{i0}, & L_{ij} \left[\frac{\text{m}^5 \text{K}}{\text{Js}} \right] &= \frac{\ell^2}{c_v \tau^*} \tilde{L}_{ij} \end{aligned}$$

and with the dimensionless mass and heat diffusivities

$$D_i^\alpha \left[\frac{\text{m}^2}{\text{s}} \right] = \frac{\ell^2}{\tau^*} \tilde{D}_i^\alpha \quad \text{and} \quad k^\alpha \left[\frac{\text{J}}{\text{msK}} \right] = \frac{\ell^2 c_v}{\tau^*} \tilde{k}^\alpha$$

the energy and mass diffusion equations [equations (7.32) and (7.33)] can directly be used with dimensionless quantities.

The phase-field equations with the surface contributions $a(\phi, \nabla\phi)$ and $w(\phi)$ are treated as follows: The surface entropy densities $\gamma_{\alpha\beta}$, the kinetic coefficients τ , and the interface width ϵ are scaled as

$$\gamma_{\alpha\beta} \left[\frac{\text{J}}{\text{m}^2 \text{K}} \right] = \gamma_0 \cdot \tilde{\gamma}_{\alpha\beta}, \quad \tau = \tau_0 \cdot \tilde{\tau}, \quad \text{and} \quad \epsilon[\text{m}] = \tilde{\epsilon} \cdot \ell$$

Inserting these quantities into the phase-field equation [equation (7.40)] (both sides with the dimension of an entropy density $[\text{J}/\text{m}^3 \text{K}]$) and division by c_v gives

$$\begin{aligned} \tilde{\tau} \tilde{\epsilon} \frac{\tau_0 \ell}{\tau^* c_v} \partial_i \phi_\alpha &= \tilde{\epsilon} \frac{\gamma_0}{\ell c_v} \left(\tilde{\nabla} \cdot \tilde{a}_{,\tilde{\nabla} \phi_\alpha}(\phi, \tilde{\nabla} \phi) - \tilde{a}_{,\phi_\alpha}(\phi, \tilde{\nabla} \phi) \right) \\ &\quad - \frac{1}{\tilde{\epsilon}} \frac{\gamma_0}{\ell c_v} \tilde{w}_{,\phi_\alpha}(\phi) - \frac{\tilde{f}_{,\phi_\alpha}(T, \mathbf{c}, \phi)}{\tilde{T}} - \tilde{\lambda}, \quad \alpha = 1, \dots, N \end{aligned}$$

By making the choice $\gamma_0 = \ell c_v$ and $\tau = \frac{\ell \tau_0}{c_v}$, the phase-field equation retains its original form (equation 7.40) with dimensionless quantities instead of the dimensional ones.

Finally, if we consider the Gibbs–Thomson equation describing the motion of a sharp interface with the curvature κ

$$\beta_{\alpha\beta} v \left[\frac{\text{m}}{\text{s}} \right] = \gamma_{\alpha\beta} \left[\frac{\text{J}}{\text{m}^2 \text{K}} \right] \kappa \left[\frac{1}{\text{m}} \right] + \frac{([f]_\alpha^\beta - \sum_i \bar{\mu}_i [c_i]_\alpha^\beta)}{T} \left[\frac{\text{J}}{\text{m}^3 \text{K}} \right] \quad (7.44)$$

it can easily be seen that the mobility coefficients $\beta_{\alpha\beta}$ are equal to the kinetic coefficient τ in the phase-field equation. The square brackets $[f]_\alpha^\beta$ and $[c_i]_\alpha^\beta$ indicate a jump of the respective value.

The length scale parameter ℓ can be related to the size of the domain resolved with N_x grid points. In the case of the pure Ni system, the size of a thermal dendrite is $N_x \cdot \ell \cdot \Delta \tilde{x} = 2 \cdot 10^{-5} \text{m}$. With a feasible number of grid points $N_x = 500$ and a dimensionless cell spacing $\Delta \tilde{x} = 1.0$, we have $\ell = 4.0 \cdot 10^{-8} \text{m}$.

7.4.5 Finite Difference Discretization and Staggered Grid

The complete set of evolution equations of the model [equations (7.32)–(7.34)] can be treated in the following divergence form:

$$\frac{\partial u_l}{\partial t} = r h s = \nabla \cdot \vec{f} \left(u_m, \frac{\partial u_m}{\partial x_i} \right) \quad (7.45)$$

where u_l, u_m stands for the respective field quantity (l, m indicative for e, c_i, ϕ_α).

All terms including field variables and their spatial derivatives are arguments of the vector-valued function \vec{f} . In case of the phase-field equation [equation (7.34)] the right-hand side (*rhs*) contains additional source terms, which in general are small by a value compared to the divergence term, so that the general character of equation (7.45) is kept preserved.

Concerning the time discretization, an explicit forward Euler scheme can be applied with a time update according to

$$u_l^{n+1} = u_l^n + \Delta t \cdot (rhs)^n$$

(time steps are superscripted). This explicit scheme requires a control of the temporal step width Δt for each individual equation. For the case of a 3D simulation and an identical grid step width Δx in each space dimension, the criterion for stability suggests a step width of

$$\Delta t \leq \min \left\{ \frac{\Delta x^2}{6k_{\max}}, \frac{\Delta x^2}{6D_{\max}}, \frac{\Delta x^2}{6\frac{\gamma_{\alpha\beta}}{\tau_0}} \right\}$$

where k_{\max} and D_{\max} are the maximum values for all heat and mass diffusion coefficients, $\gamma_{\alpha\beta}$ is the maximum surface entropy coefficient among all appearing phase boundaries, and τ_0 is the kinetic coefficient.

Since the right-hand sides of the conserved order parameter equations [equations (7.32)–(7.33)] as well as the first term in equation (7.34) consist of a divergence term, a two-step algorithm can be applied: First, the vector flux quantities are calculated using right-sided finite differences and they are stored in a memory buffer for multiple access. This buffer holds the flux values of three adjacent 2D layers during the layerwise calculation, shifted through the 3D grid along the z -direction. In a second step, the divergence is evaluated using left-sided differences. This results in an extremely memory saving numerical scheme without any redundant calculations, since for each field variable only a single 3D array must be stored in memory. The spatial discretization of the phase-field and of the balance equations may be treated differently as described in the following paragraph.

Balance Equations

In the nonlinear energy and mass diffusion equations, the physical diffusion coefficients are incorporated in the Onsager coefficients L_{ij} , which may depend on ϕ , c , and T . The discretization on a regular grid with spatial indices (i, j, k) for the following simplified energy equation (mass diffusion cross terms have been omitted for clarity)

$$\frac{\partial e}{\partial t} = -\nabla \cdot \vec{J}_0 = -\nabla \cdot \left(L_{00}(T, c, \phi) \nabla \left(\frac{1}{T} \right) \right) \quad (7.46)$$

can be accomplished in the FTCS scheme (forward in time, centered in space). First, to compute the divergence on the right-hand side of equation (7.46), all components of the energy flux vector are assembled. For a grid cell with indices (i, j, k) the spatial derivatives are approximated with right-sided finite differences, and the Onsager coefficients L_{00} are evaluated at the respective intermediate grid positions, in the center boundary of two adjacent grid cells [see Figure 7-20(b)]. For example, the x component of the energy flux reads:

$$L_{00} \frac{\partial}{\partial x} \left(\frac{1}{T} \right) \Big|_{i,j,k} \simeq L_{00} \Big|_{i+\frac{1}{2},j,k} D_x^+ \left(\frac{1}{T} \right) \Big|_{i,j,k} \quad (7.47)$$

with

$$L_{00} \Big|_{i+\frac{1}{2},j,k} = \frac{1}{2} (L_{00} \Big|_{i+1,j,k} + L_{00} \Big|_{i,j,k})$$

and

$$D_x^+ \left(\frac{1}{T} \right) \Big|_{i,j,k} = \frac{1}{\Delta x} \left(\frac{1}{T} \Big|_{i+1,j,k} - \frac{1}{T} \Big|_{i,j,k} \right)$$

where D_x^+ indicates the forward difference operator in x -direction. As a next step, the divergence operation is carried out using left-sided finite differences of the flux components in equation (7.47). The resulting spatial–temporal scheme has an accuracy of convergence of order $(\Delta x)^2$ in space and of order Δt in time.

Phase-Field Equations

For the phase-field equation [equation (7.34)], the correct treatment of anisotropy in the gradient entropy density $a(\phi, \nabla\phi)$ is important, especially when reducing the numerical interface to a desirable low number of grid points. It is convenient to keep the formulation using generalized gradient vectors $q_{\alpha\beta} = \phi_\alpha \nabla\phi_\beta - \phi_\beta \nabla\phi_\alpha$ [Figure 7-20(a)] in the solution algorithm. We take $a(\phi, \nabla\phi)$ in the form of equations (7.26) and (7.27) and carry out the variational derivatives with respect to $\nabla\phi_\alpha$ and ϕ_α to get the two anisotropic *rhs* terms of the phase-field equation, namely:

$$a_{,\nabla\phi_\alpha} = \sum_{\beta \neq \alpha} \frac{\partial A_{\alpha\beta}}{\partial q_{\alpha\beta}} (-\phi_\beta) \quad (7.48)$$

$$a_{,\phi_\alpha} = \sum_{\beta \neq \alpha} \frac{\partial A_{\alpha\beta}}{\partial q_{\alpha\beta}} \nabla\phi_\beta \quad (7.49)$$

$$\text{with } \frac{\partial A_{\alpha\beta}}{\partial q_{\alpha\beta}} = 2\gamma_{\alpha\beta} \left(a_{\alpha\beta}(q_{\alpha\beta}) \frac{\partial a_{\alpha\beta}}{\partial q_{\alpha\beta}} |q_{\alpha\beta}|^2 + a_{\alpha\beta}^2(q_{\alpha\beta}) q_{\alpha\beta} \right) \quad (7.50)$$

The entropy flux term $a_{,\nabla\phi_\alpha}$ in equation (7.48) needs a special attention due to the influence of the anisotropy function $a_{\alpha\beta}(q_{\alpha\beta})$. Three different vectors $q_{\alpha\beta}^x$, $q_{\alpha\beta}^y$, and $q_{\alpha\beta}^z$ are computed for each grid point, evaluated at the upper side x , y and z boundaries of the cell volume. The spatial derivatives of ϕ_α on these staggered grid positions include a combination of right-sided and central differences of the neighbored grid positions (see Figure 7-20). To compute each spatial component of the vector-valued function $a_{,\nabla\phi_\alpha}$, the respective vector $(q_{\alpha\beta}^x, q_{\alpha\beta}^y, q_{\alpha\beta}^z)$ is used.

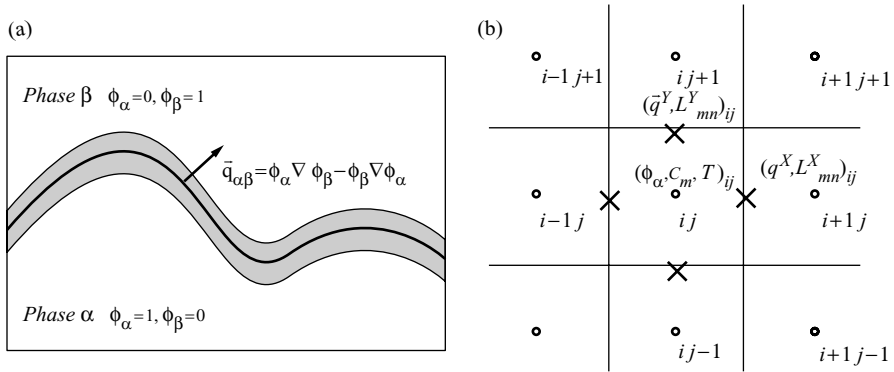


FIGURE 7-20 (a) Schematic view of a two-phase region with a diffuse interface (shaded) and a generalized gradient vector $\vec{q}_{\alpha\beta}$. (b) 2D sketch of the finite difference grid with the field variables ϕ , c , T at central positions (\circ), \vec{q} vectors and with transport coefficients L_{mn} at staggered positions (\times).

The divergence of $a_{,\nabla}\phi_\alpha$ appearing in equation (7.34) is calculated subsequently with left-sided differences, taking advantage of the same three-layer buffer mechanism as described previously. For the second anisotropic term, $a_{,\phi_\alpha}$, the gradient vectors $q_{\alpha\beta}$ are evaluated at central positions using central differences of ϕ_α .

This treatment is equivalent to a 16-point stencil, and it is a second-order accuracy in space that is obtained. For the 2D case the discretization is equivalent to the scheme published in ref. [EJHS00]. If no anisotropy is present, $q_{\alpha\beta}$ can be calculated exclusively with right-sided differences of ϕ_α within a 7-point stencil, in this case sufficient to guarantee second-order accuracy.

7.4.6 Optimization of the Computational Algorithm

Several **adaptive strategies** reduce the computational effort:

- The equations for the phase fields are exclusively solved in their respective diffuse interface area by the use of an activation flag field. This field is set in each time step by an optimized gradient test routine.
- A dynamical memory concept reduces the memory costs, especially for a high number of phase-field variables: For the majority of the grid points, ϕ_α assumes values of 1.0 or 0.0 in all regions of pure bulk phases and can be referenced by pointers to constant unit vectors.

Since both strategies are especially effective with a narrow interface profile, the use of a multiobstacle potential in equation (7.31) has some advantages. This potential has a strong separating character and reduces the number of interface grid points (with $0 < \phi_{i,j,k} < 1$).

For most alloy systems, there are great differences in the thermal and mass diffusivities leading to differences in the evolutionary timescale. If equations (7.32)–(7.33) are solved conjointly, the stability criterion demands the use of the small time step width, which is usually the heat diffusion scale. To reduce the computational effort, different step widths, integer multiples of the smallest (heat diffusion) scale, can be used to solve the three kinds of equations (7.32), (7.33), and (7.34). By taking integer multiples, a synchronization of the individual time discretizations is guaranteed automatically.

7.4.7 Parallelization

Two parallelization concepts of the finite difference algorithm are introduced and can be chosen aside or in combination: For high performance computing on Linux clusters, **distributed computing** is realized via the (LAM-) message passing interface (MPI) routines. The approach splits the 3D simulation grid into multiple subgrids, so that each available node gets assigned a specific part of the simulation space. To avoid unnecessary complexity of the code and to reduce boundary data exchange, the simulation is exclusively subdivided along the z -direction. In this way, minimal interdependencies between neighboring simulation subdomains and a maximum speedup factor can be realized. By mapping the number N_z of grid points in z -direction to its value n_z in each node subdomain, the same code base as well as simulation description can serve for serial as well as for distributed simulations, effectively decoupling the simulation from the hardware executing it. An MPI-based boundary exchange mechanism ensures that before each simulation time step commences, the outer planes of each node are copied into the excess space, provided by its immediate neighbor.

An additional approach to exploit the power of multiprocessor workstations and supercomputers can be realized by **shared memory parallelization** via OpenMP: the execution of the spacial loops is subdivided into different threads running on different processors of a single node. Therefore, only slight modifications of the code are necessary, for example, the introduction of indexed loop variables.

An important feature for parallel phase-field simulations is the implementation of an appropriate load balancing mechanism. Due to the advancing fronts, which require more calculations, the demand for computational power is locally nonuniform. This can be achieved by performing statistics of the load of each node and of the data transfer times. A redistribution of the simulation area after an optimal number of time steps is initiated. Additionally, to adapt the code to networks with lower capabilities requiring data compression, the run-length and quadtree encoding of the exchanged boundary data need to be evaluated.

7.4.8 Adaptive Finite Element Method

An alternative way to solve the evolution equations is the use of an adaptive finite element method such as provided by the differential equation analysis library (DEAL II, [BK99]).

As outlined by Provatas *et al.* [PGD98b], a numerical method based on an adaptive mesh refinement discretization exhibits a computationally efficient technique to solve the phase-field equations describing the evolution and dynamics of phase boundaries. The usage of an adaptive finite element method allows microstructure simulations with high spatial resolutions at relatively low computing times. To employ a finite element discretization, the evolution equations are solved in a weak form with a semi-implicit time scheme. Exploiting the utilities of the DEAL library, the adaptive mesh refinement works with a self-defined criterion. After a predefined number of time steps, the mesh is locally adopted, that is, refined or coarsened, by the following criterion. For each numerical cell, a “gradient indicator” is computed by the expression

$$E = \left(\sum_{\alpha=1}^N |\nabla \phi_{\alpha}| + \chi_c \sum_{i=1}^K |\nabla c_i| + \chi_T |\nabla T| \right) d^2 \quad (7.51)$$

where d is the diagonal of a cell and χ_c, χ_T are constants. The cell will be refined if $E > E_{\max}$ and coarsened if $E < E_{\min}$. By this choice, $d \sim 1/\sqrt{E}$ and hence, the criterion relation has a direct physical meaning. In regions with high gradients of the physical fields ϕ_{α}, c_i , and T , the mesh is finer than elsewhere in the domain, see for example Figure 7-21.

7.4.9 Simulations of Phase Transitions and Microstructure Evolution

Since the early 1990s, the phase-field methodology has emerged as a powerful mathematical and computational modeling technique to capture a wide variety of crystal growth and solidification structures. In the following sections, simulation examples of phase transitions in binary and ternary alloys and grain growth in polycrystalline material are presented.

We consider the case of binary peritectic and eutectic systems with components A and B . Both types of alloy systems consist of three phases; two solid phases which we will denote α and β as well as a liquid phase L . Eutectic and peritectic growth plays an important role in the field of solidification, partly, because both systems exhibit a rich variety of complex microstructures that is far from well understood. In addition, eutectics are often used in castings and offer the potential of producing high strength composite materials by solidification alone.

In a peritectic system, beneath the peritectic temperature, a new solid phase β is formed from the so-called parent solid phase α and from the undercooled liquid via a peritectic reaction $L + \alpha \rightarrow \beta$. Nucleation of the β phase often occurs at the α - L interface. The β phase then grows around the α phase until either the α phase is completely melted or it is entirely engulfed within the new β phase. Examples of systems with peritectic phase transitions are Fe-C, Sn-Sb, as well as magnetic and high temperature superconducting alloys.

To construct a peritectic phase diagram, we choose a free energy density $f(T, c, \phi)$ with dimensionless melting temperatures $T_A^{\alpha} = 0.85, T_A^{\beta} = 0.7, T_B^{\alpha} = 1.35, T_B^{\beta} = 1.55$ and latent

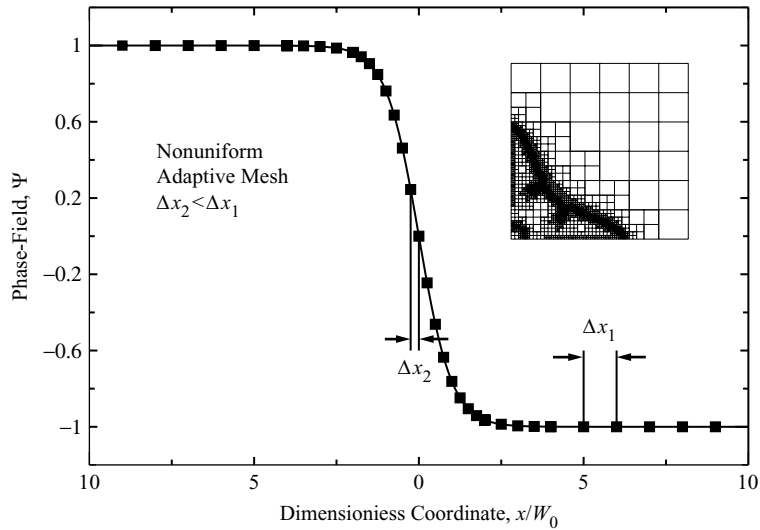


FIGURE 7-21 Adaptive mesh at the region of a dendritic surface.

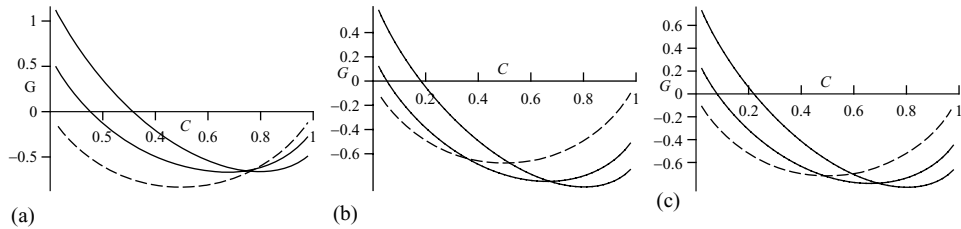


FIGURE 7-22 Bulk free energies of the three phases α , β , and liquid L for, (a) $T = 1.2$ (above), $T = 1.0$ (at), and $T = 0.98$ (below the peritectic temperature).

heats $L_A^\alpha = 1.79$, $L_A^\beta = 1.55$, $L_B^\alpha = 1.55$, $L_B^\beta = 1.79$. The solute compositions on either side of a stationary interface are related by the common tangent construction to the free energy surfaces, which in turn determines the phase diagram. The reference temperature is taken to be the peritectic temperature $T^* = 1.0$. The peritectic phase diagram can be derived by computing the common tangents. The curves of the bulk free energy densities $f_\alpha(T, c)$ for the three phases $\alpha = 1, 2, 3$ are displayed in Figure 7-22(a)–(c) for three different temperatures $T = 1.2$, $T = 1.0$, and $T = 0.98$. A common tangent of all three curves can be observed at $T = 1.0$ where all three phases are in equilibrium. The peritectic phase diagram in Figure 7-23 is obtained from these energy densities.

The simulations shown in Figure 7-24 were performed at a temperature $T = 0.98$, just below the peritectic temperature. According to the peritectic phase diagram, the equilibrium concentrations of the two solid phases at this temperature ($T = 0.98$) are $c = 0.53$ and 0.67 for the β and α phases, respectively. For the initial concentration in the liquid phase, we chose a mean value of $c = 0.32$, which lies between the β liquidus line and the metastable extension of the α liquidus line. The diffusivities in the two solid phases D_α and D_β were set to be zero. The images in both columns of Figure 7-24 illustrate the growth of the new β solid phase on top of the parent α solid phase. A small β nucleus was introduced on top of the melting α solid phase. Immediately, the β phase grows along the α – L phase boundary. Since the supply of

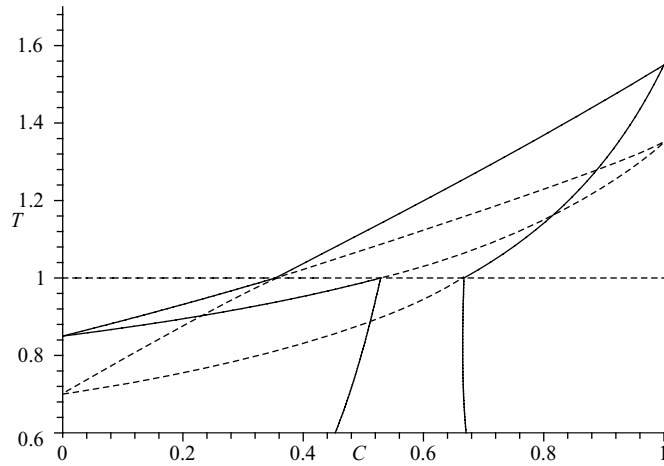


FIGURE 7-23 A phase diagram of a binary peritectic alloy constructed by the method of common tangents.

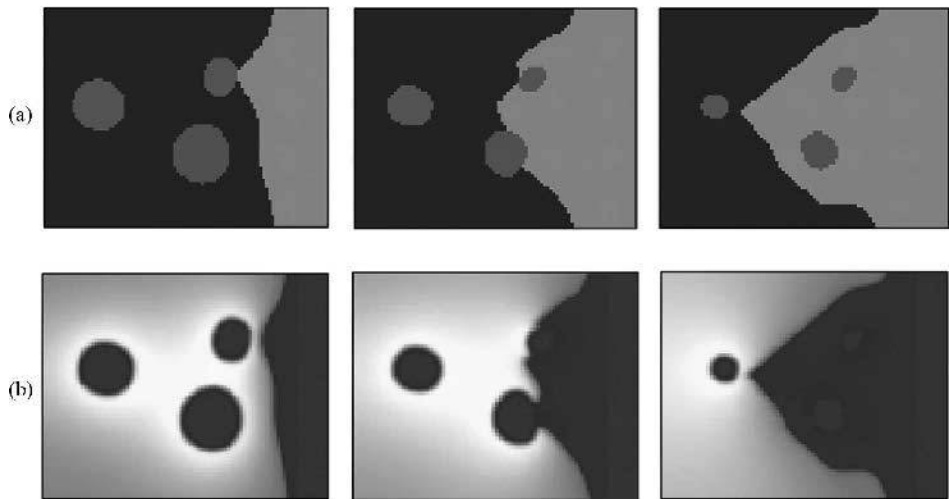


FIGURE 7-24 Phase-field simulations of peritectic phase transformations where a new β solid phase (front to the right) grows from an undercooled melt and from a parent α solid phase (particles in the melt). The images show three subsequent time steps of (a) the different phases and of (b) the concentration in the melt during the evolution.

solute needed for the growth of β phase is provided by dissolving of α solid phase and by diffusion in the liquid, the trijunctions formed by all three phases lead the growth direction. Once the β solid completely covers the α phase, the α - β interface is stationary in the case of a purely liquid diffusion controlled process with zero solid diffusivity. In this situation, the α phase no longer melts after the α -liquid interface has disappeared. However, the β phase continues to grow from the liquid phase alone with a diminishing solidification velocity. The last colored pictures in each of the two columns of Figure 7-24 illustrate the concentration field in the liquid phase.

In a eutectic system, two thermodynamically distinct solid phases (labeled α and β) may solidify from an undercooled liquid phase (L) when its temperature is beneath the eutectic temperature, that is, $L \rightarrow \alpha + \beta$. In directional solidification, the advancing of regular steady state growth structures of the two solid phases can be observed forming periodic structures of eutectic lamellae with alternating α and β phase regions or rods with one of the solid phases embedded in a matrix of the other solid phase. The characteristic eutectic phase diagram can be constructed in an analogous way as indicated for the peritectic system. The symmetric eutectic phase diagram in Figure 7-25 is constructed for a free energy density $f(c, \phi)$ with equal values of the latent heats $L_i^\alpha = 7.0, i = 1, 2, \alpha = 1, 2, 3$. The melting temperatures of the components A and B are chosen to be symmetric in the α - and β -solid phases with the dimensionless values $T_A^{S_1} = T_B^{S_2} = 1.0$ and $T_A^{S_2} = T_B^{S_1} = 0.7$. The simulations in Figure 7-26 are performed with different initial conditions corresponding to the eutectic composition c_E , to an off-eutectic composition c_1 , and with temperatures T_1 and T_2 above and below the eutectic line.

In Figure 7-26(a) and (b), the evolution of an alternating S_1 – S_2 front at a temperature T_1 above the eutectic point T_E for two initial compositions c_1 and c_E is investigated. At the initial configuration (c_1, T_1) , primary growth of S_1 solid phase occurs in the form of a cellular array [see Figure 7-26(a)]. The initially placed S_2 phase dissolves in the liquid. The growing S_1 solid phase rejects the component B into the intercellular region and into the liquid. At the point (c_E, T_1) , the system is in the region of pure liquid phase, and it can be observed in Figure 7-26(b) that both solid phases S_1 and S_2 are thermodynamically unstable. The initially set lamellar S_1 – S_2 front melts. A and B atoms are released from the S_1 and S_2 solid phases and are redistributed by diffusion in the liquid to finally result in a homogeneous distribution of concentrations in the melt. In Figure 7-26(c) and (d) an undercooled system at a temperature T_2 below the eutectic temperature T_E is considered. At both conditions (c_1, T_2) and (c_E, T_2) , the two solid phases S_1 and S_2 cooperatively grow into the liquid by forming a regular lamellar structure. The concentration field of B atoms in the liquid is shown. During growth, the two solid phases enhance each others' growth conditions by rejecting and adopting opposite components. According to the different initial compositions c_1 and c_E , the volume fractions of S_1 and S_2 are different. At the off-eutectic composition c_1 , the volume fraction of S_1 is larger than the volume fraction of S_2 , Figure 7-26(c), whereas at the eutectic composition c_E , the two solid phases grow with equal phase fractions in the case of a symmetric phase diagram, Figure 7-26(d).

By defined cuts along certain lines of the computational domain, concentration profiles and diffusion lengths can be extracted from the phase-field simulations. Figure 7-27 displays the variation of component B in growth/melting direction for the numerical experiments at the initial points (c_1, T_1) , (c_E, T_1) , (c_1, T_2) , and (c_E, T_2) . The according diagrams perpendicular to the growth direction ahead of the solid–liquid front are shown in Figure 7-28. The selected positions, at which the concentration profiles were taken, are marked in Figure 7-26 by solid and dashed lines. For primary phase solidification, the rejected B atoms lead to the formation of cells by the Mullins–Sekerka instability mechanism. It can be observed that the S_1 solid phase [solid line of Figure 7-27(a)] grows with a composition of the solidus line in accordance with the phase diagram. An enrichment of B atoms in the regions between the S_1 cells can be seen in the diagram in Figure 7-27(a) (dashed line). In the case of eutectic growth, Figure 7-26(c) and (d), both solid phases evolve with the compositions $c_{S_1} = 0.05$ and $c_{S_2} = 0.95$ of the solidus lines (solid and dashed lines in Figure 7-27(c) and (d), respectively).

The rejected B atoms form a concentration enrichment in front of the S_1 solid phase. By diffusion in the liquid, the excessive B atoms are transported in lateral direction, driving the propagation of the S_2 solid phase. The A atoms behave in an analogous manner. A characteristic

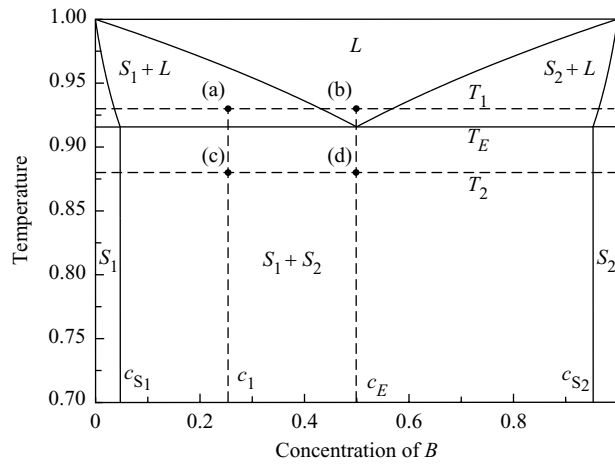


FIGURE 7-25 Symmetric phase diagram of a binary eutectic system with positions (a), (b), (c), and (d) related to the microstructures in Figure 7-26.

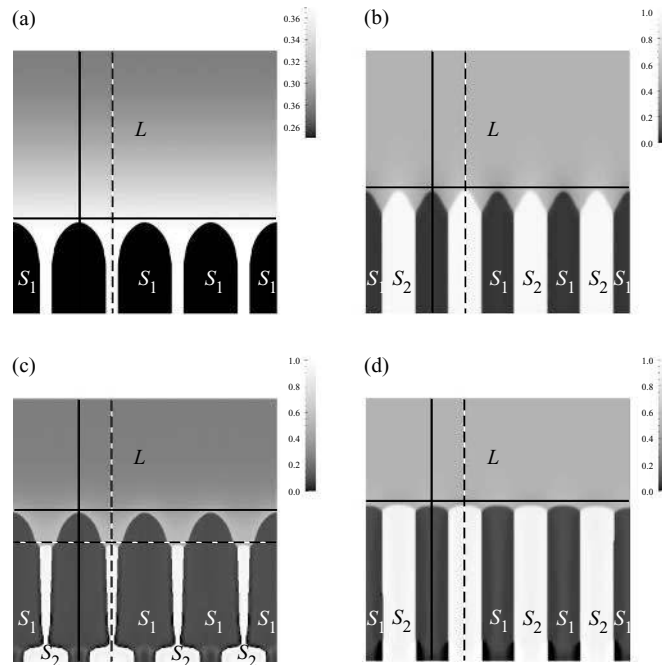


FIGURE 7-26 Simulated microstructures of a binary eutectic alloy, (a)–(d) correspond to different initial temperature and composition points in the phase diagram marked in Figure 7-25, (a) growth of primary S_1 cells at (c_1, T_1) , (b) melting of both solid phases S_1 and S_2 at (c_E, T_1) , (c) lamellar growth at an off-eutectic composition (c_1, T_2) and (d) lamellar growth at a eutectic composition (c_E, T_2) with $T_1 > T_E$, $T_2 < T_E$, and $c_1 < c_E$.

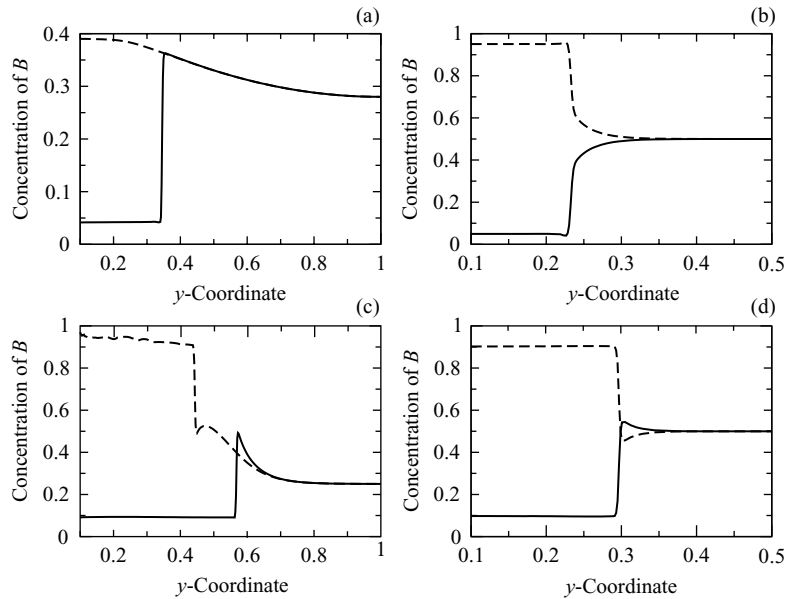


FIGURE 7-27 Concentration of B atoms along the y -axis at the two positions marked in Figure 7-26. The solid line belongs to a cut through the S_1 solid phase, whereas the dashed line represents a cut through the S_2 phase or the region between the S_1 cells.

periodic in space profile of alternating concentration enriched and depleted zones can be observed in front of the growing S_1 and S_2 lamellae in Figure 7-28(c) and (d). The dashed line in Figure 7-28(c) belongs to a horizontal cut through the S_1 lamellae just ahead of the retarded S_2 phase. The melting process at (c_E, T_1) establishes concentration profiles opposite to the eutectic growth case [Figures 7-27 and 7-28(b)]. The concentration of B atoms is reduced ahead of the dissolving S_1/S_2 front and vice versa for the A atoms.

For regular lamellar growth structures, a validation of the phase-field model can be achieved by comparing the volume phase fractions f_α and f_β of the simulated lamellae with the classical lever rule [KF92], Figure 7-29. The simulations were conducted at five different initial compositions in a region of the phase diagram between the two solidus lines and below the eutectic temperature. The resulting phase width W_1 and W_2 of the α and β lamellae at a point of stationary evolution (see the inlay of Figure 7-29) are in good agreement with the lever rule prediction.

To investigate the influence of the surface entropy on the growth of eutectic lamellae, a set of simulations has been carried out for surface entropies of isotropic type. The value of the entropy density $\gamma_{\alpha\beta}$ was varied in a range $0.3 \times 10^{-3} \leq \gamma_{\alpha\beta} \leq 2.7 \times 10^{-3}$ while keeping $\gamma = \gamma_{\alpha L} = \gamma_{\beta L} = 1 \times 10^{-3}$ constant. The measured angles at the triple junctions well compare with the Young's force balance law [Woo73] as shown in Figure 7-30.

For equal surface entropy densities, a 120° angle condition can be observed at the triple junctions. In the case $\gamma_{\alpha\beta} > \gamma_{\alpha L} + \gamma_{\beta L}$, wetting occurs at the α - β phase boundary, Figure 7-30. As a result of minimization of energy, the liquid phase penetrates the α - β interface. The α - β phase boundary rapidly disappears, and two new boundaries α - L and β - L of less total energy are formed. If the surface entropies are anisotropic, further effects due to shear forces at the boundaries influence the angle condition as expressed by the Herring torque terms [Woo73].

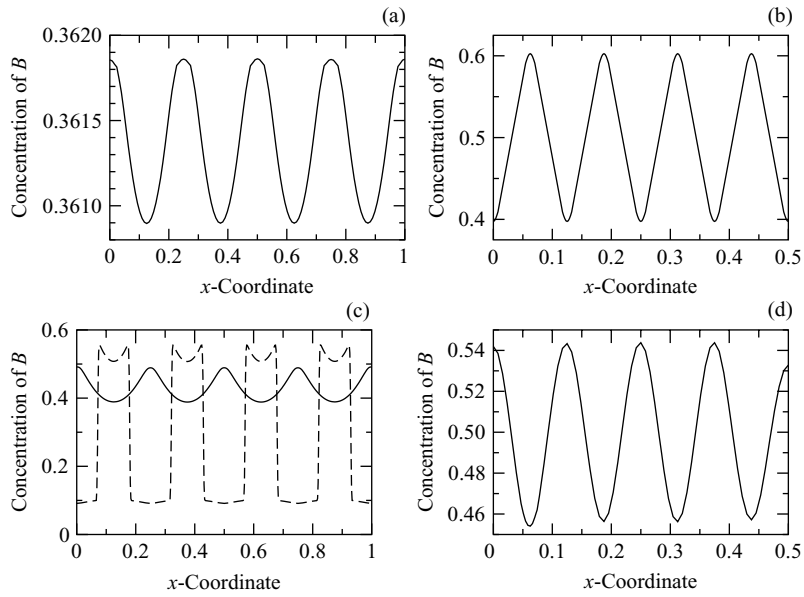


FIGURE 7-28 Concentration of B atoms along the x -axis ahead of the solid–liquid front (solid lines). The dashed line in (c) corresponds to a cut in front of the solid phase S_2 at the position marked in Figure 7-26.

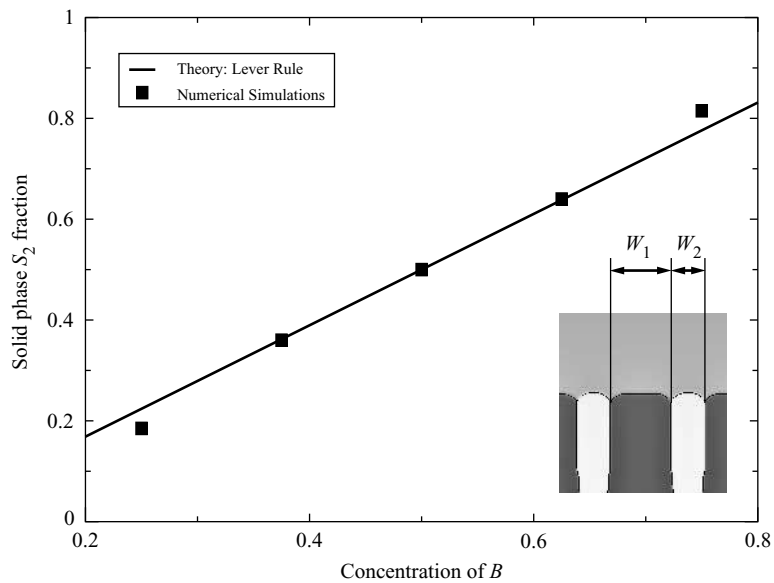


FIGURE 7-29 Simulated volume phase fractions of the two solid phases α and β for different melt compositions at an isothermal undercooling in comparison with the classical lever rule.

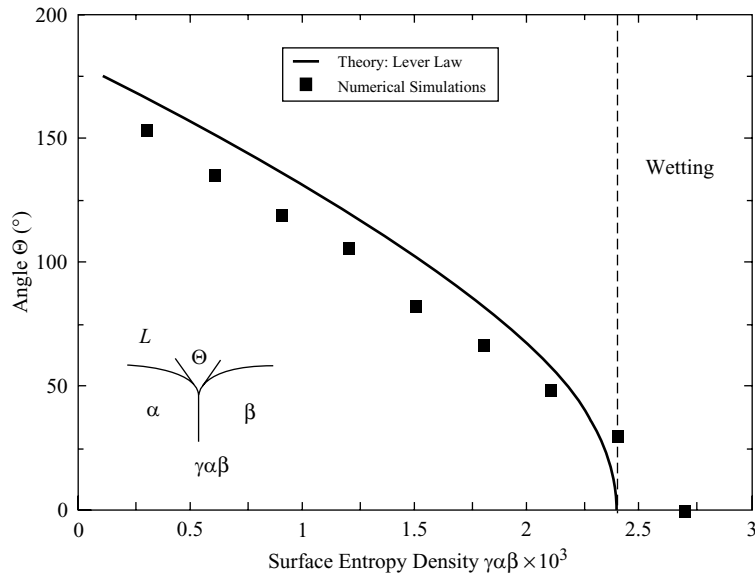


FIGURE 7-30 Angle Θ at the eutectic triple junction for different surface entropies $\gamma_{\alpha\beta}$. Measured values from the simulations in comparison with the classical Young's law. The vertical line marks the transition to the region where wetting occurs.

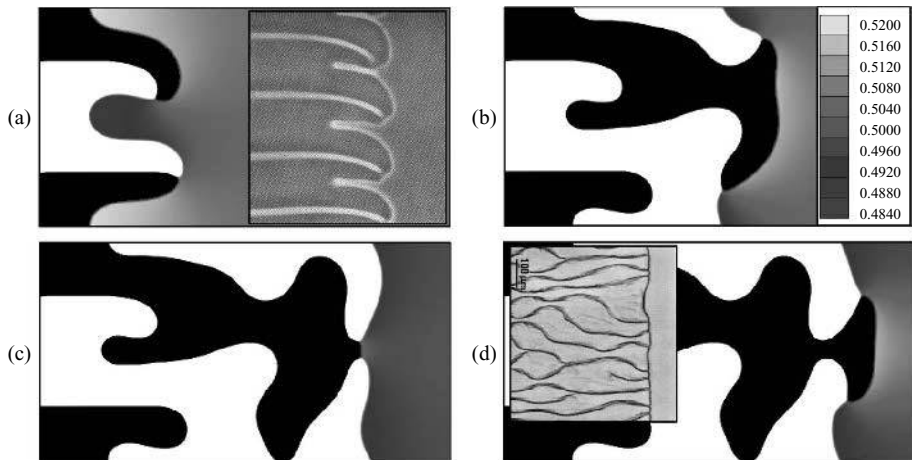


FIGURE 7-31 Simulation of an irregular eutectic structure. The black and white regions correspond to the α and β solid phase. The inlays in (a) and (d) correspond to experimental oscillatory microstructures observed by Jackson and Hunt [JH66] and by Ginibre et al. [GAF97].

For large initial spacings of the eutectic lamellae and for the implication of small amounts of noise at the solid–liquid front, the system is unable to develop a steady state growth shape and it evolves in a disordered manner, Figure 7-31. The initial lamellae quickly develop concave hollows which deepen. Subsequently, the two solid phases compete and alternately overgrow one another. During the evolution, a liquid droplet is engulfed by the solid at the solid–solid

interface; subsequently, it rapidly solidifies. Nucleation of the opposite phase within the concave portion of the interfaces stabilizes the lamellae and regular lamellar growth to continue at half the original lamellae spacing. This suggests that nucleation ahead of the eutectic front may provide a method of stabilizing the lamellae.

Morphological instabilities with a regular oscillatory structure of lamellar eutectics are reported in experiments with a transparent organic alloy and also in numerical studies by Karma and Sarkissian [KS96]. In three dimensions, an analogous type of oscillation can be observed for eutectic microstructure formations, Figure 7-32. Performing an alternating topological change, α solid rods are embedded in a β matrix followed by the opposite situation of β crystals embedded in an α matrix.

As a next example, we apply the phase-field model to simulations of solidification processes in ternary A–B–C alloys. Simulation results of ternary dendritic and ternary eutectic growth are exemplarily illustrated. In particular, the ternary $\text{Ni}_{60}\text{Cu}_{40-x}\text{Cr}_x$ alloy system is considered as a prototype system to investigate the influence of interplaying solute fields on the interface stability, on the growth velocity, and on the characteristic type of morphology. The Ni–Cu–Cr system serves as an extension of the binary Ni–Cu system which has been explored by phase-field modeling (e.g., ref. [WB94]) and by molecular dynamics simulations in several papers (e.g., refs. [HSAF99, HAK01]). Hence, physical parameters of Ni–Cu are relatively well established.

A series of numerical computations for different alloy compositions varying from $\text{Ni}_{60}\text{Cu}_{28}\text{Cr}_{12}$ to $\text{Ni}_{60}\text{Cu}_{12}\text{Cr}_{28}$ has been carried out in Figure 7-33. The concentration of Ni was kept constant at 60 at.%, and the initial undercooling was fixed at 20 K measured from the equilibrium liquidus line in the phase diagram at a given composition of the melt. A morphological transition from dendritic to globular growth occurs at a melt composition of about $\text{Ni}_{60}\text{Cu}_{20}\text{Cr}_{20}$.

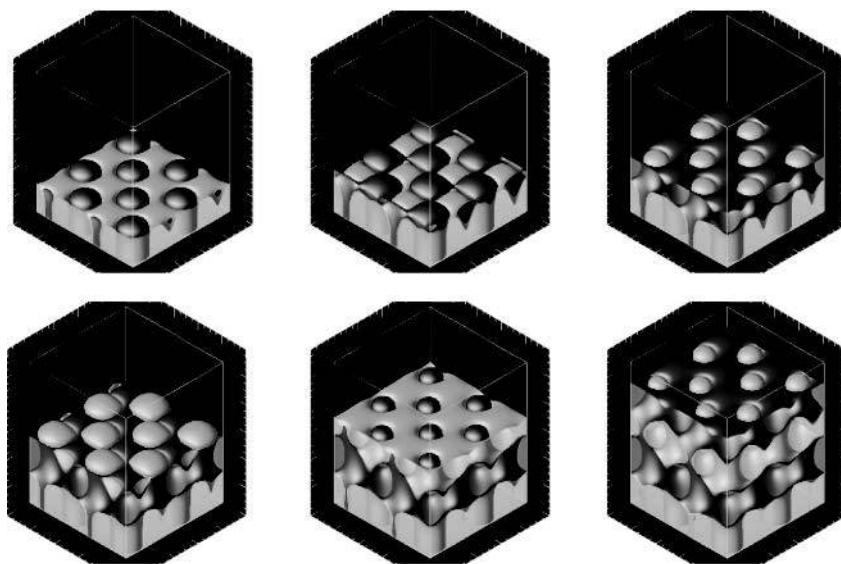


FIGURE 7-32 Topological change of the microstructure with α rods embedded in a β matrix phase and vice versa. The structure formation results from regular 3D oscillations along the solid–solid interface.

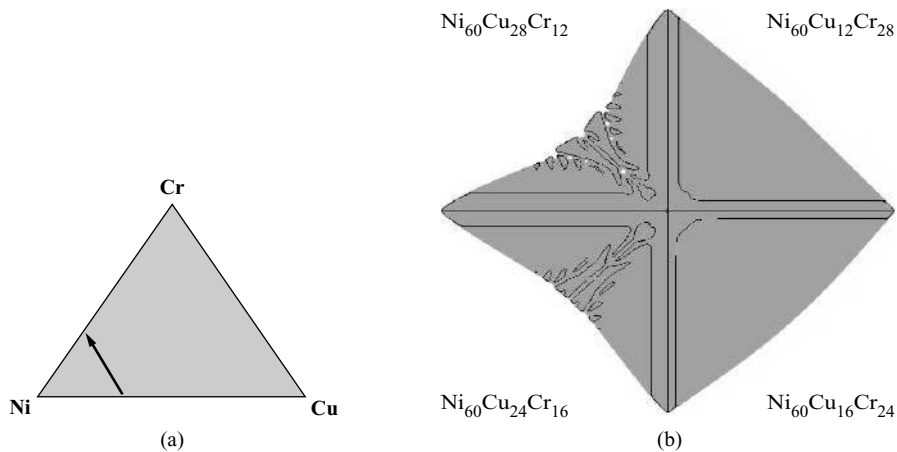


FIGURE 7-33 (a) Schematic drawing of an isothermal cut through the ternary Ni–Cu–Cr phase diagram. The arrow marks the path where the simulations of the morphological changes were performed. (b) Dendritic to globular morphological transition for different alloy compositions. The atomic percents of Cu and Cr are exchanged while keeping Ni fixed at 60at.%. The gray region corresponds to the solid phase, and the solid lines represent the isolines of average concentration of Ni in the solid phase.

The left side of Figure 7-33(b) shows the dendritic morphologies observed for Cr concentrations less than 20at.%. The right side of Figure 7-33(b) displays globular morphologies for Cr concentrations crossing this threshold. The velocity of the dendritic–globular tip increases linearly from 1.19 cm/s to 3.24 cm/s with increasing the concentration of Cr. The morphological transition is related to the transition from a two-phase region (above the solidus line) to a one-phase region (below the solidus line) in the phase diagram.

In a ternary eutectic system, we considered two regions within a completely symmetric model phase diagram, namely, (i) the region of four phase equilibrium at the ternary eutectic composition and temperature and (ii) a region where one component has a minor contribution of the amount of a ternary impurity. At the ternary composition $c = 0.3$, three solid phases α , β , and γ grow into an undercooled melt L with equal volume fractions by a ternary eutectic reaction $L \rightarrow \alpha + \beta + \gamma$. Different permutations of lamellar structures $\alpha\beta\gamma\dots$ and $\alpha\beta\alpha\gamma\dots$ are possible. Phase-field simulations can be used to investigate which phase sequence is favored to grow at certain solidification conditions. An example for an $\alpha\beta\gamma\dots$ configuration is displayed in Figure 7-34(a) showing in addition the concentration of one of the components ahead of the growing eutectic front. In three dimensions, a regular hexagonal shape of the three solid phases occurs for isotropic growing phases at three different time steps [Figure 7-34(b)]. The hexagonal symmetry breaks if anisotropy is included and if, hence, strong facets form in preferred growth directions.

In a eutectic phase system with ternary impurity, it can be observed that the impurity is pushed ahead of the solidifying lamellae and builds up. For the simulation in Figure 7-35, we have set an initial composition vector of $(c_A, c_B, c_C) = (0.47, 0.47, 0.06)$ so that c_C is the concentration component of minor amount. The main components c_A and c_B are incorporated in the growing α – β solid front whereas the impurity is rejected by both growing solid phases. To further observe the effect of eutectic cell/colony formation, computations in larger domains including noise as well as nucleation have to be performed.

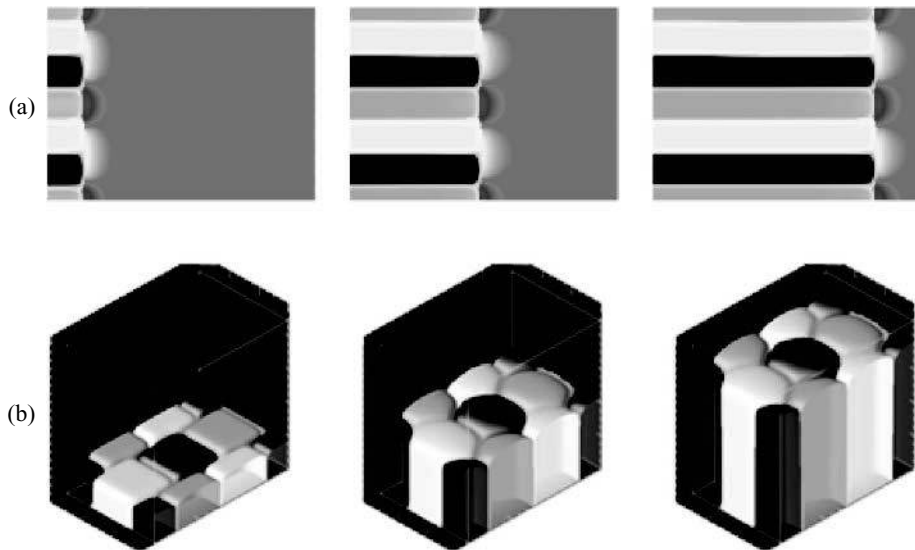


FIGURE 7-34 (a) Phase transformations in a ternary eutectic system with three solid phases growing into an undercooled melt. The concentration of component A is shown for different time steps ahead of a regular $\alpha\beta\gamma$ configuration. (b) Ternary eutectic growth in three dimensions forming a steadily propagating hexagonal structure.

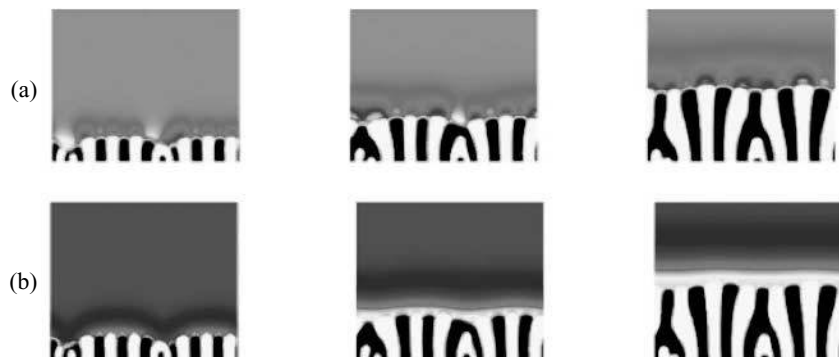


FIGURE 7-35 (a) Concentration profile of the main component c_A in the melt. (b) The ternary impurity c_C is pushed ahead of the growing eutectic front so that the concentration enriched zones of component c_C can be observed at the solid-liquid interface.

Another important field of application for phase-field modeling is the computation of grain structure evolution and anisotropic curvature flow in a polycrystalline material. In this case, the phase-field variables $\phi_\alpha, \alpha = 1, \dots, N$ represent the state of crystals with different crystallographic orientations. Figure 7-36 shows the effect of grain boundary motion on the growth selection in comparison with an experimental microstructure. As initial configuration, a distribution of small grains was posed along a thin layer at the upper wall of the simulation box. The grains started to grow toward the bottom of the domain. Certain grains with their crystal orientation in the direction of the shear movement of the lower wall grew faster than the neighboring grains, which finally ceased to grow.

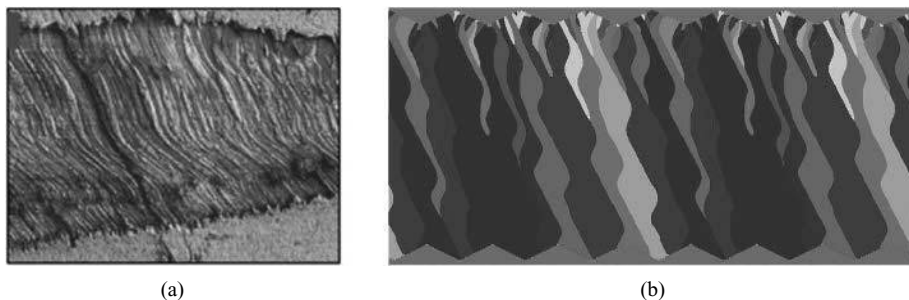


FIGURE 7-36 Grain selection process as a result of grain boundary motion of differently oriented crystals, (a) experimental microstructure observed in geological material, [Hil] and, (b) phase-field simulation. Grains with their growth direction in alignment with the shear movement of the lower wall dominate the structure formation.

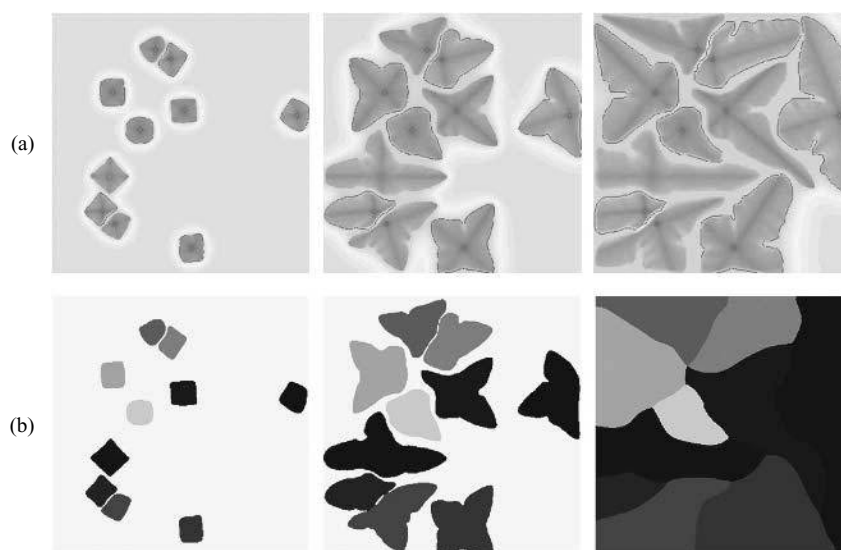


FIGURE 7-37 Dendritic growth of 10 Ni-Cu nuclei with different crystal orientation; (a) illustration of the Ni concentration in the crystallized solid dendrites and in the surrounding melt, (b) view of the sharp crystal boundaries showing the different crystal orientations in gray shades.

The final example in Figure 7-37 shows a distribution of differently oriented nuclei growing into an isothermally undercooled Ni–Cu melt. To reach the state of complete crystallization, the system is quenched. The individual dendrites match and form grain boundaries of a polycrystalline grain structure. After the solidification is finalized, the grain boundaries continue to migrate as a result of curvature minimization.

The phase-field model recovers the generic features of grain growth such as grain boundary motion, crystalline curvature flow, the force balance known as Young’s law at triple junctions, the (in)stability of quadruple junctions, wetting phenomena, and the symmetry behavior of neighboring triple junctions in microstructures of polycrystalline thin films.

7.5 Acknowledgments

The author of this chapter thanks her co-workers Frank Wendler and Denis Danilov for providing images of the simulation results.

Bibliography

- [Alm99] R. F. Almgren. *SIAM J. Appl. Math.*, 59:2086–2107, 1999.
- [AMW98] D. M. Anderson, G. B. McFadden, and A. A. Wheeler. *Annu. Rev. Fluid Mech.*, 30:139–165, 1998.
- [BGJ82] J. Q. Broughton, G. H. Gilmer, and K. A. Jackson. Crystallization rates of a Lennard–Jones liquid. *Phys. Rev. Lett.*, 49:1496–1500, 1982.
- [BJWH93] M. Barth, F. Joo, B. Wei, and D. M. Herlach. Measurement of the enthalpy and specific heat of undercooled nickel and iron melts. *J. Non-Crystalline Solids*, 156–158:398–401, 1993.
- [BK99] W. Bangerth and G. Kanschat. *Concepts for Object-Oriented Finite Element Software—The deal.II Library, Preprint 99-43 (SFB 359)*. IWR, Heidelberg, 1999.
- [Bre90] E. A. Brener. Effects of surface energy and kinetics on the growth of needle-like dendrites. *Journal of Crystal Growth*, 99:165–170, 1990.
- [Cag89] G. Caginalp. *Phys. Rev. A*, 39:5887, 1989.
- [CF] G. Caginalp and P. C. Fife. *Phys. Rev. B*, 33:7792, 1986.
- [CH74] J. W. Cahn and D. W. Hoffman. *Acta Metall.*, 22:1205, 1974.
- [Cha77] B. Chalmers. *Principles of Solidification*. Krieger, Melbourne, FL, 1977.
- [Che95] Long-Qing Chen. *Scr. Metall. Mater.*, 32:115, 1995.
- [CT82] S. R. Coriell and D. Turnbull. Relative roles of heat transport and interface rearrangement rates in the rapid growth of crystals in undercooled melts. *Acta Metall.*, 30:2135–2139, 1982.
- [CX93] G. Caginalp and W. Xie. *Phys. Rev. E*, 48:1897, 1993.
- [CY94] Long-Qing Chen and Wei Young. *Phys. Rev. B*, 50:15752, 1994.
- [Eck92] K. Eckler. *Dendritisches Wachstum in unterkühlten Metallschmelzen*. Ph.D. thesis, Ruhr-Universität Bochum, 1992.
- [EJHS00] B. Engquist, L. Johnsson, M. Hammill, and F. Short, eds. *Computation of Dendrites on Parallel Distributed Memory Architectures*, Vol. 13. Springer-Verlag, Berlin, 2000.
- [FPG+04] O. Funke, G. Phanikumar, P. K. Galenko, M. Kolbe, and D. M. Herlach. Dendrite growth velocities during solidification of levitated undercooled nickel melts. *J. Appl. Phys.*, 2004.
- [GAF97] M. Ginibre, S. Akamatsu, and G. Faivre. *Phys. Rev. E*, 56:780–796, 1997.
- [GN00] H. Garcke and B. Nestler. *Math. Models and Methods in Applied Sciences*, 10:895–921, 2000.
- [GNS98] H. Garcke, B. Nestler, and B. Stoth. *Physica D*, 115:87, 1998.
- [GNS99a] H. Garcke, B. Nestler, and B. Stoth. *SIAM J. Appl. Math.*, 60:295–315, 1999.
- [GNS99b] H. Garcke, B. Nestler, and B. Stoth. *J. Interfaces Free Boundary Problems*, 1:175–198, 1999.
- [GNS04] H. Garcke, B. Nestler, and B. Stinner. *SIAM J. Appl. Math.*, 64:775–799, 2004.
- [Haa94] P. Haasen. *Physikalische Metallkunde*. Springer-Verlag, Berlin, 3rd ed. 1994.
- [HAK01] J. J. Hoyt, M. Asta, and A. Karma. Method for computing the anisotropy of the solid–liquid interfacial free energy. *Phys. Rev. Lett.*, 86:5530–5533, 2001.
- [HAK03] J. J. Hoyt, M. Asta, and A. Karma. Atomistic and continuum modelling of dendritic solidification. *Mater. Sci. Eng. R*, 41:121–153, 2003.
- [HC72] D. W. Hoffman and J. W. Cahn. *Surf. Sci.*, 31:368, 1972.
- [Hil] C. Hilgers. *Vein Growth in Fractures—Experimental, Numerical and Real Rock Studies*. Shaker Verlag, Germany, 2000.
- [HSAF99] J. J. Hoyt, B. Sadigh, M. Asta, and S. M. Foiles. Kinetic phase field parameters for Cu–Ni system derived from atomistic computations. *Acta Mater.*, 47:3181–3187, 1999.
- [JH66] K. A. Jackson and J. D. Hunt. *Trans. Metall. Soc. AIME*, 236:1129, 1966.
- [Kar01] A. Karma. *Phys. Rev. Lett.*, 87:115701–1–115701–4, 2001.

- [KF92] W. Kurz and D. J. Fischer. *Fundamentals of Solidification*. Trans Tech Publications, Aedermannsdorf, Switzerland, 3rd ed. 1992.
- [KLP00] A. Karma, Y. H. Lee, and M. Plapp. *Phys. Rev. E*, 61:3996–4006, 2000.
- [Kob91] R. Kobayashi. *Bull. Jpn. Soc. Ind. Appl. Math.*, 1:22, 1991.
- [Kob93] R. Kobayashi. *Physica D*, 63:410, 1993.
- [Kob94] R. Kobayashi. *Exp. Math*, 3:60, 1994.
- [KR96] A. Karma and W.-J. Rappel. *Phys. Rev. E*, 53:3017–3020, 1996.
- [KR98] A. Karma and W.-J. Rappel. *Phys. Rev. E*, 57:4323–4349, 1998.
- [KS96] A. Karma and A. Sarkissian. *Met. Mater. Trans.*, 27:634, 1996.
- [KWC98] R. Kobayashi, J. A. Warren, and W.C. Carter. *Physica D*, 119:415, 1998.
- [Lan86] J. S. Langer. *Directions in Condensed Matter Physics*. World Scientific, Singapore, 1986.
- [LBT92] H. Löwen, J. Bechoefer, and L. S. Tuckerman. *Phys. Rev. A*, 45:2399, 1992.
- [Luc91] S. Luckhaus. *Eur. J. Appl. Math.*, 1:101–111, 1991.
- [LV83] S. Luckhaus and A. Visintin. *Manuscripta Math.*, 43:261–288, 1983.
- [MWA00] G. B. McFadden, A. A. Wheeler, and D. M. Anderson. *Physica D*, 144:154, 2000.
- [NW] B. Nestler and A. A. Wheeler. *Phys. Rev. E*, 57:2602–2609, 1998.
- [PF90] O. Penrose and P. C. Fife. *Physica D*, 43:44, 1990.
- [PGD98a] N. Provatas, N. Goldenfeld, and J. Dantzig. *Phys. Rev. Lett.*, 80:3308, 1998.
- [PGD98b] N. Provatas, N. Goldenfeld, and J. Dantzig. *Phys. Rev. Lett.*, 80:3308, 1998.
- [PGD99] N. Provatas, N. Goldenfeld, and J. Dantzig. *J. Comp. Phys.*, 148:265–290, 1999.
- [PK00] M. Plapp and A. Karma. *Phys. Rev. Lett.*, 84:1740–1743, 2000.
- [Sch96] A. Schmidt. *J. Comp. Phys.*, 125:293–312, 1996.
- [Sch98] M. Schwarz. *Kornfeinung durch Fragmentierung von Dendriten*. Ph.D. thesis, Ruhr-Universität Bochum, 1998.
- [Son95] H. M. Soner. *Arch. Rat. Mech. Anal.*, 131:139–197, 1995.
- [SPN+96] I. Steinbach, F. Pezzolla, B. Nestler, J. Rezend, M. Seesselberg, and G. J. Schmitz. *Physica D*, 94:135, 1996.
- [Sto96] B. Stoth. *Eur. J. Appl. Math.*, 7:603–634, 1996.
- [STSS97] M. Seesselberg, J. Tiaden, G. J. Schmitz, and I. Steinbach. In J. Beek and H. Jones (B. B. R. Sheffield), eds., *Proceedings of the Fourth Decennial International Conference on Solidification Processing, Sheffield, UK*, pp. 426–430, 1997.
- [TA98] T. Tönhardt and G. Amberg. *J. Cryst. Growth*, 194:406, 1998.
- [Vis96] A. Visintin. 1996.
- [WB94] J. A. Warren and W. J. Boettinger. *Acta Metall. Mater.*, 43:689, 1994.
- [WBM92] A. A. Wheeler, W. J. Boettinger, and G. B. McFadden. *Phys. Rev. A*, 45:7424, 1992.
- [WBM93] A. A. Wheeler, W. J. Boettinger, and G. B. McFadden. *Phys. Rev. E*, 47:1893, 1993.
- [WM96] A. A. Wheeler and G. B. McFadden. *Eur. J. Appl. Math.*, 7:369, 1996.
- [WM97] A. A. Wheeler and G. B. McFadden. *Proc. R. Soc. London, Ser. A*, 453:1611, 1997.
- [WMS93] A. A. Wheeler, B. T. Murray, and R. J. Schaefer. *Physica D*, 66:243, 1993.
- [Woo73] D. P. Woodruff. *The Solid–Liquid Interface*. Cambridge, 1973.
- [WS96] S.-L. Wang and R. F. Sekerka. *J. Comp. Phys.*, 127:110, 1996.
- [WSW+93] S.-L. Wang, R. F. Sekerka, A. A. Wheeler, S. R. Coriell, B. T. Murray, R. J. Braun, and G. B. McFadden. *Physica D*, 69:189, 1993.

8 Introduction to Discrete Dislocation Statics and Dynamics

—*Dierk Raabe*

8.1 Basics of Discrete Plasticity Models

This chapter deals with the simulation of plasticity of metals at the microscopic and mesoscopic scale using space- and time-discretized dislocation statics and dynamics. The complexity of discrete dislocation models is due to the fact that the mechanical interaction of ensembles of such defects is of an elastic nature and, therefore, involves long-range interactions.

Space-discretized dislocation simulations idealize dislocations outside the dislocation cores (few atomic positions in the center of the dislocation) as linear defects which are embedded within an otherwise homogeneous, isotropic or anisotropic, linear elastic medium. Therefore, the elastic material outside of the dislocation cores can be described in terms of the Hooke law of elasticity. This applies for both straight infinite dislocations (2D discretization) and dislocation segments (3D discretization).

The simulation work in this field can be grouped into simulations which describe the dislocations in two dimensions and in three dimensions. 2D calculations are conducted either with dislocations that cannot leave their actual glide plane (view into the glide plane) [FM66, BKS73, Rön87, Moh96], or with nonflexible infinite straight dislocations which may leave their glide plane but cannot bow out (view along the line vector of the dislocation) [LK87, GA89, GSSL89, Amo90, GLSL90, GH92, LBN93, WL95]. 3D simulations which decompose each dislocation into a “spaghetti-type” sequence of piecewise straight segments with a scaling length much below the length of the complete dislocation line are independent of such geometrical constraints [KCC⁺92, DHZ92, DC92, DK94, RHZ94, Raa96a, Raa96b, Dev96, FGC96, Raa98].

The motion of the dislocations or of the dislocation segments in their respective glide planes are usually described by assuming simple phenomenological viscous flow laws. “Viscous flow” means that the dislocation is in an overdamped state of motion so that its velocity is linearly proportional to the local net force which acts in the dislocation glide plane. Viscous motion phenomenologically describes strain rate sensitive flow.

A more detailed formulation of the dynamics of dislocations can be obtained by solving Newton’s second law of motion for each dislocation or dislocation segment, respectively. This formulation which takes into account the effective mass of the dislocation (which is a measure for the reluctance of the dislocation against acceleration) is of relevance only for very small and

for very large dislocation velocities. The solution of the temporal evolution of the dislocation positions is as a rule obtained by simple finite difference algorithms.

It is not the aim of this chapter to provide an exhaustive review of the large number of analytical statistical, phenomenological models that dominate the field of mesoscopic non-space-discretized materials modeling, but to concentrate on those simulations that are discrete in both space and time, and explicitly incorporate the properties of individual lattice defects in a continuum formulation. The philosophy behind this is twofold. First, the various classical phenomenological mesoscale modeling approaches, which are discrete in time but not in space, have already been the subject of numerous thorough studies in the past, particularly in the fields of crystal plasticity, recrystallization phenomena, and phase transformation [Koc66, Arg75, KAA75, Mug80, MK81, Mug83, EM84, PA84, EK86, GA87, Koc87, EM91]. These models usually provide statistical rather than discrete solutions and can often be solved without employing time-consuming numerical methods. This is why they often serve as a physical basis for deriving phenomenological constitutive equations that can be incorporated in advanced larger-scale finite element, self-consistent, or Taylor-type simulations [Arg75, GZ86, ABH⁺87, Koc87, NNH93, KK96]. However, since such constitutive descriptions only provide an *averaged* picture of the material response to changes in the external constraints, they are confined to statistical predictions and do not mimic details of the microstructural evolution. Hence, they are beyond the scope of this chapter. Second, physically sound micro- and mesoscale material models that are discrete in both space and time must incorporate the statics and kinetics of *individual* lattice defects. This makes them superior to the more descriptive statistical models in that they allow simulations that are more precise in their microscopical predictions due to the smaller number of phenomenological assumptions involved. An overview of statistical analytical dislocation models can be found in the more recent overview volume of Raabe *et al.* [RBC04].

8.2 Linear Elasticity Theory for Plasticity

8.2.1 Introduction

This section reviews the theoretical framework of linear continuum elasticity theory as required for the formulation of basic dislocation mechanics. The mechanical interaction between different dislocations (2D case) or different dislocation segments (3D case) is transmitted by the constituent elements (atoms) of the material. In this approach the material is described as an isotropic or anisotropic linear elastic unbounded homogeneous continuum in which the dislocations are embedded as elementary carriers of displacement and stress. This statement already implies some essentials associated with the mathematical treatment of dislocations, namely, that they are outside their cores simulated as line defects in the framework of linear elasticity. Large strains occurring close to the dislocation cores are naturally excluded from the elastic treatment. For this purpose an inner cutoff radius in the order of the magnitude of the Burgers vector is used.

The dislocations are generally treated as stationary defects, that is, their displacement field does not depend on time. This implies that for all derivations the time-independent Green's function may be used.

While in the pioneering studies [DC92, Kub93, Raa96a] the field equations for the isotropic elastic case were used for 3D simulations, this chapter presents the general anisotropic field approach [Raa96b, Raa98]. For this reason the following sections recapitulate the elementary concepts of isotropic and anisotropic linear elastic theory. On this basis the field equations for both infinite dislocations (2D) and finite dislocation segments (3D) will be developed.

In what follows, the notation x_1, x_2, x_3 will be used in place of x, y, z for the Cartesian coordinate system. This notation has the particular advantage that in combination with Einstein's summation convention it permits general results to be expressed and manipulated in a concise manner. The summation convention states that any term in which the same Latin suffix occurs twice stands for the sum of all the terms obtained by giving this suffix each of its possible values. For instance, the trace of the strain tensor can be written as ε_{ii} and interpreted as

$$\varepsilon_{ii} \equiv \sum_{i=1}^3 \varepsilon_{ii} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} \quad (8.1)$$

For the trace of the displacement gradient tensor the same applies:

$$\frac{\partial u_i}{\partial x_i} \equiv \sum_{i=1}^3 \frac{\partial u_i}{\partial x_i} = \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} \quad (8.2)$$

On the other hand, certain results are more conveniently expressed in vector notation (bold symbols, e.g., \mathbf{u}) or as components (suffixes 1, 2, 3, e.g., u_1, u_2, u_3). In lengthy terms, partial spatial derivatives will be expressed by the abbreviation $v_{1,2}$ instead of $\partial v_1 / \partial x_2$.

8.2.2 Fundamentals of Elasticity Theory

The Displacement Field

In a solid unstrained body the position of each infinitesimal volume element¹ can be described by three Cartesian coordinates, x_1, x_2 , and x_3 . In a strained condition the position of the volume element considered will shift to a new site described by $x_1 + u_1, x_2 + u_2, x_3 + u_3$, where the triple u_1, u_2, u_3 is referred to as displacement parallel to the x_1 -, x_2 -, and x_3 -axis, respectively.

The displacement field corresponds to the values of u_1, u_2 , and u_3 at every coordinate x_1, x_2, x_3 within the material. In general, the displacement is a vector field which depends on all three spatial variables. It maps every point of the body from its position in the undeformed to its position in the deformed state. For instance, translations represent trivial examples of displacement, namely, that of a rigid-body motion where u_1, u_2 , and u_3 are constants.

The Strain Field

Let the corners of a volume element, which is much larger than the atomic volume, be given by the coordinates $(x_1, x_2, x_3), (x_1 + \Delta x_1, x_2, x_3), (x_1, x_2 + \Delta x_2, x_3)$ and so on. During straining, the displacement of the corner with the coordinates (x_1, x_2, x_3) will amount to (u_1, u_2, u_3) . Since the displacement is a function of space it can be different for each corner. Using a Taylor expansion the displacements can be described by

$$\begin{pmatrix} u_1 + \frac{\partial u_1}{\partial x_1} \delta x_1 + \frac{\partial u_1}{\partial x_2} \delta x_2 + \frac{\partial u_1}{\partial x_3} \delta x_3 \\ u_2 + \frac{\partial u_2}{\partial x_1} \delta x_1 + \frac{\partial u_2}{\partial x_2} \delta x_2 + \frac{\partial u_2}{\partial x_3} \delta x_3 \\ u_3 + \frac{\partial u_3}{\partial x_1} \delta x_1 + \frac{\partial u_3}{\partial x_2} \delta x_2 + \frac{\partial u_3}{\partial x_3} \delta x_3 \end{pmatrix} \quad (8.3)$$

¹ Since for small displacements the elastic bulk modulus is proportional to the spatial derivative of the interatomic forces, any cluster of lattice atoms can be chosen as an infinitesimal volume element.

Using concise suffix notation, equation (8.3) can be rewritten

$$(u_1 + u_{1,j} \delta x_j, \quad u_2 + u_{2,j} \delta x_j, \quad u_3 + u_{3,j} \delta x_j) \quad (8.4)$$

where the summation convention is implied. The abbreviation $u_{1,2}$ refers to the spatial derivative $\partial u_1/\partial x_2$. These partial derivatives represent the components of the displacement gradient tensor $\partial u_i/\partial x_j = u_{i,j}$. In linear elasticity theory only situations in which the derivatives $\partial u_i/\partial x_j$ are small compared with 1 are treated. If the extension of the considered volume element $\Delta x_1, \Delta x_2, \Delta x_3$ is sufficiently small, the displacement described by equation (8.3) can be written

$$\begin{pmatrix} u_1 + \frac{\partial u_1}{\partial x_1} \Delta x_1 + \frac{\partial u_1}{\partial x_2} \Delta x_2 + \frac{\partial u_1}{\partial x_3} \Delta x_3 \\ u_2 + \frac{\partial u_2}{\partial x_1} \Delta x_1 + \frac{\partial u_2}{\partial x_2} \Delta x_2 + \frac{\partial u_2}{\partial x_3} \Delta x_3 \\ u_3 + \frac{\partial u_3}{\partial x_1} \Delta x_1 + \frac{\partial u_3}{\partial x_2} \Delta x_2 + \frac{\partial u_3}{\partial x_3} \Delta x_3 \end{pmatrix} \quad (8.5)$$

Using suffix notation equation (8.5) is found to be

$$(u_1 + u_{1,j} \Delta x_j, \quad u_2 + u_{2,j} \Delta x_j, \quad u_3 + u_{3,j} \Delta x_j) \quad (8.6)$$

For the corner of the volume element with the coordinates $(x_1, x_2 + \Delta x_2, x_3 + \Delta x_3)$ equation (8.5) reduces to

$$\left(u_1 + \frac{\partial u_1}{\partial x_2} \Delta x_2 + \frac{\partial u_1}{\partial x_3} \Delta x_3, \quad u_2 + \frac{\partial u_2}{\partial x_2} \Delta x_2 + \frac{\partial u_2}{\partial x_3} \Delta x_3, \right. \\ \left. u_3 + \frac{\partial u_3}{\partial x_2} \Delta x_2 + \frac{\partial u_3}{\partial x_3} \Delta x_3 \right) \quad (8.7)$$

Similar displacement expressions can be obtained for the other corners of the volume element being considered.

For situations where all of the derivatives except those denoted by $\partial u_1/\partial x_1$, $\partial u_2/\partial x_2$, and $\partial u_3/\partial x_3$ are equal to zero, it is straightforward to see that a rectangular volume element preserves its shape. In such a case the considered portion of material merely undergoes positive or negative elongation parallel to its edges. For the x_1 -direction the elongation amounts to $(\partial u_1/\partial x_1) \Delta x_1$. Hence, the elongation per unit length amounts to $(\partial u_1/\partial x_1) \cdot (\Delta x_1)/(\Delta x_1) = \partial u_1/\partial x_1$. This expression is referred to as strain in the x_1 -direction and is indicated by ε_{11} . Positive values are defined as tensile strains and negative ones as compressive strains. The sum of these strains parallel to x_1 , x_2 , and x_3 defines the dilatation, which equals the change in volume per unit volume associated with a given strain field, that is, $\varepsilon_{ii} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} = \text{div } \mathbf{u}$, where div is the operator $\partial/\partial x_1 + \partial/\partial x_2 + \partial/\partial x_3$. In case of a nonzero dilatation the strain components describe the change in both shape and size. The situation is different when each of the derivatives denoted by $\partial u_1/\partial x_1$, $\partial u_2/\partial x_2$, $\partial u_3/\partial x_3$ is zero, but the others are not. In such cases the considered initial rectangular volume element is no longer preserved but can both rotate and assume a rhombic shape. A single component of the displacement gradient tensor, for instance, $\partial u_2/\partial x_1$, denotes the angle by which a line originally in the x_1 -direction rotates towards the x_2 -axis during deformation. However, the rotation of an arbitrary boundary line of a small volume element does not necessarily imply that the material is deformed. One could rotate the boundary line simply by rotating the other boundaries accordingly. Such an

operation would leave the body undeformed and is therefore referred to as rigid-body rotation. However, if different boundary lines rotate by different angles the volume element undergoes both deformation and rigid-body rotation. By subtracting the rotation components from the displacement gradients, one obtains the elements which describe the shape changes. These components are denoted as shear strains $\frac{1}{2}(\partial u_1/\partial x_2 + \partial u_2/\partial x_1)$, $\frac{1}{2}(\partial u_1/\partial x_3 + \partial u_3/\partial x_1)$, and $\frac{1}{2}(\partial u_2/\partial x_3 + \partial u_3/\partial x_2)$. They describe half the angular change between mutually orthogonal lines that were initially parallel to each other. In engineering terms one often encounters the so-called technical strain γ_{ij} , which is defined by $\gamma_{ij} = 2\varepsilon_{ij}$. However, this quantity does not represent a tensor component and will not be used in what follows.

From the preceding, it becomes clear that a deformation state is entirely characterized by the displacement vector field u_i . However, this quantity is an inconvenient representation of deformation since it does not naturally separate shape changes from rigid-body rotations or volume changes. For this purpose the displacement gradient tensor $u_{i,j}$, which in the general case still contains the strain tensor ε_{ij} (tensile/compressive and shear components) and the rigid-body rotation ω_{ij} , seems more appropriate if adequately dismantled. Simple geometrical considerations show that the former portion corresponds to the symmetric part of the displacement gradient tensor and the latter one to its antisymmetric (skew symmetric) part:

$$\begin{pmatrix} u_{1,1} & u_{1,2} & u_{1,3} \\ u_{2,1} & u_{2,2} & u_{2,3} \\ u_{3,1} & u_{3,2} & u_{3,3} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 2u_{1,1} & u_{1,2} + u_{2,1} & u_{1,3} + u_{3,1} \\ u_{2,1} + u_{1,2} & 2u_{2,2} & u_{2,3} + u_{3,2} \\ u_{3,1} + u_{1,3} & u_{3,2} + u_{2,3} & 2u_{3,3} \end{pmatrix} \\ + \frac{1}{2} \begin{pmatrix} 0 & u_{1,2} - u_{2,1} & u_{1,3} - u_{3,1} \\ u_{2,1} - u_{1,2} & 0 & u_{2,3} - u_{3,2} \\ u_{3,1} - u_{1,3} & u_{3,2} - u_{2,3} & 0 \end{pmatrix} \quad (8.8)$$

Using suffix notation this can be concisely rewritten as

$$u_{i,j} = \varepsilon_{ij} + \omega_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (8.9)$$

From the linear decomposition of the displacement gradient tensor it is evident that the equivalences $\varepsilon_{ij} = \varepsilon_{ji}$ and $\omega_{ij} = -\omega_{ji}$ apply.

Since the trace elements of the antisymmetric part of the displacement gradient tensor are by definition equal to zero, only three independent components remain. These represent small positive rotations about the axes perpendicular to the displacements from which they are derived, that is, $\omega_{23} = \omega_{x_1}$ denotes a rotation about the x_1 -axis, $\omega_{13} = \omega_{x_2}$ about the x_2 -axis, and $\omega_{12} = \omega_{x_3}$ about the x_3 -axis. By using the totally antisymmetric Levi-Civita operator ϵ_{ijk} , the components of which are defined to be 1 if the suffixes are in cyclic order, -1 if they are in reverse cyclic order, and 0 if any two suffixes are the same, the three rotations can be compactly written as

$$\omega_{x_k} = \frac{1}{2} \left(\epsilon_{ijk} \frac{\partial u_i}{\partial x_j} \right) \quad (8.10)$$

Summarizing the three rotation components as a vector, one obtains

$$\boldsymbol{\omega} = \frac{1}{2} \text{curl } \mathbf{u} = \frac{1}{2} \nabla \times \mathbf{u} \quad (8.11)$$

where ∇ denotes the operator $\left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3}\right)$ and \times the vector product.

The Stress Field

The introduction of the traction vector serves as a starting point for deriving the stress tensor. The traction is defined by

$$\mathbf{T} = \lim_{\Delta A \rightarrow 0} \left(\frac{\Delta \mathbf{F}}{\Delta A} \right) = \frac{d\mathbf{F}}{dA} \quad (8.12)$$

where \mathbf{T} denotes the traction vector, A the area, and \mathbf{F} the externally imposed force vector. In suffix notation the traction can be written

$$T_i = \lim_{\Delta A \rightarrow 0} \left(\frac{\Delta F_i}{\Delta A} \right) = \frac{dF_i}{dA} \quad (8.13)$$

Since the traction vector depends on the inclination of the area element considered, it is pertinent to look for a more general form to describe the effect of external forces on the material. Such a description can be found by considering the traction vectors of three orthogonal sections, the unit normal vectors of which are denoted by n_1, n_2, n_3 :

$$T_j = \sigma_{ij} n_i \quad (8.14)$$

Thus, by definition of equation (8.14), the stress σ_{ij} is a tensor field which connects \mathbf{T} to \mathbf{n} at any point within the material. Equation (8.14) is referred to as the Cauchy stress formula.

The components σ_{ij} with $i = j$ give the respective force component along the positive x_j -axis acting through the area element having its normal along the same direction. They are referred to as normal stresses. The components σ_{ij} with $i \neq j$ give the corresponding two orthogonal force components acting in the same area element along the two respective positive x_j -axes, where $i \neq j$. They are referred to as shear stresses. Considering momentum equilibrium under static conditions one obtains $\sigma_{ij} = \sigma_{ji}$. By solving the eigenvalue problem

$$|\sigma_{ij} - \delta_{ij}\sigma| = \sigma^3 - I_1 \sigma^2 + I_2 \sigma - I_3 = 0 \quad (8.15)$$

where δ_{ij} is the Kronecker symbol, and I_1, I_2, I_3 the invariants of the stress state, one obtains the principal stresses $\sigma = \sigma_1, \sigma_2, \sigma_3$. The principal axes are the eigenvectors associated with this stress tensor. The invariants amount to

$$\begin{aligned} I_1 &= \sigma_{ii} = \sigma_{11} + \sigma_{22} + \sigma_{33} \\ &= \sigma_1 + \sigma_2 + \sigma_3 \\ I_2 &= \frac{1}{2} (\sigma_{ii}\sigma_{jj} - \sigma_{ij}\sigma_{ij}) = \sigma_{11}\sigma_{22} - \sigma_{12}^2 + \sigma_{22}\sigma_{33} - \sigma_{23}^2 + \sigma_{11}\sigma_{33} - \sigma_{13}^2 \\ &= \sigma_1 \sigma_2 + \sigma_1 \sigma_3 + \sigma_2 \sigma_3 = \begin{vmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{vmatrix} + \begin{vmatrix} \sigma_{22} & \sigma_{23} \\ \sigma_{23} & \sigma_{33} \end{vmatrix} + \begin{vmatrix} \sigma_{11} & \sigma_{13} \\ \sigma_{13} & \sigma_{33} \end{vmatrix} \\ I_3 &= \det(\sigma_{ij}) = \sigma_{11}\sigma_{22}\sigma_{33} + 2\sigma_{12}\sigma_{13}\sigma_{23} - \sigma_{11}\sigma_{23}^2 - \sigma_{22}\sigma_{13}^2 - \sigma_{33}\sigma_{12}^2 \\ &= \sigma_1 \sigma_2 \sigma_3 = \begin{vmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{vmatrix} \end{aligned} \quad (8.16)$$

Since conservative dislocation motion is practically unaffected by the hydrostatic stress, it is pertinent to dismantle a given static stress state into its deviatoric and its hydrostatic portions. The latter contribution is given by

$$\sigma^h = \frac{1}{3} \sigma_{ii} = \frac{1}{3} (\sigma_{11} + \sigma_{22} + \sigma_{33}) = \frac{1}{3} I_1 \quad (8.17)$$

The deviatoric stress tensor can then be written

$$\begin{aligned} \sigma_{ij}^d &= \sigma_{ij} - \delta_{ij} \sigma^h = \begin{pmatrix} \sigma_{11} - \sigma^h & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} - \sigma^h & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} - \sigma^h \end{pmatrix} = S_{ij} \\ &= \begin{pmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{pmatrix} = \begin{pmatrix} S_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & S_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & S_{33} \end{pmatrix} \end{aligned} \quad (8.18)$$

The operation of adding or subtracting hydrostatic contributions to or from the deviator corresponds to a shift of the Mohr circle parallel to the abscissa or to the generation of a closed stress surface which is a hypersurface in stress space. Since the first invariant of the stress deviator is equal to zero, the characteristic polynom reduces to

$$|\sigma_{ij} - \sigma^h - \delta_{ij} \sigma| = \sigma^3 + J_2 \sigma_d - J_3 = 0 \quad (8.19)$$

where J_2 and J_3 are the remaining two invariants of the deviator and $\sigma = S_1, S_2, S_3$ the corresponding principal stresses. The invariants of the stress deviator are

$$\begin{aligned} J_1 &= S_{ii} = 0 \\ J_2 &= \frac{1}{2} (S_{ij} S_{ij}) = \frac{1}{2} (S_{11}^2 + S_{22}^2 + S_{33}^2) + \sigma_{12}^2 + \sigma_{23}^2 + \sigma_{13}^2 \\ &= \frac{1}{2} (S_1^2 S_2^2 S_3^2) = \frac{1}{6} [(\sigma_{11} - \sigma_{22})^2 + (\sigma_{22} - \sigma_{33})^2 + (\sigma_{33} - \sigma_{11})^2] + \sigma_{12}^2 + \sigma_{23}^2 + \sigma_{13}^2 \\ J_3 &= \det (S_{ij}) = \frac{1}{3} (S_{ij} S_{jk} S_{ki}) = S_1 + S_2 + S_3 = \begin{vmatrix} S_{11} & S_{12} & S_{13} \\ S_{12} & S_{22} & S_{23} \\ S_{13} & S_{23} & S_{33} \end{vmatrix} = \begin{vmatrix} S_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & S_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & S_{33} \end{vmatrix} \end{aligned} \quad (8.20)$$

where S_{ij} are the components and $S_1, S_2,$ and S_3 the principal stresses of the deviator.

8.2.3 Equilibrium Equations

For deriving the differential equations of equilibrium one has to apply Newton's second law to a small rectangular volume element, $\delta x_1 \delta x_2 \delta x_3$, under externally imposed forces \mathbf{F} and body forces \mathbf{P} acting on it. A body force is one that acts directly on every particle of the body, rather than being applied by tractions at its boundaries and transmitted through the various particles by means of internal stresses. Common examples of body forces are forces due to gravity and centrifugal forces. It is assumed that the external forces may vary with position so that they can be dissimilar on opposite faces of the volume element. Using a Taylor expansion

for the variation of stress as a function of position and Newton's second law gives the dynamic equations of equilibrium:

$$\begin{aligned}
m \frac{\partial^2 u_1}{\partial t^2} &= \left(\sigma_{11} + \frac{\partial \sigma_{11}}{\partial x_1} \delta x_1 - \sigma_{11} \right) \delta x_2 \delta x_3 + \left(\sigma_{12} + \frac{\partial \sigma_{12}}{\partial x_2} \delta x_2 - \sigma_{12} \right) \delta x_1 \delta x_3 \\
&\quad + \left(\sigma_{13} + \frac{\partial \sigma_{13}}{\partial x_3} \delta x_3 - \sigma_{13} \right) \delta x_1 \delta x_2 + P_1 \delta x_1 \delta x_2 \delta x_3 \\
m \frac{\partial^2 u_2}{\partial t^2} &= \left(\sigma_{22} + \frac{\partial \sigma_{22}}{\partial x_2} \delta x_2 - \sigma_{22} \right) \delta x_1 \delta x_3 + \left(\sigma_{21} + \frac{\partial \sigma_{21}}{\partial x_1} \delta x_1 - \sigma_{21} \right) \delta x_2 \delta x_3 \\
&\quad + \left(\sigma_{23} + \frac{\partial \sigma_{23}}{\partial x_3} \delta x_3 - \sigma_{23} \right) \delta x_2 \delta x_1 + P_2 \delta x_1 \delta x_2 \delta x_3 \\
m \frac{\partial^2 u_3}{\partial t^2} &= \left(\sigma_{33} + \frac{\partial \sigma_{33}}{\partial x_3} \delta x_3 - \sigma_{33} \right) \delta x_1 \delta x_2 + \left(\sigma_{31} + \frac{\partial \sigma_{31}}{\partial x_1} \delta x_1 - \sigma_{31} \right) \delta x_3 \delta x_2 \\
&\quad + \left(\sigma_{32} + \frac{\partial \sigma_{32}}{\partial x_2} \delta x_2 - \sigma_{32} \right) \delta x_3 \delta x_1 + P_3 \delta x_1 \delta x_2 \delta x_3 \quad (8.21)
\end{aligned}$$

where \mathbf{u} is the displacement vector, t the time, \mathbf{P} the body force vector, $\partial^2 \mathbf{u} / \partial t^2$ the acceleration, and m the mass. Hence, dividing by $(\delta x_1 \delta x_2 \delta x_3)$ and proceeding to the limit where these infinitesimals tend to zero, one obtains for the static equilibrium

$$\begin{aligned}
\frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \sigma_{12}}{\partial x_2} + \frac{\partial \sigma_{13}}{\partial x_3} + P_1 &= 0 \\
\frac{\partial \sigma_{22}}{\partial x_2} + \frac{\partial \sigma_{21}}{\partial x_1} + \frac{\partial \sigma_{23}}{\partial x_3} + P_2 &= 0 \\
\frac{\partial \sigma_{33}}{\partial x_3} + \frac{\partial \sigma_{31}}{\partial x_1} + \frac{\partial \sigma_{32}}{\partial x_2} + P_3 &= 0
\end{aligned} \quad (8.22)$$

Using the Einstein convention, equation (8.22) can be written

$$\frac{\partial \sigma_{ij}}{\partial x_j} + P_i = \sigma_{ij,j} + P_i = 0 \quad (8.23)$$

Equation (8.23) represents the basic field equation to be satisfied for any stress field in a volume element in static equilibrium. In other words the divergence of the stress tensor $\text{div}(\sigma_{ij})$ must vanish in the absence of body forces. This equation is sometimes referred to as the condition for translational equilibrium, while the expression $\sigma_{ij} = \sigma_{ji}$ denotes rotational equilibrium.

8.2.4 Compatibility Equations

Strict compatibility is achieved when the strains can be expressed in terms of a single-valued, continuously differentiable displacement. This condition implies that any integration of strain (displacement gradient) around an infinitesimal closed loop is equal to zero and independent of the path so that gaps and overlaps are avoided. To put the matter formally, one may write

$$\oint \frac{\partial \mathbf{u}}{\partial S} dS = 0 \quad (8.24)$$

It should be mentioned that this condition is not fulfilled in the case of dislocated elastic media.

The definition of strain as expressed by equation (8.9) can be regarded as a set of six independent differential equations for the displacement components u_i . Although there exist only three independent displacement components, an arbitrary choice of the strains does not, in general, allow the strain–displacement relations expressed by equation (8.9) to be integrated. A single-valued, continuously differentiable displacement vector is only obtained when the strains satisfy the six independent compatibility conditions,

$$\epsilon_{pmk} \epsilon_{qnj} \epsilon_{kj, nm} = 0 \quad (8.25)$$

8.2.5 Hooke's Law—the Linear Relationship between Stress and Strain

For small imposed forces the time-independent reversible response of the material which is quantified by the displacement field can be linearly related to the force. The strains expressed by the symmetric part of the displacement gradient tensor then describe a linear elastic shape change which is proportional to the stress [equation (8.26)]. This relation is referred to as Hooke's law. Experience substantiates its validity for a broad class of materials and stress–strain regimes of practical relevance:

$$\epsilon_{ij} = S_{ijkl} \sigma_{kl} \quad \sigma_{ij} = C_{ijkl} \epsilon_{kl} \quad (8.26)$$

These expressions hold for the general anisotropic elastic case, where a component of the strain tensor (stress tensor) can depend on each of the stress components (strain components). The proportionality or elastic constants then consist of the four-rank tensors S_{ijkl} , or C_{ijkl} . The former quantities are referred to as elastic compliances and the latter ones as elastic stiffnesses. For uniaxial tensile deformation the stiffness represents the stress that is required to yield an elastic elongation of 100%, which of course is not realistic because of plastic deformation or material failure. The general form of Hooke's law accounts for the anisotropy imposed by the interatomic bond and the crystalline nature of matter.

The stiffnesses and compliances are related by

$$C_{ijmn} S_{mnpq} = \frac{1}{2} (\delta_{ip}\delta_{jq} + \delta_{iq}\delta_{jp}) \quad (8.27)$$

Exploiting the symmetries $\epsilon_{ij} = \epsilon_{ji}$ and $\sigma_{ij} = \sigma_{ji}$ allows one to reduce the number of independent elastic constants from 81 to 36.

$$\begin{aligned} C_{ijkl} &= C_{ijlk} & C_{ijkl} &= C_{jikl} \\ S_{ijkl} &= S_{ijlk} & S_{ijkl} &= S_{jikl} \end{aligned} \quad (8.28)$$

Since classical elasticity is based on the assumption of reversible displacements, the only work done is due to elastic deformation. This fact defines additional constraints on the elastic constants:

$$\sigma_{ij} = C_{ijkl} \epsilon_{kl} = \frac{\partial W}{\partial \epsilon_{ij}} \quad (8.29)$$

which implies

$$C_{ijkl} = \frac{\partial^2 W}{\partial \epsilon_{ij} \partial \epsilon_{kl}} \quad (8.30)$$

where W is the Helmholtz free energy density, which is a path-independent state function. From this condition it follows that

$$C_{ijkl} = C_{klij} \quad (8.31)$$

These additional relations make it possible to reduce the number of independent elastic constants from 36 to 21. For materials obeying Hooke's law these symmetry conditions allow one to express equations (8.21)–(8.23) for the dynamic case more conveniently as

$$C_{ijkl} u_{k,lj} + P_i = \rho \frac{\partial^2 u_i}{\partial t^2} \quad (8.32)$$

where ρ is the mass density, and for the static case as

$$C_{ijkl} u_{k,lj} + P_i = 0 \quad (8.33)$$

Accordingly, the stresses are given by

$$\sigma_{ij} = C_{ijkl} u_{k,l} \quad (8.34)$$

For simplifying the complicated tensor notation, which requires up to four indices, Voigt [Voi9] and [Voi10] suggested the so-called matrix notation. This notation is a transformation rule by which index pairs ij are mapped into a single index m according to the following scheme:

$$\begin{array}{cccccccccc} ij & 11 & 22 & 33 & 23 & 32 & 13 & 31 & 12 & 21 \\ m & 1 & 2 & 3 & 4 & 4 & 5 & 5 & 6 & 6 \end{array} \quad (8.35)$$

By employing this transformation convention one can define the symmetric 6×6 matrices C'_{mn} and S'_{mn} . However, the following rules must be considered.

$$\begin{aligned} C'_{mn} &= C_{ijkl} \text{ for } 1 \leq m, n \leq 6 \\ S'_{mn} &= \begin{cases} S_{ijkl} & : \text{ if both } m \text{ and } n = 1, 2, 3 \\ 2 S_{ijkl} & : \text{ if either } m \text{ or } n \text{ but not both } = 1, 2, 3 \\ 4 S_{ijkl} & : \text{ if both } m \text{ and } n = 4, 5, 6 \end{cases} \end{aligned} \quad (8.36)$$

When using the matrix notation C_{mn} and S_{mn} it must be considered that for coordinate transformation or invariant determination, it is more useful to use the stiffnesses and compliances in their original tensorial form C_{ijkl} and S_{ijkl} . However, for conducting matrix inversions it is easier to use the notation suggested as equation (8.35) [Ste73, BBS79a]. For cubic crystal symmetry the reference coordinate system is chosen to coincide with the crystal axes $[100]$, $[010]$, $[001]$. The stiffness tensor then reduces to the simplest possible form:

$$C_{ijkl}^{\text{cub}} = \begin{pmatrix} C_{1111} & C_{1122} & C_{1122} & 0 & 0 & 0 \\ C_{1122} & C_{1111} & C_{1122} & 0 & 0 & 0 \\ C_{1122} & C_{1122} & C_{1111} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{2323} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{2323} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{2323} \end{pmatrix} \quad (8.37)$$

In matrix notation this can be rewritten

$$C_{mn}^{\text{cub}} = \begin{pmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{44} \end{pmatrix} \quad (8.38)$$

For hexagonal materials the stiffness tensor is referred to two axes in the basal plane and a third one normal to it. Hexagonal metals are isotropic in their basal plane, so that the stiffness tensor is invariant with respect to the orientation of the coordinate basis vectors in the basal plane:

$$C_{ijkl}^{\text{hex}} = \begin{pmatrix} C_{1111} & C_{1122} & C_{1133} & 0 & 0 & 0 \\ C_{1122} & C_{1111} & C_{1133} & 0 & 0 & 0 \\ C_{1133} & C_{1133} & C_{3333} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{2323} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{2323} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(C_{1111} - C_{1122}) \end{pmatrix} \quad (8.39)$$

In matrix notation it can be rewritten

$$C_{mn}^{\text{hex}} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{13} & 0 & 0 & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(C_{11} - C_{12}) \end{pmatrix} \quad (8.40)$$

While in cubic crystals hence only three independent elastic constants remain, that is, C_{11} , C_{12} , and C_{44} , in the hexagonal lattice five constants C_{11} , C_{12} , C_{13} , C_{33} , and C_{44} must be considered. In the case of cubic symmetry, the inversion of the stiffnesses to the compliances and vice versa leads to the relations

$$\begin{aligned} C_{11} &= \frac{(S_{11} + S_{12})}{(S_{11} - S_{12})(S_{11} + 2S_{12})} \\ C_{12} &= \frac{-S_{12}}{(S_{11} - S_{12})(S_{11} + 2S_{12})} \\ C_{44} &= \frac{1}{S_{44}} \end{aligned} \quad (8.41)$$

$$\begin{aligned}
S_{11} &= \frac{(C_{11} + C_{12})}{(C_{11} - C_{12})(C_{11} + 2C_{12})} \\
S_{12} &= \frac{-C_{12}}{(C_{11} - C_{12})(C_{11} + 2C_{12})} \\
S_{44} &= \frac{1}{C_{44}}
\end{aligned}
\tag{8.42}$$

Before using general anisotropic elastic field equations in simulations, it is useful to test the predictions in the isotropic limit. A material is defined as elastically isotropic if the elastic properties are independent of direction. In the case of cubic crystal symmetry, this is realized when $C_{44} = (C_{11} - C_{12})/2$. The deviation from isotropy can in the cubic lattice be quantified by the so-called Zener anisotropy ratio A^z [HL68]:

$$A^z = \frac{2C_{44}}{C_{11} - C_{12}} \tag{8.43}$$

Indeed, most metals deviate considerably from the isotropic limit. Tungsten has the smallest deviation with a Zener ratio of $A^z = 1$ and lithium the largest one with a ratio of $A^z = 9.39$ (Table 8-1). The two elastic constants of isotropic materials are often expressed in terms of μ and ν , which are defined by

$$\mu = C_{44} = C_{2323} = \frac{1}{2}(C_{11} - C_{12}) \tag{8.44}$$

and

$$\nu = \frac{C_{12}}{C_{11} + C_{12}} = \frac{C_{1122}}{C_{1111} + C_{1122}} = -\frac{S_{12}}{S_{11}} = -\frac{S_{1122}}{S_{1111}} \tag{8.45}$$

where μ is often referred to as the shear modulus or modulus of rigidity and ν as Poisson's ratio. The elastic modulus E , which relates elastic stress and strain in the case of tensile deformation, can be expressed in terms of μ and ν :

$$E = 2\mu(1 + \nu) \tag{8.46}$$

For relating the mean hydrostatic stress to dilatation one additionally defines the bulk modulus B^{el} [HL68].

$$B^{\text{el}} = \left(\frac{1}{3}\right) \frac{\sigma_{kk}}{\varepsilon_{ii}} = \lambda + \frac{2}{3}\mu = \frac{E}{3(1 - 2\nu)} \tag{8.47}$$

TABLE 8-1 Elastic Stiffness Constants for Some Cubic Metals

<i>Element</i>	<i>Structure</i>	C_{1111} (GPa)	C_{1122} (GPa)	C_{2323} (GPa)	A^z	ν
Ag	fcc	12.40	9.34	4.61	3.013	0.43
Al	fcc	10.82	6.13	2.85	1.215	0.36
Au	fcc	18.60	15.70	4.20	2.987	0.46
Cr	bcc	35.00	5.78	10.10	0.691	0.14
Cu	fcc	16.84	12.14	7.54	3.209	0.42
Fe	bcc	24.20	14.65	11.20	2.346	0.38
Li	bcc	1.48	1.25	1.08	9.391	0.46

In addition to these constants the Lamé constants μ (see preceding definition) and λ are common as well:

$$\lambda = \frac{2\mu\nu}{1-2\nu} \quad (8.48)$$

In terms of Lamé's constants the elastic modulus is given by

$$E = \frac{\mu(3\lambda + 2\mu)}{\mu + \lambda} \quad (8.49)$$

The use of Lamé's constants allows one to give a compact tensorial expression of the elastic constants and thus of Hooke's law in the isotropic limit:

$$C_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu (\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}) \quad (8.50)$$

$$\sigma_{ij} = \lambda \varepsilon_{kk} \delta_{ij} + 2\mu \varepsilon_{ij} \quad (8.51)$$

The inverse form can be written

$$\varepsilon_{ij} = \frac{1}{2\mu} \sigma_{ij} - \frac{\lambda}{2\mu(3\lambda + 2\mu)} \sigma_{kk} \delta_{ij} \quad (8.52)$$

For presenting strains and stresses associated with dislocations in the isotropic limit, the constant ν is often used instead of λ . The elastic constants are then given by

$$C_{ijkl} = \mu \left(\delta_{ik} \delta_{jl} + \frac{2\nu}{1-2\nu} \delta_{ij} \delta_{kl} \right) \quad (8.53)$$

Accordingly the compliances can be written

$$S_{lnpq} = \frac{1}{2\mu} \left(\delta_{lp} \delta_{nq} - \frac{\nu}{1+\nu} \delta_{ln} \delta_{pq} \right) \quad (8.54)$$

Using equations (8.53) and (8.54) Hooke's law can be written

$$\sigma_{ij} = 2\mu \left(\varepsilon_{ij} + \frac{\nu}{1-2\nu} \delta_{ij} \varepsilon_{kk} \right) \quad (8.55)$$

or in its inverse form

$$\varepsilon_{ij} = \frac{1}{2\mu} \left(\sigma_{ij} - \frac{\nu}{1+\nu} \delta_{ij} \sigma_{kk} \right) \quad (8.56)$$

The equations of equilibrium can also be expressed in a compact form by using μ and ν :

$$\varepsilon_{ikp} \varepsilon_{jmq} \sigma_{pq,km} - \frac{\nu}{1+\nu} (\delta_{ij} \sigma_{pp,kk} - \sigma_{pp,ij}) = 0 \quad (8.57)$$

Some general restrictions for the values of the elastic constants are imposed by the positive definiteness of the strain energy density function W , namely,

$$C_{44} > 0 \quad C_{11} > |C_{12}| \quad C_{11} + 2C_{12} > 0 \quad (8.58)$$

From these conditions it follows that

$$\lambda > 0 \quad \text{and} \quad -1 < \nu < \frac{1}{2} \quad (8.59)$$

8.2.6 Elastic Energy

Tractions and body forces that act on the surface and interior of a small volume element can give rise to an incremental displacement $\delta \mathbf{u}$. Under the assumption of quasi-static conditions, which allows equilibrium of the acting forces, the work done on the considered volume element is given by

$$\delta W = \iint_S \sigma_{ij} \delta u_j dS_i + \iiint_V P_j \delta u_j dV \quad (8.60)$$

where \mathbf{P} is the body force and S the surface enclosing the volume element V [Ste73, BBS79a, Mur87]. By employing Gauss' divergence theorem one can convert surface integrals into volume integrals

$$\iiint_V A_{ij,k} dV = \iint_S A_{ij} dS_k \quad (8.61)$$

where A_{ij} is a tensor field of arbitrary rank (here of rank 2). Equation (8.60) can then be rewritten

$$\delta W = \iiint_V (P_j \delta u_j + \sigma_{ij,i} \delta u_j + \sigma_{ij} \delta u_{j,i}) dV \quad (8.62)$$

By exploiting the conditions for static equilibrium, equation (8.23), one obtains

$$\delta W = \iiint_V \sigma_{ij} \delta u_{j,i} dV \quad (8.63)$$

This equation substantiates the fact that rigid-body rotations do not contribute to the elastic work. Owing to the symmetry of the stress tensor, $\sigma_{ij} = \sigma_{ji}$, and the antisymmetry of the components of the displacement gradient tensor which describe the rigid-body rotation, $\omega_{ij} = -\omega_{ji}$, the corresponding sum of these products will always balance. Considering an infinitesimal volume element, the work per volume can be written

$$\delta W = \sigma_{ij} \delta \varepsilon_{ij} \quad (8.64)$$

With the relation between stress and strain restricted to a linear one, the elastic potential, that is, the elastic strain energy per volume, can be derived by inserting Hooke's law:

$$W = \frac{1}{2} \sigma_{ij} \varepsilon_{ij} = \frac{1}{2} C_{ijkl} \varepsilon_{ij} \varepsilon_{kl} \quad (8.65)$$

8.2.7 Green's Tensor Function in Elasticity Theory

Green's functions represent a group of useful tools in solving partial differential equations. However, they can also be employed in solving ordinary differential equations. This can be briefly demonstrated by the following example. The differential equation

$$\frac{d^2 x}{dt^2} + \omega^2 x = f(t) \quad \text{with} \quad x_0 = x'_0 = 0 \quad (8.66)$$

describes the oscillation of a point mass suspended by a spring which is characterized by some given force function $f(t)$. This force function can be dismantled into a sequence of single impulses.

$$f(t) = \int_0^{\infty} f(t') \delta(t' - t) dt' \quad (8.67)$$

where $\delta(t' - t)$ is the Dirac delta function describing an infinitesimally narrow function localized at t' , the integral of which is 1. In the next step, one solves equation (8.66), but with the original force function $f(t)$ replaced by the delta function $\delta(t' - t)$; that is, one obtains the response $G(t, t')$ of the system to a unit impulse at t' . Equation (8.66) can then be rewritten

$$\frac{d^2 G(t, t')}{dt^2} + \omega^2 G(t, t') = \delta(t' - t) \quad (8.68)$$

Finally, one obtains a solution of the original differential equation by adding up the responses of many such small unit impulses.

$$x(t) = \int_0^{\infty} G(t, t') f(t') dt' \quad (8.69)$$

This example demonstrates that the use of the Green's function often makes it possible to find a solution of a nonhomogeneous differential equation by solving the system for a delta function on the right-hand side.

In dislocation theory the time-independent Green's tensor function $G_{ij}(\mathbf{x}, \mathbf{x}')$ is used for the integration of the differential equations of elasticity, equation (8.32). The tensor field $G_{ij}(\mathbf{x}, \mathbf{x}')$ gives the displacement along the $x e_i$ axis at \mathbf{x} in response to a delta-type unit point force exerted parallel to the $x e_j$ axis at \mathbf{x}' . This approach is valid for an infinite body with homogeneous elastic properties in static equilibrium. In the present case the Green's tensor satisfies the conditions of translational invariance, centrosymmetry, and reciprocity [BBS79a].

$$G_{ij}(\mathbf{x}, \mathbf{x}') = G_{ij}(\mathbf{x} - \mathbf{x}') = G_{ji}(\mathbf{x} - \mathbf{x}') = G_{ij}(\mathbf{x}' - \mathbf{x}) \quad (8.70)$$

For the derivatives similar relations follow:

$$G_{ij,s}(\mathbf{x} - \mathbf{x}') = -G_{ji,s'}(\mathbf{x} - \mathbf{x}') = G_{ij,s}(\mathbf{x}' - \mathbf{x}) \quad (8.71)$$

where primed subscripts denote derivatives with respect to \mathbf{x}' and unprimed subscripts those to \mathbf{x} . The same notation is used for the second derivatives,

$$G_{ij,sk}(\mathbf{x} - \mathbf{x}') = -G_{ji,s'k'}(\mathbf{x} - \mathbf{x}') = G_{ij,sk}(\mathbf{x}' - \mathbf{x}) \quad (8.72)$$

Provided that the dislocation motion is uniform, the time-independent Green's tensor may be used for dislocation dynamics as well. In the case of non-uniform motion the time-dependent Green's tensor must be employed [Bar96].

The Green's tensor field and its derivatives required in dislocation theory must, for the general anisotropic case, be evaluated by numerical techniques. However, in the isotropic limit it can be derived analytically [IL92]:

$$G_{ij}(\mathbf{x} - \mathbf{x}') = \frac{1}{16\pi\mu(1-\nu)} \frac{1}{|\mathbf{x} - \mathbf{x}'|} \left[(3-4\nu)\delta_{ij} + \frac{(x_i - x'_i)(x_j - x'_j)}{(\mathbf{x} - \mathbf{x}')^2} \right] \quad (8.73)$$

As will be shown later not the Green's tensor function itself but its spatial derivatives are required to solve dislocation field equations. For the general anisotropic case the N th derivative of equation (8.73) is given by

$$\begin{aligned} G_{ij,s_1,s_2,\dots,s_N}(\mathbf{x} - \mathbf{x}') &= \frac{1}{8\pi^2} \oint_{|\mathbf{z}|=1} (\mathbf{z}\mathbf{z})_{ij}^{-1} \delta_{s_1,s_2,\dots,s_N}[\mathbf{z}(\mathbf{x} - \mathbf{x}')] dS \\ &= \frac{1}{8\pi^2 |\mathbf{x} - \mathbf{x}'|^{N+1}} \oint_{|\mathbf{z}|=1} (\mathbf{z})_{ij}^{-1} z_{s_1} z_{s_2} \dots z_{s_N} \times \delta^{(N)}(\mathbf{z}\mathbf{T}) dS \end{aligned} \quad (8.74)$$

where \mathbf{T} is defined by $\mathbf{x} - \mathbf{x}' = \mathbf{T} |\mathbf{x} - \mathbf{x}'|$. \mathbf{z} is the integration variable lying in the plane $\mathbf{T}\mathbf{z} = 0$. The second-rank symmetric matrix integrand $(\mathbf{z}\mathbf{z})_{ij}^{-1}$ is the inverse of the Christoffel stiffness matrix $(\mathbf{z}\mathbf{z})_{ij}$ [BAG⁺72]. This operator with the general form $(\mathbf{ab})_{jk}$ is defined by

$$(\mathbf{ab})_{jk} = a_i C_{ijkl} b_l \quad (8.75)$$

The inverse of the symmetric stiffness matrix with the general form $(\mathbf{aa})_{ij}^{-1}$ is given by

$$(\mathbf{aa})_{ij}^{-1} = \frac{\epsilon_{ism} \epsilon_{jrw} (\mathbf{nn})_{mw} (\mathbf{nn})_{sr}}{2 \epsilon_{pgn} (\mathbf{nn})_{1p} (\mathbf{nn})_{2g} (\mathbf{nn})_{3n}} \quad (8.76)$$

Using the preceding relations and the definition for $G_{ij}(\mathbf{x} - \mathbf{x}')$, the displacement due to a point force \mathbf{F} can be expressed by

$$u_i(\mathbf{x}) = G_{ij}(\mathbf{x} - \mathbf{x}') F_j \quad (8.77)$$

The first derivative can then be written

$$u_{i,m}(\mathbf{x}) = G_{ij,m}(\mathbf{x} - \mathbf{x}') F_j \quad (8.78)$$

Following Hooke's law and the symmetry relations, equations (8.28) and (8.31), the resulting stress can then be written

$$\sigma_{kp} = C_{kpi m} u_{i,m} = C_{kpi m} G_{ij,m}(\mathbf{x} - \mathbf{x}') F_j \quad (8.79)$$

Using the static equilibrium condition the governing equation for the Green's tensor can be derived:

$$C_{kpi m} G_{ij,m p}(\mathbf{x} - \mathbf{x}') + \delta_{kj} \delta(\mathbf{x} - \mathbf{x}') = 0 \quad (8.80)$$

The general solution for the displacements expressed in terms of the Green's function is

$$\begin{aligned} u_j(\mathbf{x}) &= \iiint_V G_{jk}(\mathbf{x} - \mathbf{x}') f_k(\mathbf{x}') dV' + \iint_S C_{kpi m} u_{i,m'}(\mathbf{x}') G_{kj}(\mathbf{x} - \mathbf{x}') dS'_p \\ &\quad - \iint_S C_{kpi m} u_k(\mathbf{x}') G_{ij,m'}(\mathbf{x} - \mathbf{x}') dS'_p \end{aligned} \quad (8.81)$$

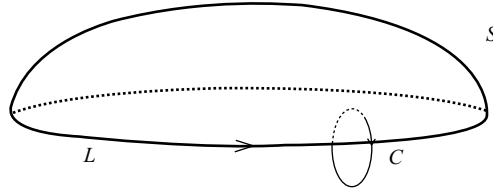


FIGURE 8-1 A closed dislocation loop L and integration circuit C used for the Burgers vector determination, equation (8.82). S is an arbitrary surface bounded by the dislocation line L [Vol07, Mur63, BBS79a].

where $f_k(\mathbf{x}')$ is the body force density. Equations (8.77)–(8.81) are generally valid in anisotropic linear elastic media and are not confined to dislocation problems.

A special solution for the displacement due to a dislocation is obtained by considering the appropriate displacement boundary conditions. By definition, L is a dislocation if for any closed circuit C the following equation applies (see Figure 8-1):

$$\oint_C u_{j,k}(\mathbf{x}) dx_k = \begin{cases} b_j & \text{if } C \text{ irreducibly encloses } L \\ 0 & \text{otherwise} \end{cases} \quad (8.82)$$

Since the dislocation is defined by displacement boundary conditions one may set the body force density equal to zero, so that equation (8.81) simplifies to

$$u_j(\mathbf{x}) = -C_{kpin} b_k \iint_S G_{ij,m'}(\mathbf{x} - \mathbf{x}') dS'_p \quad (8.83)$$

This equation is referred to as Volterra's displacement formula [Vol07]. The surface S can be any surface bounded by L . The distortion field can be derived by differentiating equation (8.83) with respect to \mathbf{x} :

$$u_{j,s}(\mathbf{x}) = -C_{kpin} b_k \iint_S G_{ij,m's'}(\mathbf{x} - \mathbf{x}') dS'_p \quad (8.84)$$

After the application of Stokes' theorem, by which a surface integral can be rendered into a line integral, equation (8.84) reads:

$$u_{j,s}(\mathbf{x}) = -C_{kpin} b_k \epsilon_{qps} \oint_L G_{ij,m}(\mathbf{x} - \mathbf{x}') dx'_q \quad (8.85)$$

This expression is commonly referred to as Mura's equation [Mur63].

8.2.8 The Airy Stress Function in Elasticity Theory

In field theories it can be pertinent to express vector quantities in terms of gradients of a scalar potential. In some theories these scalar potentials have an obvious physical significance, for instance, in gravitation theory. In elasticity theory an equivalent method can be used, although endowed with less physical significance. This approach allows one to express stress and displacement fields in terms of spatial derivatives of scalar potentials. The latter are referred

to as stress or displacement functions. In 2D problems of elasticity the following form of expressing stresses as derivatives of a potential was suggested by G. B. Airy in 1862 and is therefore referred to as the Airy stress function:

$$\sigma_{11} = \frac{\partial^2 \phi}{\partial x_2^2} \quad \sigma_{22} = \frac{\partial^2 \phi}{\partial x_1^2} \quad \sigma_{12} = -\frac{\partial^2 \phi}{\partial x_2 \partial x_1} \quad (8.86)$$

For tackling 3D problems a different notation is often used, namely,

$$\sigma_{11} = \frac{\partial^2 \psi}{\partial x_1^2} \quad \sigma_{22} = \frac{\partial^2 \psi}{\partial x_2^2} \quad \sigma_{12} = \frac{\partial^2 \psi}{\partial x_1 \partial x_2} \quad (8.87)$$

Equation (8.86) transforms as a Cartesian tensor. As was outlined in the preceding sections, stress and strain fields must satisfy the equations of equilibrium and compatibility if they are to describe permissible states of elastic bodies. This condition imposes certain constraints on the selection of stress functions. In general these constraints are expressed by requiring that the stress functions are solutions of certain partial differential equations. This stipulation is readily demonstrated by substituting equation (8.86) into the equilibrium equations (8.23), assuming plane strain conditions and the absence of body forces:

$$\begin{aligned} \sigma_{11,1} + \sigma_{12,2} &= \frac{\partial^3 \phi}{\partial x_2^2 \partial x_1} + \frac{\partial^3 \phi}{\partial x_1 \partial x_2^2} = 0 \\ \sigma_{21,1} + \sigma_{22,2} &= \frac{\partial^3 \phi}{\partial x_1^2 \partial x_2} + \frac{\partial^3 \phi}{\partial x_2 \partial x_1^2} = 0 \end{aligned} \quad (8.88)$$

Thus, the Airy stress function automatically satisfies the equilibrium equations, provided the body forces are equal to zero. If the body forces are not zero, they must be expressed as conservative forces, that is, as negative gradients of a scalar potential, so that equation (8.86) can be cast into a generalized form. However, the constraints imposed by compatibility must be included as well. By using Hooke's law, equation (8.51), the compatibility equations (8.25), and the Airy stress function, equation (8.86), one obtains the biharmonic equation for plane strain conditions:

$$\frac{\partial^4 \phi}{\partial x_1^4} + 2 \frac{\partial^4 \phi}{\partial x_1^2 \partial x_2^2} + \frac{\partial^4 \phi}{\partial x_2^4} = \left(\frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} \right)^2 \phi = 0 \quad (8.89)$$

Except for some well-investigated cases, the analytical solution of the biharmonic equation is an intricate task. Thus, most modern approaches to solving it are based on representing the underlying stress function in terms of analytic functions of the complex variable and subsequent numerical solution.

8.3 Dislocation Statics

8.3.1 Introduction

In the preceding sections the fundamentals of linear isotropic and anisotropic elasticity theory that are required in dislocation theory have been reviewed. In this section the basic field equations for displacement, strain, and stress associated with infinite dislocations and finite

dislocation segments will be derived. These equations form the basis for conducting space- and time-discretized numerical simulations of dislocation dynamics.

Two stages of dislocation field equations will be discussed. First, the field description of infinite dislocations in two dimensions will be derived, exploiting anti-plane strain conditions for the screw dislocation and plane strain conditions for the edge dislocation. Second, compact tensorial 3D field equations will be derived for arbitrary finite dislocation segments.

The latter field equations make it possible to dismantle 3D dislocations into sequentially arranged, piecewise straight, dislocation segments. This approach enables one to calculate the statics and dynamics associated with arbitrary 3D dislocation arrangements that are not amenable to closed analytical approaches.

For solving the various dislocation field problems one can use the equilibrium equations (8.23), combined with the compatibility equations (8.25), or the Green's function method, equation (8.81). While the first two equation systems are used in the sextic approach [Str58], the latter method is commonly referred to as the integral formalism [BBS79a]. Which of these methods will be used in the following depends on the required field quantities and on existing symmetries. All field quantities are given for both the isotropic and the general anisotropic case.

8.3.2 Two-Dimensional Field Equations for Infinite Dislocations in an Isotropic Linear Elastic Medium

Edge Dislocation

The 2D field equation for edge dislocations with an infinite extension of their dislocation line can be derived by solving the compatibility equations (8.25) for plane strain conditions under consideration of the equilibrium equations (8.23). The plane strain state describes a situation where all particles of a body are displaced parallel to an arbitrary plane, and the displacements are independent of their coordinate parallel to the plane normal. For an infinite edge dislocation with its Burgers vector parallel to say x_1 and tangent vector parallel to say x_3 , only one displacement component, namely, that parallel to the dislocation line, u_3 , is equal to zero. The two remaining displacements are not equal to zero. However, they are independent of x_3 [HL68]. To express this matter formally, one can write

$$u_1 = f(x_1, x_2) \quad u_2 = g(x_1, x_2) \quad u_3 = 0 \quad \frac{\partial u_1}{\partial x_3} = 0 \quad \frac{\partial u_2}{\partial x_3} = 0 \quad (8.90)$$

While the stresses that surround a screw dislocation can be immediately expressed in terms of its quite simple displacement field, this method is less tractable in the case of the edge dislocation. Since the direct determination of the displacement field of the edge dislocation is too complicated, the prescribed plane strain state advocates the use of the Airy stress function, equations (8.86)–(8.88). By using the substitution

$$(\sigma_{11} + \sigma_{22}) = \nabla^2 \phi \quad (8.91)$$

the biharmonic equation (8.88) can be transformed into the harmonic Laplace equation. A detailed description of the solution of the harmonic equation is given by Hirth and Lothe [HL68]. The result for the stress function ϕ amounts to

$$\phi = -\frac{\mu b x_2}{4\pi(1-\nu)} \ln(x_1^2 + x_2^2) \quad (8.92)$$

The stress field of the infinite edge dislocation parallel to x_3 is then obtained by employing equation (8.86):

$$\begin{aligned}
 \sigma_{11} &= -\frac{\mu b}{2\pi(1-\nu)} \frac{x_2(3x_1^2 + x_2^2)}{(x_1^2 + x_2^2)^2} \\
 \sigma_{22} &= \frac{\mu b}{2\pi(1-\nu)} \frac{x_2(x_1^2 - x_2^2)}{(x_1^2 + x_2^2)^2} \\
 \sigma_{12} &= \frac{\mu b}{2\pi(1-\nu)} \frac{x_1(x_1^2 - x_2^2)}{(x_1^2 + x_2^2)^2} \\
 \sigma_{33} &= \nu(\sigma_{11} + \sigma_{22}) = -\frac{\mu b \nu}{\pi(1-\nu)} \frac{x_2}{x_1^2 + x_2^2}
 \end{aligned} \tag{8.93}$$

The strain field is readily obtained by using Hooke's law, equation (8.51).

Screw Dislocation

The 2D field equation for screw dislocations with an infinite extension of their dislocation line can be derived by solving the equilibrium equations (8.23) under anti-plane strain conditions. The anti-plane strain state describes a situation where all particles of a body are displaced in a direction normal to an arbitrary plane, and the displacements are independent of this direction. For an infinite screw dislocation with its Burgers vector and tangent vector parallel to say x_3 , only one component, that is, u_3 , which is parallel to the Burgers vector and independent of x_3 , appears in the displacement field:

$$u_1 = 0 \quad u_2 = 0 \quad u_3 = f(x_1, x_2) \quad \frac{\partial u_3}{\partial x_3} = 0 \tag{8.94}$$

For the infinite homogeneous linear elastic body, Hooke's law in the isotropic limit is given by the expression

$$\sigma_{ij} = \lambda \varepsilon_{kk} \delta_{ij} + 2\mu \varepsilon_{ij} \tag{8.95}$$

Since all displacements except u_3 are zero and u_3 is independent of x_3 , the dilatation of bodies in an anti-plane strain state must always be equal to zero, that is, $\varepsilon_{kk} = 0$. Hooke's law then reduces to

$$\sigma_{ij} = 2\mu \varepsilon_{ij} \tag{8.96}$$

Inserting the preceding equation into the equilibrium equation in the absence of body forces, $\sigma_{ij,j} = 0$, leads to

$$\varepsilon_{ij,j} = \frac{1}{2} \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) = 0 \tag{8.97}$$

For the infinite screw dislocation this expression reduces to the harmonic equation

$$\frac{\partial^2 u_3}{\partial x_1^2} + \frac{\partial^2 u_3}{\partial x_2^2} = 0 \tag{8.98}$$

The solution of this expression is readily obtained by considering that the displacement u_3 increases from zero to b by traversing a closed circuit about the dislocation line vector. As a

reasonable approach for expressing u_3 as a function of the angle θ about the dislocation line one can use

$$u_3 = b \left(\frac{\theta}{2\pi} \right) \quad \text{with} \quad \theta = \arctan \left(\frac{x_2}{x_1} \right) \quad (8.99)$$

which is indeed a solution of the harmonic equation. By differentiating u_3 with respect to x_1 and x_2 one obtains the strain field and by inserting the result into equation (8.96) one obtains the stress field of an infinite screw dislocation with its line vector being tangent to x_3 :

$$\begin{aligned} \sigma_{13} = \sigma_{31} &= -\frac{\mu b}{2\pi} \frac{x_2}{x_1^2 + x_2^2} \\ \sigma_{23} = \sigma_{32} &= \frac{\mu b}{2\pi} \frac{x_1}{x_1^2 + x_2^2} \end{aligned} \quad (8.100)$$

8.3.3 Two-Dimensional Field Equations for Infinite Dislocations in an Anisotropic Linear Elastic Medium

Introduction

Explicit analytical expressions for the stress field equations in the anisotropic case can be obtained by using the sextic approach. A detailed description of this mathematical procedure, which is based on solving the equilibrium equations (8.23) combined with the compatibility equations (8.25), was given by Stroh [Str58], Steeds [Ste73], and Mura [Mur87].

The basic task in this eigenvalue problem is to find the roots of a sextic equation. Relatively simple analytical solutions are available for certain cases of high symmetry, namely, when the dislocation line is either parallel or perpendicular to a twofold or sixfold axis. A large number of possible solutions for real crystals were derived and thoroughly discussed by Steeds [Ste73]. This section presents the general field solution for the simplest possible case, namely, for twofold symmetry.

Edge Dislocation

The coordinate system is defined in such a manner that the dislocation line points in the negative x_3 -direction. Assuming that one of the axes perpendicular to the dislocation line ($-x_3$) is parallel to a twofold axis, and using the matrix notation given in equation (8.35), the stiffness tensor can be written in the dislocation coordinate system:

$$\mathbf{C} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{pmatrix} \quad (8.101)$$

For further calculations it is pertinent to introduce some abbreviations, namely,

$$\bar{C} = (C_{11} C_{22})^{1/2} \quad \lambda = \left(\frac{C_{11}}{C_{22}} \right)^{1/4} \quad \phi = \frac{1}{2} \arccos \left(\frac{C_{12}^2 + 2 C_{12} C_{66} - \bar{C}_{11}^2}{2 \bar{C}_{11} C_{66}} \right) \quad (8.102)$$

Furthermore, the solutions are confined to the case

$$2C_{66} + C_{12} - \bar{C}_{11} > 0 \quad (8.103)$$

so that ϕ gives a real value. Finally, it is useful to define

$$q^2 = x_1^2 + 2x_1x_2\lambda\cos\phi + x_2^2\lambda^2 \quad t^2 = x_1^2 - 2x_1x_2\lambda\cos\phi + x_2^2\lambda^2 \quad (8.104)$$

Using these constraints, the equations for the displacement and stress field can be written

$$\begin{aligned} u_1 = & -\frac{b_1}{4\pi} \left[\arctan\left(\frac{2x_1x_2\lambda\sin\phi}{x_1^2 - (\lambda x_2)^2}\right) + \frac{\bar{C}_{11}^2 - C_{12}^2}{2\bar{C}_{11}^2\sin(2\phi)} \ln\left(\frac{q}{t}\right) \right] \\ & - \frac{b_2}{4\pi\lambda\bar{C}_{11}\sin(2\phi)} \left[(\bar{C}_{11} - C_{12})\cos\phi \ln(qt) \right. \\ & \left. - (\bar{C}_{11} + C_{12})\sin\phi \arctan\left(\frac{x_1^2\sin(2\phi)}{(\lambda x_2)^2 - x_1^2\cos(2\phi)}\right) \right] \quad (8.105) \end{aligned}$$

$$\begin{aligned} u_2 = & \frac{\lambda b_1}{4\pi\bar{C}_{11}\sin(2\phi)} \left[(\bar{C}_{11} - C_{12})\cos\phi \ln(qt) \right. \\ & \left. - (\bar{C}_{11} + C_{12})\sin\phi \arctan\left(\frac{(\lambda x_2)^2\sin(2\phi)}{x_1^2 - (\lambda x_2)^2\cos(2\phi)}\right) \right] \\ & - \frac{b_2}{4\pi} \left[\arctan\left(\frac{2x_1x_2\lambda\sin\phi}{x_1^2 - (\lambda x_2)^2}\right) - \frac{\bar{C}_{11}^2 - C_{12}^2}{2\bar{C}_{11}^2\sin(2\phi)} \ln\left(\frac{q}{t}\right) \right] \quad (8.106) \end{aligned}$$

Switching partly to the fourth-rank tensorial form of the stiffness tensor, the stress field associated with the infinite edge dislocation can compactly be written in tensorial notation:

$$\begin{aligned} \sigma_{ij} = & -\frac{b_1\lambda(C_{12} - \bar{C}_{11})}{4\pi(qt)^2\bar{C}_{11}C_{66}\sin\phi} \\ & \left\{ C_{ij11} \left[(\bar{C}_{11} + C_{12} + C_{66})x_1^2x_2 + \lambda^2C_{66}x_2^3 \right] - C_{ij12}(C_{12} + \bar{C}_{11})x_1(x_1^2 - (\lambda x_2)^2) \right. \\ & \left. - \frac{C_{ij22}}{C_{22}} \left[(C_{12}^2 + \bar{C}_{11}C_{12} + 2C_{12}C_{66} + \bar{C}_{11}C_{66})x_1^2x_2 - \bar{C}_{11}C_{66}\lambda^2x_2^3 \right] \right\} \\ & + \frac{b_2\lambda(C_{12} - \bar{C}_{11})}{4\pi(qt)^2\bar{C}_{11}C_{66}\sin\phi} \\ & \left\{ C_{ij22} \left[(\bar{C}_{11} + C_{12} + C_{66})(\lambda x_2)^2x_1 + C_{66}x_1^3 \right] - C_{ij12}(C_{12} + \bar{C}_{11})x_2(x_1^2 - (\lambda x_2)^2) \right. \\ & \left. - \frac{C_{ij11}}{C_{11}} \left[(C_{12}^2 + \bar{C}_{11}C_{12} + 2C_{12}C_{66} + \bar{C}_{11}C_{66})(\lambda x_2)^2x_1 - \bar{C}_{11}C_{66}x_1^3 \right] \right\} \quad (8.107) \end{aligned}$$

Screw Dislocation

For infinite screw dislocations the corresponding expressions are, for the displacement field,

$$u_3 = -\frac{b_3}{2\pi} \arctan\left[\frac{(C_{44}C_{55} - C_{45}^2)^{1/2}x_2}{C_{44}x_1 - C_{45}^2x_2}\right] \quad (8.108)$$

and for the stress field,

$$\begin{aligned}\sigma_{13} &= -\frac{b_3}{2\pi} \left(C_{44} C_{55} - C_{45}^2 \right)^{1/2} \left(\frac{C_{45} x_1 - C_{55} x_2}{C_{44} x_1^2 - 2 C_{45} x_1 x_2 + C_{55} x_2^2} \right) \\ \sigma_{23} &= -\frac{b_3}{2\pi} \left(C_{44} C_{55} - C_{45}^2 \right)^{1/2} \left(\frac{C_{44} x_1 - C_{45} x_2}{C_{44} x_1^2 - 2 C_{45} x_1 x_2 + C_{55} x_2^2} \right)\end{aligned}\quad (8.109)$$

8.3.4 Three-Dimensional Field Equations for Dislocation Segments in an Isotropic Linear Elastic Medium

Analytical calculations of displacement, strain, and stress fields associated with dislocation arrays of low symmetry lead to very complicated expressions. For complex dislocation structures a closed analytical treatment is thus no longer possible.

For nonetheless calculating arbitrary dislocation arrays, it is hence straightforward to approximate real dislocation arrangements by sequences of piecewise straight segments, which are much shorter as compared with the entire dislocation.

The local field quantities can then be computed by a summation of the contributions of all individual segments assembled in the array. The summation is possible since the line integrals that occur in Mura's expression for the calculation of the displacement field, equation (8.85), transform like vectors for each dislocation line segment [HL82, Mur87]. Consequently, all tensor quantities obtained for the individual segments can be transformed to a common coordinate system.

Owing to the fact that dislocation lines must not end within an otherwise perfect region of crystal, it is clear that the segmentation of dislocation lines is only allowed if the segments are interconnected or terminate at free surfaces under consideration of image forces.

However, the fundamental problem remains, of how the occurrence of connected isolated segments can be *physically* interpreted. An elegant justification is given in Figure 8-2, which shows how a curved dislocation line can be approximated by combining dislocation loops with identical Burgers vectors but alternating line vectors. While the parallel portions of these loops align to form a continuous dislocation line, the antiparallel portions can be arranged in a manner to allow mutual annihilation. This construction also substantiates the fact that a segmented dislocation cannot terminate within an otherwise perfect crystal region. At the end points of the dislocation, which consist of the parallel segments, the underlying loops do not simply vanish but continue in the crystal. This shows that the introduction of a geometrical cutoff at these end points would entail an error which amounts to the stresses contributed by the remaining semi-infinite loop portions. This error vanishes if the segmented loop approximation is closed. A more detailed discussion of such constructions has been published by Brown [Bro67] and Bacon *et al.* [BBS79a].

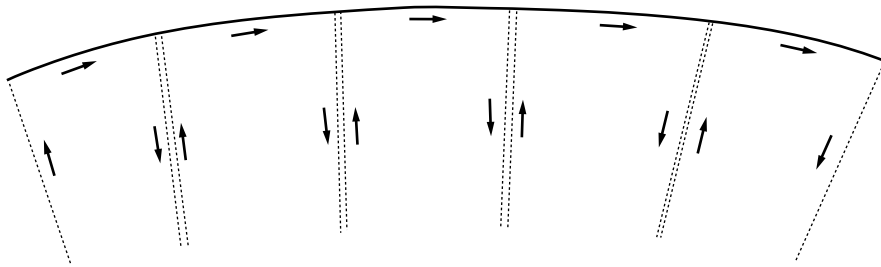


FIGURE 8-2 Approximation of a curved dislocation line by combining dislocation loops with identical Burgers vectors but alternating line vectors (arrows).

Pioneering contributions on the discretization of 3D dislocations into sequences of piecewise straight segments and the subsequent calculation of field quantities arising from the segments were published [de 60, Bro67, IO67, AB74, BBS79a, HL82, Mur87, Dev95]. For the derivation of the corresponding strain and stress tensors the authors employed either the sextic theory of Stroh or the integral theory using Green's tensor function method.

As a starting point for presenting 3D field expressions for piecewise straight dislocation segments it is convenient to follow the derivation of Hirth and Lothe [HL82]. The stresses are first derived in rectangular coordinates x_1, x_2, x_3 . The vector \mathbf{R} indicates the spacing between the coordinates that are fixed on the dislocation line, $\mathbf{r}'^T = (x'_1, x'_2, x'_3)$, and the field coordinates under inspection, $\mathbf{r}^T = (x_1, x_2, x_3)$, so that

$$\begin{aligned} \mathbf{R} &= \mathbf{r} - \mathbf{r}' \\ R &= \sqrt{(x_1 - x'_1)^2 + (x_2 - x'_2)^2 + (x_3 - x'_3)^2} \end{aligned} \quad (8.110)$$

applies. It is assumed that the dislocation line vector is parallel to x_3 . For simplicity, the dislocation line passes through the origin, so that $x'_1 = 0$ and $x'_2 = 0$. The preceding expression then reduces to

$$R = \sqrt{(x_1)^2 + (x_2)^2 + (x_3 - x'_3)^2} \quad (8.111)$$

By combining the expressions for the stress, equations (8.26) or (8.34), rendered into the isotropic limit, equation (8.55), with the Green's function for the isotropic case, equation (8.73), and the formula for the displacement gradient after applying Stokes' theorem, equation (8.73), one obtains a convenient line integral expression for the stress,

$$\begin{aligned} \sigma_{11} &= \frac{\mu}{4\pi(1-\nu)} \int b_m \epsilon_{imz} \left(\frac{\partial^3 R}{\partial x_i \partial x_1^2} - \frac{\partial}{\partial x_i} \nabla^2 R \right) dx'_3 \\ &= \frac{\mu}{4\pi(1-\nu)} \int \left[b_1 \left(-\frac{\partial^3 R}{\partial x_2 \partial x_1^2} + \frac{\partial}{\partial x_2} \nabla^2 R \right) + b_2 \left(\frac{\partial^3 R}{\partial x_1^3} + \frac{\partial}{\partial x_1} \nabla^2 R \right) \right] dx'_3 \end{aligned} \quad (8.112)$$

where μ is the bulk shear modulus, ν Poisson's ratio, $\mathbf{b}^T = (b_1, b_2, b_3)$ the Burgers vector, and ϵ_{ijk} the totally antisymmetric Levi-Civita operator, the components of which are defined to be 1 if the suffixes are in cyclic order, -1 if they are in reverse cyclic order, and 0 if any two suffixes are the same. The spatial variables $x'_1 = 0$ and $x'_2 = 0$ were set equal to zero. It must be noted that the partial derivatives $\partial/\partial x_i$ are equal to $-\partial/\partial x'_i$. A more detailed derivation of this expression is given by Hirth and Lothe [HL82].

Dropping the terms that are independent of x'_3 the line integrals in equation (8.112) can be solved according to

$$\begin{aligned} \int \frac{\partial}{\partial x_2} \nabla^2 R dx'_3 &= \int \frac{\partial}{\partial x_2} \frac{2}{R} dx'_3 = - \int \frac{2x_2}{R^3} dx'_3 = \\ &= - \frac{2x_2}{(x_1^2 + x_2^2) R} = - \frac{2x_2}{R} \frac{(x'_3 - x_3)}{[R^2 - (x'_3 - x_3)^2]} \end{aligned} \quad (8.113)$$

Proceeding in this manner for all the stress components ($x'_1 = 0, x'_2 = 0$) leads to

$$\sigma'_{11} = b_1 \frac{x_2}{R(R+\lambda)} \left[1 + \frac{x_1^2}{R^2} + \frac{x_1^2}{R(R+\lambda)} \right] + b_2 \frac{x_1}{R(R+\lambda)} \left[1 - \frac{x_1^2}{R^2} - \frac{x_1^2}{R(R+\lambda)} \right] \quad (8.114)$$

$$\sigma'_{22} = -b_1 \frac{x_2}{R(R+\lambda)} \left[1 - \frac{x_2^2}{R^2} - \frac{x_2^2}{R(R+\lambda)} \right] - b_2 \frac{x_1}{R(R+\lambda)} \left[1 + \frac{x_2^2}{R^2} + \frac{x_2^2}{R(R+\lambda)} \right] \quad (8.115)$$

$$\sigma'_{33} = b_1 \left[\frac{2\nu x_2}{(R+\lambda)} + \frac{x_2 \lambda}{R^3} \right] + b_2 \left[-\frac{2\nu x_1}{(R+\lambda)} - \frac{x_1 \lambda}{R^3} \right] \quad (8.116)$$

$$\sigma'_{12} = -b_1 \frac{x_1}{R(R+\lambda)} \left[1 - \frac{x_2^2}{R^2} - \frac{x_2^2}{R(R+\lambda)} \right] + b_2 \frac{x_2}{R(R+\lambda)} \left[1 - \frac{x_1^2}{R^2} - \frac{x_1^2}{R(R+\lambda)} \right] \quad (8.117)$$

$$\sigma'_{13} = -b_1 \frac{x_1 x_2}{R^3} + b_2 \left(-\frac{\nu}{R} + \frac{x_1^2}{R^3} \right) + b_3 \frac{x_2(1-\nu)}{R(R+\lambda)} \quad (8.118)$$

$$\sigma'_{23} = b_1 \left(\frac{\nu}{R} - \frac{x_2^2}{R^3} \right) + b_2 \frac{x_1 x_2}{R^3} - b_3 \frac{x_1(1-\nu)}{R(R+\lambda)} \quad (8.119)$$

where $\sigma'_{ij} = \sigma_{ij} 4\pi(1-\nu)/\mu$ and $\lambda = x'_3 - x_3$. The stress at \mathbf{r} from a straight segment which lies between $x'_3(\mathbf{A})$ and $x'_3(\mathbf{B})$ then amounts to

$$\sigma'_{ij}{}^{\mathbf{A} \rightarrow \mathbf{B}}(\mathbf{r}) = [\sigma_{ij}(\mathbf{r})]_{\mathbf{r}'=\mathbf{B}} - [\sigma_{ij}(\mathbf{r})]_{\mathbf{r}'=\mathbf{A}} \quad (8.120)$$

The preceding sets of equations are limited in their applicability in that they depend on the coordinate system employed. Furthermore, they are only formulated for situations where straight segments cut through the origin. Therefore, in the following text they are transformed into a dyadic form which is more convenient for numerical purposes.

Following de Wit [de 60] and Devincere [Dev95], the starting point of the derivation is the expression for the stress field associated with an infinite straight dislocation line at a point \mathbf{r} in an unbounded, isotropic, linear, homogeneous, elastic medium:

$$\sigma_{ij}(\mathbf{r}) = \frac{\mu b_n}{8\pi} \left[q_{,mq} (\epsilon_{jmn} t_i + \epsilon_{imn} t_j) + \frac{2}{(1-\nu)} \epsilon_{kmn} (q_{,mi} - q_{,mq} \delta_{ij}) t_k \right] \quad (8.121)$$

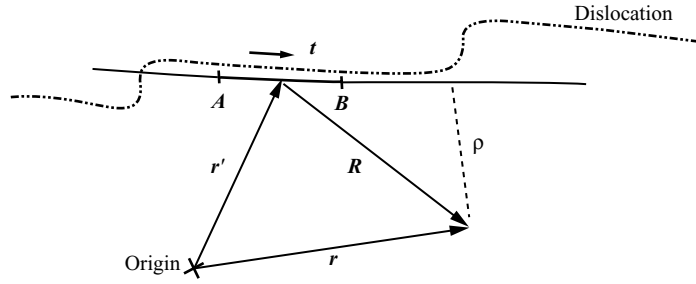


FIGURE 8-3 Schematic diagram of the vector geometry used for deriving the field equations (8.123) for infinite straight dislocations.

where \mathbf{t} is the unit vector tangent to the dislocation line, \mathbf{b} the Burgers vector, and q the indefinite line integral along the dislocation line [de 67]. Symbols following the commas refer to spatial derivatives.² After deriving the spatial derivatives of q and introducing the tensor operator

$$[\mathbf{a} \mathbf{b} \mathbf{c}]_{ij} = \frac{1}{2} [(\mathbf{a} \times \mathbf{b})_i c_j + (\mathbf{a} \times \mathbf{b})_j c_i] \quad (8.122)$$

one obtains for the stress field [Dev95]

$$\sigma_{ij}(\mathbf{r}) = \frac{\mu}{\pi Y^2} \left\{ [\mathbf{b} \mathbf{Y} \mathbf{t}]_{ij} - \frac{1}{(1-\nu)} [\mathbf{b} \mathbf{t} \mathbf{Y}]_{ij} - \frac{(\mathbf{b}, \mathbf{Y}, \mathbf{t})}{2(1-\nu)} \left[\delta_{ij} + t_i t_j + \frac{2}{Y^2} (\rho_i Y_j + \rho_j Y_i + \frac{L}{|\mathbf{R}|} Y_i Y_j) \right] \right\} \quad (8.123)$$

where $(\mathbf{b}, \mathbf{Y}, \mathbf{t})$ is the mixed product of the vectors involved. The vectors and scalars that enter the preceding field equation are

$$\begin{aligned} \mathbf{R} &= \mathbf{r} - \mathbf{r}' & L &= \mathbf{R} \cdot \mathbf{t} & \rho &= \mathbf{R} - L\mathbf{t} \\ \mathbf{Y} &= \mathbf{R} + |\mathbf{R}|\mathbf{t} = (L + |\mathbf{R}|)\mathbf{t} + \rho \end{aligned} \quad (8.124)$$

where \mathbf{R} is the spacing between the point in the middle of the segment, \mathbf{r}' , and the considered field point, \mathbf{r} , and ρ is the portion of \mathbf{R} normal to the dislocation line \mathbf{t} . These various vectors and scalars are shown in Figure 8-3. The stress of the segment between A and B is then computed by

$$\sigma_{ij}^{A \rightarrow B}(\mathbf{r}) = [\sigma_{ij}(\mathbf{r})]_{\mathbf{r}'=B} - [\sigma_{ij}(\mathbf{r})]_{\mathbf{r}'=A} \quad (8.125)$$

8.3.5 Three-Dimensional Field Equations for Dislocation Segments in an Anisotropic Linear Elastic Medium

In the present approach the dislocation segments are outside their cores (inner cutoff at $\approx |\mathbf{b}|$ where \mathbf{b} is the Burgers vector) described as linear defects that are embedded within an otherwise homogeneous, linear elastic, anisotropic medium in static equilibrium having an arbitrary direction in a 3D space. As in the case of linear elasticity, each dislocation consists of piecewise

² The tedious calculation of the derivatives $q_{,ijk}$ was reviewed by de Wit [de 67] and Devincere [Dev95].

straight segments with a scaling length much smaller than the length of the original dislocation line to be described (scaling length of the segments $\approx |b|$). The stress field associated with a polygonal dislocation loop is obtained by summing over the stress contributions of all segments [BBS79a]. The mathematical problem of describing stress fields of arbitrarily shaped 3D dislocations is thus reduced to the determination of the 3D stress field of a single dislocation segment.

The fundamental theorem from which to start relates the field of an arbitrary planar dislocation to that of an infinite straight dislocation line. Figure 8-4 shows a planar dislocation loop L containing a field point P . The angles α and θ are measured anticlockwise from a fixed coplanar reference datum to the unit vector \mathbf{t} , which is tangential to an elemental arc ds on the dislocation loop L , and to the vector \mathbf{x} , which points from the arc to the field point P , respectively [Bro67, BBS79b]. The stress field at P is given by

$$\sigma_{ij} = \frac{1}{2} \oint_L \frac{1}{r^2} \left[\Sigma_{ij} + \frac{d^2 \Sigma_{ij}}{d\theta^2} \right] \sin(\theta - \alpha) ds \quad (8.126)$$

where Σ_{ij} is the angular stress factor and r the distance between the arc segment and the field point. This tensor expresses the angular dependence in the field which is associated with an infinite, straight dislocation line with the same Burgers vector as the loop L . It must be emphasized that, according to Brown [Bro67], its dislocation line points from the considered arc to the field point, that is, it is tangent to \mathbf{x} rather than to the loop portion ds . Equation (8.126), which is referred to as Brown's theorem, holds for the displacement field as well. Its use reduces the solution for a finite dislocation arc ds to the calculation of the field of an infinite straight dislocation which is characterized by Σ_{ij} .

Integration of equation (8.126) between points A and B gives the stress field contribution of a straight dislocation segment as a function of its Euclidean distance d from the field point [equation (8.127); Figure 8-5]. The angular stress factors and their angular derivatives in equation (8.127) then refer to the two infinite straight dislocations, which point from the start and the end of the segment toward the field coordinate (Figure 8-5).

$$\sigma_{ij} = \frac{1}{2d} \left[-\Sigma_{ij} \cos(\theta - \alpha) + \frac{d\Sigma_{ij}}{d\theta} \sin(\theta - \alpha) \right]_{\theta_1}^{\theta_2} \quad (8.127)$$

Since the two auxiliary dislocation lines that limit the segment are infinite and intersect at the field point, they construct two segments rather than one, the second being generated from the first through a point-mirror operation.

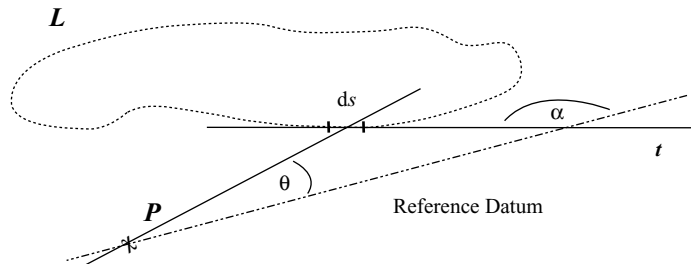


FIGURE 8-4 Geometry for the definition of the stress field at the field point P due to an arbitrary planar dislocation.

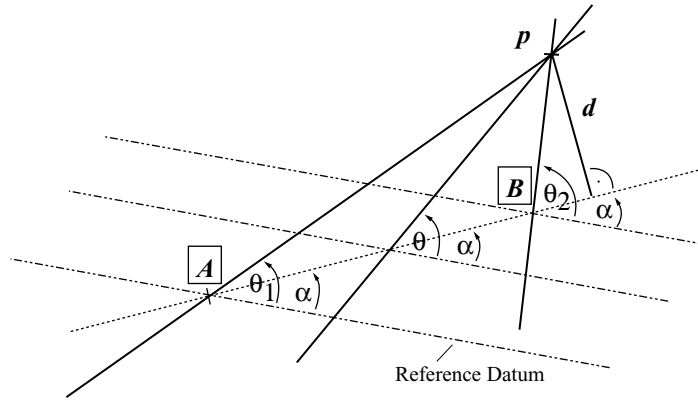


FIGURE 8-5 Definition of a straight dislocation segment $A \rightarrow B$.

Using equation (8.127) as a starting point, Asaro and Barnett [AB74] have proposed a method to transform the 3D calculation of the segment field contributions to a set of piecewise planar problems. Each planar field calculation can then be carried out by applying the integral formalism [Mur63, Bar72, BBS79a, IL92] or the sextic approach [ERS53, Str58]. The former method involves the integration of the Green's function of anisotropic elasticity. The latter approach is based on solving the equilibrium equations under appropriate boundary conditions, which leads to a 2D characteristic equation of which the eigenvalues are complex and occur in conjugate pairs.

In this study the stress fields of the infinite dislocations are derived by integration of the time-independent Green's tensor field $G_{ij}(\mathbf{x}, \mathbf{x}')$, which gives the displacement along the x_i -axis at \mathbf{x} in response to a delta-type unit point force exerted parallel to the x_j -axis at \mathbf{x}' . This approach is valid for an infinite body with homogeneous elastic properties in static equilibrium. In the present case the Green's tensor satisfies the conditions of translational invariance, centrosymmetry, and reciprocity [BBS79a], equation (8.70). Provided that the dislocation motion is uniform, the time-independent Green's tensor may be used for dislocation dynamics as well. In the case of nonuniform motion the time-dependent Green's tensor must be employed. The integral approach provides two advantages as compared with the sextic method. First, it is directly applicable to crystal defects other than dislocations. Second, the integral solutions pass into the isotropic limit where the Zener ratio is equal to 1, that is, $C_{2323} = \frac{1}{2}(C_{1111} - C_{1122})$ for arbitrary values of Poisson's ratio, $C_{1122}/(C_{1111} + C_{1122})$. The main shortcoming of this method is the required numerical integration of the Green's tensor, which is more time-consuming than the solution of the eigenvalue problem in the sextic approach [Bar96].³ Following Asaro and Barnett [AB74], one first has to define the local coordinate system of the infinite dislocations, Figure 8-6. The normal \mathbf{n} to the plane which contains the infinite dislocations, the segment line, and the field point can be expressed as the outer product of any pair of the vectors involved. The unit vector parallel to the infinite dislocation line \mathbf{t} is described by two unit vectors \mathbf{e} and \mathbf{a} normal to \mathbf{n} . The unit vector \mathbf{m} is the angular derivative of \mathbf{t} [AHBL73, AB74].

$$\mathbf{t} = \mathbf{e} \cos(\theta) + \mathbf{a} \sin(\theta) \quad \mathbf{m} = \frac{d\mathbf{t}}{d\theta} = -\mathbf{e} \sin(\theta) + \mathbf{a} \cos(\theta) \quad (8.128)$$

³ Numerical integrations that appear in large vector loops, especially, degrade the speed of dislocation calculations considerably.

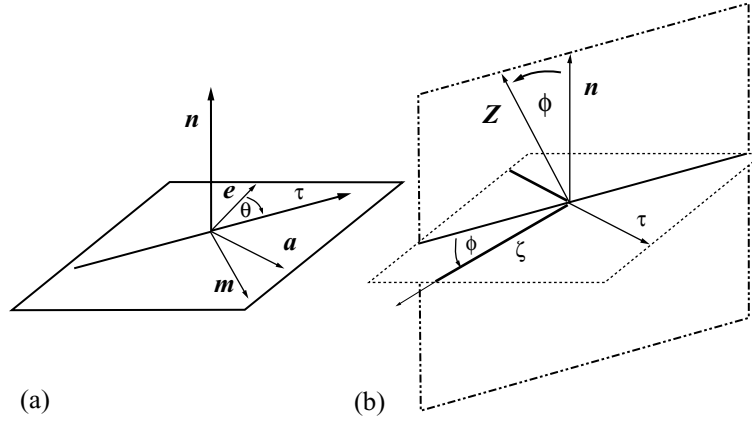


FIGURE 8-6 (a) Dislocation coordinate system, (b) integration variables and their relation to the dislocation coordinate system [AB74].

Since m is a unit vector and located in the same plane as the infinite dislocation line, it can be computed as the vector product of n and t . Employing the Radon transform for the development of the single line integral solution for $G_{ij}(\mathbf{x} - \mathbf{x}')$ leads to the compact tensorial expression:

$$G_{ij}(\mathbf{x} - \mathbf{x}') = \frac{1}{8\pi^2 |\mathbf{x} - \mathbf{x}'|} \oint_{|z|=1} (zz)_{ij}^{-1} ds \quad (8.129)$$

The variable z is a unit vector normal to t , Figure 8-6, defined by

$$z = n \cos(\phi) - m \sin(\phi) \quad (8.130)$$

The normal to the plane described by the unit integration contour, $|z| = 1$, equation (8.129), is thus parallel to t . The second-rank symmetric matrix integrand $(zz)_{ij}^{-1}$ is the inverse of the Christoffel stiffness matrix $(zz)_{ij}$ [BAG⁺72]. The latter operator is for the general nonsymmetric case, $(ab)_{jk}$, defined by equation (8.75). The inverse of the symmetric stiffness matrix with the form $(aa)_{ij}^{-1}$ is given by equation (8.76). The Einstein summation convention is used throughout the calculations. For skipping redundant integrations the symmetries of equations (8.75) and (8.76) should be exploited. For media with cubic lattice symmetry one can use simplified expressions for the inverse stiffness matrix [BAG⁺72]:

$$(aa)_{11}^{-1} = \frac{e(e+f) - ef a_1^2 + (f^2 - 1)(a_2 a_3)^2}{(C_{1122} + C_{2323}) \Delta} \quad (8.131)$$

$$(aa)_{12}^{-1} = - \frac{(a_1 a_2) [(f-1)a_3^2 + e]}{(C_{1122} + C_{2323}) \Delta} \quad (8.132)$$

$$\Delta = e^2(e+f) + e(f^2 - 1) [(a_1 a_2)^2 + (a_1 a_3)^2 + (a_2 a_3)^2] + (f-1)^2(f+2)(a_1 a_2 a_3)^2 \quad (8.133)$$

$$e = \frac{C_{2323}}{C_{1122} + C_{2323}} \quad f = \frac{C_{1111} - C_{2323}}{C_{1122} + C_{2323}} \quad (8.134)$$

The remaining elements $(\mathbf{a}\mathbf{a})_{ij}^{-1}$ are derived by cyclic permutation of the indices of \mathbf{a} . The two matrices, equations (8.75) and (8.76), are related through

$$(\mathbf{a}\mathbf{a})_{ij}^{-1} (\mathbf{a}\mathbf{a})_{jk} = \delta_{ik} \quad (8.135)$$

where δ_{ik} is the Kronecker symbol. The elements of the fourth-ranked tensor of the elastic constants C_{ijkl} are throughout expressed in crystal coordinates. The orientation-dependent part of the Green's tensor, equation (8.129), is given by Q_{ij} ,

$$Q_{ij} = -\frac{1}{2\pi} \int_0^{2\pi} (\mathbf{z}\mathbf{z})_{ij}^{-1} d\phi \quad (8.136)$$

Equation (8.129) can then be written

$$G_{ij}(\mathbf{x} - \mathbf{x}') = -\frac{1}{4\pi |\mathbf{x} - \mathbf{x}'|} Q_{ij} \quad (8.137)$$

Additionally, one requires the nonsymmetric transposed matrix S_{ij}^T , which is defined by

$$S_{ij}^T = \frac{1}{2\pi} \int_0^{2\pi} [(\boldsymbol{\zeta}\mathbf{z}) (\mathbf{z}\mathbf{z})^{-1}]_{ij} d\phi \quad (8.138)$$

The integration variable $\boldsymbol{\zeta}$ is the angular derivative of \mathbf{z} (Figure 8-6).

$$\boldsymbol{\zeta} = \frac{d\mathbf{z}}{d\theta} = -\mathbf{n} \sin(\phi) - \mathbf{m} \cos(\phi) \quad (8.139)$$

The Christoffel stiffness matrix associated with the unit vectors $\boldsymbol{\zeta}$ and \mathbf{z} can be derived as shown in equation (8.75).

However, the integrals Q_{ij} and S_{ij}^T are not independent of each other. Instead of using equation (8.138) the matrix S_{ij}^T can be computed algebraically if Q_{ij} is known also, and vice versa. Kircher and Lothe [KL86] showed that

$$V_s X_{st} = d_t \quad (8.140)$$

with

$$X_{st} = \epsilon_{ilt} [\epsilon_{ijs} Q_{jk}^{-1} (\mathbf{n}\mathbf{n})_{kl}^{-1} + (\mathbf{n}\mathbf{n})_{ij}^{-1} Q_{jk}^{-1} \epsilon_{ijs}] \quad (8.141)$$

and

$$d_t = [(\mathbf{n}\mathbf{n})^{-1} \cdot (\mathbf{n}\mathbf{m}) \cdot \mathbf{Q} - \mathbf{Q} \cdot (\mathbf{m}\mathbf{n}) \cdot (\mathbf{n}\mathbf{n})^{-1}]_{il} \epsilon_{ilt} \quad (8.142)$$

After solving equation (8.140) the matrix \mathbf{S} and its transposed \mathbf{S}^T are obtained by using

$$(\mathbf{S}\mathbf{Q})_{ij} = \epsilon_{ijs} V_s \quad \text{and} \quad \mathbf{S} = (\mathbf{S}\mathbf{Q}) \mathbf{Q}^{-1} \quad (8.143)$$

In the examples given later in this chapter, Q_{ij} is mostly computed by integration, equation (8.136), and S_{ij}^T algebraically by equations (8.140)–(8.143) because it has a more complicated

integrand than Q_{ij} . Finally, the angular stress factor of the infinite dislocation can be calculated according to

$$\Sigma_{gh} = r C_{ghip} u_{i,p} \quad (8.144)$$

where $u_{i,p}$ is the displacement gradient tensor.

$$\Sigma_{gh} = -\frac{1}{2\pi} \epsilon_{pjw} C_{ghip} C_{mwr s} b_m t_j \left(m_s Q_{ir} + n_s \left\{ -(\mathbf{nn})_{il}^{-1} S_{lr}^T - [(\mathbf{nn})^{-1}(\mathbf{nm})]_{il} Q_{lr} \right\} \right) \quad (8.145)$$

The angular derivative of the stress factor is given by

$$\frac{d\Sigma_{gh}}{d\theta} = -\frac{1}{2\pi} \epsilon_{pjw} C_{ghip} C_{mwr s} b_m \left(m_j F_{sir} + t_j \frac{dF_{sir}}{d\theta} \right) \quad (8.146)$$

where

$$F_{sir} = m_s Q_{ir} - n_s (\mathbf{nn})_{il}^{-1} S_{lr}^T - n_s [(\mathbf{nn})^{-1}(\mathbf{nm})]_{il} Q_{lr} \quad (8.147)$$

$$\frac{dF_{sir}}{d\theta} = m_s \frac{dQ_{ir}}{d\theta} - t_s Q_{ir} - n_s (\mathbf{nn})_{il}^{-1} \frac{dS_{lr}^T}{d\theta} + n_s \left\{ [(\mathbf{nn})^{-1}(\mathbf{nt})]_{il} Q_{lr} - n_s [(\mathbf{nn})^{-1}(\mathbf{nm})]_{il} \frac{dQ_{lr}}{d\theta} \right\} \quad (8.148)$$

The angular derivatives of Q_{ij} and S_{ij}^T are given by

$$\frac{dQ_{ij}}{d\theta} = -\frac{1}{2\pi} \int_0^{2\pi} (\mathbf{zz})_{is} [(\boldsymbol{\eta}\mathbf{z}) + (\mathbf{z}\boldsymbol{\eta})]_{sr} (\mathbf{zz})_{rj}^{-1} d\phi \quad (8.149)$$

$$\frac{dS_{ij}^T}{d\theta} = \frac{1}{2\pi} \int_0^{2\pi} \left\{ [(\boldsymbol{\beta}\mathbf{z}) + (\boldsymbol{\zeta}\boldsymbol{\eta})]_{is} (\mathbf{zz})_{sj}^{-1} + [(\boldsymbol{\zeta}\mathbf{z})(\mathbf{zz})^{-1}]_{is} [(\boldsymbol{\eta}\mathbf{z}) + (\mathbf{z}\boldsymbol{\eta})]_{sr} (\mathbf{zz})_{rj}^{-1} \right\} d\phi \quad (8.150)$$

The integrands contain the unit vectors $\boldsymbol{\eta}$ and $\boldsymbol{\beta}$, which are defined by

$$\boldsymbol{\eta} = \mathbf{t} \sin(\phi) = \frac{d\mathbf{z}}{d\theta} \quad \boldsymbol{\beta} = \mathbf{t} \cos(\phi) = \frac{d\boldsymbol{\zeta}}{d\theta} \quad (8.151)$$

For closed-loop or infinite-line dislocation configurations the contribution of the first angular derivative $d\Sigma_{ij}/d\theta$ to the stress field may be dropped [BBS79b] so that equation (8.127) can be reduced to

$$\sigma_{ij} = \frac{1}{2d} [-\Sigma_{ij} \cos(\theta - \alpha)]_{\theta_1}^{\theta_2} \quad (8.152)$$

For computing dislocation fields in cubic media, the following relations among the elastic constants [equations (8.28) and (8.31)] apply:

$$C_{2222} = C_{3333} = C_{1111} \quad C_{1212} = C_{1313} = C_{2323} \quad C_{1122} = C_{1133} = C_{2233}$$

$$C_{ijkl} = C_{jikl} = C_{ijlk} = C_{klij} \quad (8.153)$$

with all remaining elements being equal to zero.

8.4 Dislocation Dynamics

8.4.1 Introduction

The earliest phenomenological models to describe the dynamics of dislocations in crystals date back to the pioneering work of Orowan [Oro34], Polanyi [Pol34], and Taylor [Tay34] in which the motion of dislocations was already understood as an intrinsic defect property. Orowan [Oro34] stated that dislocation glide consists in the movement of dislocations across their respective glide planes. Polanyi [Pol34] complemented this view by adding that dislocation glide across glide planes should take place at stresses much below the value predicted by the theoretical strength that is required to slide one lattice plane rigidly across the other. Taylor [Tay34] studied the atomic positions in the core of what is today referred to as edge dislocation, and the respective positions change during dislocation motion. Common to all these early contributions was the belief that dislocations should have a high mobility, at least in closely packed crystal structures.

The next generation of researchers elaborated these concepts in more detail by identifying two relevant scaling quantities of plasticity, namely, the maximum intrinsic lattice resistance to dislocation motion which is referred to as Peierls⁴ stress σ^P [Pei40, Nab47a] and the “relativistic” increase in the dislocation energy in the vicinity of the speed of sound [Esh49a, Fra49].

In the next step a number of contributions addressed the highly dissipative character of dislocation motion by identifying various drag forces, which act proportionally to the dislocation velocity [Esh49b, Lei50, Nab51, RHBj68]. At the same time, the theory of dislocation motion was complemented by considering inertia [GL56, GLST64, GK70] and by the advent of the concept of thermal activation [See54, Fri56, Lei57, SDMR57, See57, Krö65].

Detailed contributions to these basic aspects associated with the kinetics and thermodynamics of crystal dislocations and to the historical background were published [Fri56, Cot64, HL68, KAA75, ABHH80, Sch80, Suz91, Mug93].

Following the large body of basic work on dislocations it may be assumed that the most general approach to the prediction of the dynamics of lattice dislocations consists in solving Newton’s equation of motion discretely in time and space for each dislocation portion, with consideration of all internal and external forces acting on it. This concept can be realized either through the use of molecular dynamics at the atomic scale or space-discretized continuum dislocation dynamics at the micro- and mesoscale.

This section is concerned with an introduction to the latter technique, that is, to continuum dislocation dynamics. It presents the basic conceptual ingredients required in describing the dynamics associated with 2D and 3D dislocation arrays in a space- and time-discretized fashion. The section is organized in such a way that it presents at first a general Newtonian concept for the description of dislocation dynamics and subsequently simpler, more phenomenological viscous forms.

⁴ In all that follows, σ is used as the symbol for stresses instead of τ . The introduction of τ to indicate shear stresses is not necessary since this can be expressed through σ_{ij} where $i \neq j$.

8.4.2 Newtonian Dislocation Dynamics

Introduction

The general approach outlined in this section idealizes the crystal as a canonical ensemble in the quasi-harmonic approximation [Kra40, Sch80, RSP93]. This allows one to consider anharmonic effects such as the temperature and pressure dependence of the elastic constants and at the same time to treat the crystal in the continuum approach using a linear relation between stress and strain. The dislocations, outside their cores, can then be approximated as line defects that are embedded in a homogeneous, unbounded, linear, anisotropic, elastic medium [Vol07, ERS53, Ste73, BBS79a, HL82, Teo82, Mur87].

The dislocations are regarded as the elementary carriers of velocity fields from the gradients of which both the strain rate and the spin can be calculated. Each dislocation is described as a line defect which consists of a sequence of piecewise straight segments. The displacement and stress fields associated with these segments can be generally formulated to consider elastic anisotropy and arbitrary crystal symmetry using Brown's theorem and the integral formalism in the version of Asaro and Barnett. The stress field of each dislocation line is computed through a linear superposition of the stress contributions of all segments.

Time t and space x_i are independent variables. The atomic concentration of the vacancies c and the displacement field u_i from which the displacement gradient tensor $u_{i,j}$ can be calculated are used as space- and time-dependent state variables.⁵ Hooke's law of anisotropic elasticity and the chemical potential of the vacancies act as equations of state. Newton's law of motion and Fick's modified second law of diffusion are used as structural evolution equations assuming local mechanical equilibrium at each segment. The differential equations of motion of the individual portions belonging to the same dislocation are coupled through the line tension, which is considered discretely by calculating the self-interaction force among the segments according to the concept of Brown [Bro67]. The chemical back-driving force associated with nonconservative dislocation motion is introduced by considering the osmotic pressure that arises from emitting or adsorbing point defects in the climbing segment.

The temperature acts as a state variable which is dependent on time but independent of space, that is, it remains constant throughout the simulation box during each strain increment. This approach reflects the fact that the crystal is regarded as a canonical ensemble where each dislocation segment is embedded in an infinite heat reservoir. The dissipation of heat due to friction is assumed to be much faster than the glide velocity of the dislocation. This postulation is justified because the dissipation rate of heat is determined by the velocity of the electrons at the Fermi level while the glide velocity can only assume values below the speed of sound. In the equation of motion the temperature is included through a stochastic Langevin force as proposed by Leibfried [Lei57] and Rönnpagel *et al.* [RSP93].

Further forces such as phonon drag, electron drag, and the elastic Peach–Koehler interaction among dislocations are considered following earlier classical work [RHBj68, SdB70, Nad88, Nab89, Suz91].

This discrete approach can be referred to as a hybrid model, since it considers both the dynamics of the dislocations and the kinetics of the point defects that are generated by non-conservative dislocation motion. Simulations in this framework involve the solution of a set of coupled stochastic, nonlinear, second-order, partial differential equations for each single dislocation line discretely in space and time by using finite difference or finite element algorithms.

⁵ Tensor quantities of all orders are indicated by bold symbols or indices. Partial derivatives are abbreviated using a comma, for example, $u_{i,j} = \partial u_i / \partial x_j$.

Segmentation of Dislocations

Fundamentals Each dislocation line can be approximated by a sequence of interconnected piecewise straight dislocation segments in three dimensions [Yof60, Hok63, Li65]. Owing to the fact that dislocation lines must not end within an otherwise perfect region of crystal, their segmentation is only allowed if the segments stick together to form closed loops, are semi-infinite, or terminate at a free surface with consideration of image forces, a reaction product, a grain boundary, or some other lattice defect, where stress equilibrium is preserved. This means in the general 3D case each dislocation line consists of j sequentially arranged segments defined by $j + 1$ vectors.

Bacon *et al.* [BBS79a] elegantly justified the concept of decomposing dislocations into segments. They showed that a curvilinear dislocation line can be approximated through a seamless combination of angular dislocation loops with identical Burgers vectors and piecewise alternating tangential vectors. While the non-antiparallel portions of these loops align to form a continuous dislocation line, the antiparallel portions can be arranged in a manner to mutually annihilate each other (Figure 8-2).

This construction shows that a segmented dislocation cannot terminate within a perfect crystal portion because the underlying loops, the non-antiparallel parts of which generate the dislocation under consideration, continue in the crystal. It is thus obvious that the introduction of a geometrical cutoff at the end points of a dislocation entails an error that amounts to the stress field imposed by the remaining semi-infinite loop portions. This error vanishes if the loop approximation is closed or if image forces are considered [Yof60, Hok63, Li65, Bro67, IL92].

Orientation of Segments Two approaches are conceivable to arranging the segment tangential vector with respect to the crystal lattice. In the method suggested by Kubin *et al.* [KCC⁺92], the individual dislocation portions are aligned parallel to potential valleys with a low crystal index so as to either assume pure edge- or pure screw-type character. This technique maps the abrupt- or hard-kink model, where the influence of the Peierls force on the dislocation shape exceeds that of the self-interaction force. In the method suggested by Bacon *et al.* [BKS73], Mohles and Rönnpagel [MR96], and Raabe [Raa98], the dislocation portions can have arbitrary orientation and mixed character. This approach reflects the smooth-kink model, which implies that the Peierls force is of less influence on the dislocation shape than the self-interaction force. While the abrupt-kink model is in accord with experimental observations and theoretical predictions for crystals with covalent or ionic bonding, the smooth-kink approximation applies for crystals with metallic bonding.

Length of Segments Discrete dislocation dynamics simulations which are based on the linear elastic line approximation are at first sight not intrinsically scaled as, for instance, is apparent in molecular dynamics. However, some physical arguments can be identified, resulting essentially from dynamic aspects, which allow one to define an adequate length scale for such simulations. These are the reaction/interaction, the stress fluctuation, the bow-out, the linear elastic-limit, and the cutting criteria.

The reaction/interaction criterion means that the time that elapses during one simulation step is confined to a value prescribed by the condition that dislocation segments must not pass each other without being given a possibility to interact or react. For a given law of motion, say viscous glide, $v \propto \tau$, the half-minimum segment spacing $\Delta\lambda_{\min}^{\text{seg}}$ and the local stress τ dynamically determine the real time that is mapped by one calculation step according to $\Delta t^{\text{step}} \propto \Delta\lambda_{\min}^{\text{seg}}/2\tau$. Choosing this approach for the determination of the real time increment ensures that neighboring dislocation segments cannot pass each other within one time step with-

out an interaction or reaction. After the determination of the time increment, all dislocations are moved according to their respective local velocity.

The stress fluctuation criterion considers that changes in the local force acting on a given straight segment should not exceed a critical value that would entail significantly different velocities of neighboring segment portions. Fulfilling this condition seems difficult, at least from a continuous point of view. It is obvious that it is nearly impossible to avoid a velocity gradient on one segment within a complex dislocation array, once the friction force is reached. Thus, it is necessary to transform this continuum problem into a discrete one. This can be attained by conducting the simulation in such a way that the stress fluctuations on single segments do not lead to different velocities within a chosen discrete velocity or, respectively, stress spectrum.

The bow-out criterion means that the discretization of space should permit simulation of the activation of a Frank–Read source or of the Orowan mechanism on a realistic scale. Typical data are available from the literature [Mug93].

The linear elastic-limit principle means that the elastic distortions should be sufficiently low to justify solutions in the framework of Hooke’s law. For instance, scaling values below the magnitude of the Burgers vector would spoil the validity of approximate solutions derived in the framework of linear elasticity.

The cutting criterion means that intersecting segments mutually increase their total dislocation length by exactly one Burgers vector. In simulations where cutting represents a prevailing mechanism, the scaling parameter should thus amount to one Burgers vector.

Further physical scaling limits can be found from experimental data about kink pair width, double-kink pair extension, jog height, and single-kink width. Owing to these constraints 3D simulations involving dislocation segments are usually scaled by values of one to ten Burgers vectors or lattice parameters, respectively. In order to avoid scaling parameters that are too small, the introduction of a dynamical segment length is conceivable.

The Mechanical Equation of State

The basic problem encountered in space-discretized dislocation statics consists in calculating the local displacement gradient tensor $u_{i,j}(\mathbf{x})$ for a given spatial distribution of dislocations. Discretizing such arrays into sequentially arranged segments reduces the problem to the determination of the displacement gradient field associated with an individual straight portion of dislocation. From this quantity the local stress field can be derived using Hooke’s law.

Calculating the stresses in dyadic form allows one to rotate the contributions of all segments to a common reference lattice and to derive the net field quantity through a linear superposition [Li65].

The Mechanical Structure Evolution Equation

The Equation of Motion A general concept for the dynamics of dislocations can be formulated by assuming dynamic equilibrium of the forces at each time and portion along the dislocation line and by solving a modified form of Newton’s equation of motion for each such dislocation segment. This approach, which was introduced by Rönnpagel [Rön87] for the 2D case, can in principle be extended to 3D problems [Raa98].

$$\sum_i \mathbf{F}^i(t, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}) = 0 \quad (8.154)$$

where \mathbf{x} and t are the independent variables, $\dot{\mathbf{x}}$ the velocity, $\ddot{\mathbf{x}}$ the acceleration, and \mathbf{F}^i the various line forces that act on each portion of dislocation. The forces that enter equation (8.154) can be described as a function of the independent variables and their partial derivatives. The most important contributions in 3D simulations are the Peierls force, \mathbf{F}^P , the dislocation–dislocation

force, F^{disl} , the self force, F^{self} , the external force, F^{ext} , the obstacle force, F^{obs} , the image force, F^{ima} , the osmotic force, F^{osm} , the phonon drag force, F^{Pdrag} , the electron drag force, F^{Edrag} , the inertia force, F^{iner} , the cross-slip drag force, F^{cross} , and the thermal force, F^{therm} .

The Peierls Force The Peierls force F^{P} is a constant for the slip system being considered and represents the maximum intrinsic resistance to dislocation motion at 0 K. While the Peierls potential is very small in face-centered cubic metals, it can be of substantial relevance in body-centered metals, semiconductors, covalent, and ionic materials. The use of the Peierls concept in the case of 3D dislocation dynamics is somewhat diffuse because it originally denotes the force to bring an infinite rather than a finite dislocation from one potential valley to the next.

The Dislocation–Dislocation Force The dislocation–dislocation term F^{disl} represents the sum of all Peach–Koehler forces between the segment considered and *all* other dislocation portions in the ensemble. However, it must be underlined that this term thus does *not* include any interactions with dislocation segments that belong to the *same* dislocation. The spacing between the inspected portion of dislocation and the segments that belong to other dislocations usually exceeds the inner cutoff radius by one order of magnitude, except for dislocation reactions. Thus, this value enters the differential equation of motion at each time step as a constant. Since the character of the interactions is long-range in nature, the predictions obtained by dislocation dynamics simulations are highly sensitive to the introduction of an outer cutoff radius. A more recent paper substantiated the fact that the use of an insufficiently small outer cutoff radius can entail incorrect predictions of the internal stress fields [Raa96a]. For obtaining reliable results it is thus necessary to impose periodic boundary conditions or to account for all dislocation segments in the array under consideration of the respective image forces imposed by the free surface.

The Self Force The self force F^{self} comprises those contributions to the total force that arise from the elastic Peach–Koehler interaction between the inspected line segment and all other dislocation portions that belong to the *same* dislocation. Although the physical nature of this line force is exactly identical to the aforementioned dislocation–dislocation force, their main difference lies in the fact that the self force *directly* enters the differential equation of motion as a variable rather than as a constant. The self force vanishes for infinite straight dislocations. However, for curved dislocation lines its contribution has substantial influence on the dynamics.

For sufficiently small dislocation curvatures, the self force can be expressed in terms of the curvature-dependent portion of the line tension. In this concept the dislocation is regarded as a taut elastic string [Cot53, Str54, dK59]. The line tension mimics the back-driving force of the dislocation in reducing the core and elastic energy that arises from any increase in dislocation length in the bowed-out state as compared with the straight state. An analytical treatment of the line tension predicts a back driving force that depends on the increase in the core energy and on the local curvature. However, this classical concept neglects the dependence of the dislocation energy on the dislocation arrangement [HL82]. For large curvatures this simple analytical approximation of the line tension is no longer valid and must be replaced by more detailed energy calculations that account for the overlap of the displacement fields of neighboring portions of the same dislocation. In a thorough analysis Brown [Bro67] showed the consistency between the generalized line tension and the self force, which can be computed through the Peach–Koehler interaction among sequentially arranged segments belonging to the same defect.

The introduction of the self force turns equation (8.154) into a nonlinear differential equation. Furthermore, the hyperbolic long-range force acts as a coupling term between the partial differential equations that describe the motion of the individual segments.

The External and the Obstacle Force The externally imposed force \mathbf{F}^{ext} enters the solution of the differential equation as a constant Peach–Koehler force on each portion of all dislocations in the array. Further Peach–Koehler-type contributions can arise from lattice defects other than dislocations. Among these, particularly the contributions from obstacles \mathbf{F}^{obs} are relevant. Obstacles give rise to elastic interactions, such as are imposed by coherent precipitations or foreign atoms with a parestic and dielastic matrix coupling [Esh57, Mur87]. Details of the incorporation of obstacles in time- and space-discretized dislocation simulations have been thoroughly investigated by Rönnpagel [Rön87], Mohles [Moh96], and Mohles and Rönnpagel [MR96].

The Image Force Another term arises from stress relaxation in the vicinity of a free surface or internal interface. Such contributions are referred to as surface or image forces \mathbf{F}^{ima} . They result from the necessity that a free surface must be in the stress-free state. Image effects are modeled by including surface stresses that exactly compensate the stresses imposed by lattice defects in the interface. Thus, they generate surface forces that act attractively on each portion of dislocation.

The incorporation of image forces is straightforward in 2D systems, involving only screw dislocations or simple edge dislocation arrangements. In these cases a surface stress compensation is achieved by appropriate mirror operations carried out on each infinite dislocation line with the surface acting as a mirror plane (Figure 8-7). In 3D, geometrical difficulties in treating these forces arise from the fact that the individual dislocation portions are in most cases no longer straight, infinite, or parallel to the free surface.

According to van der Giessen and Needleman [vN95] this 3D boundary-value problem can be solved by using a finite element approach. On the other hand, Gosling and Willis [GW94] have suggested an analytical method to compensate surface stresses. More recently, Fivel *et al.* [FGC96] compared the approach of Gosling and Willis [GW94] with that of Boussinesq [Bou85] and found a good accord between both techniques.

The Osmotic Force Dislocation dynamics shows a fundamental difference from molecular dynamics. While in the latter approach one deals with a conservative system and can thus describe the local force as a negative gradient of the potential, dislocation motion is highly dissipative and reveals a strong anisotropy in its kinetic modes, that is, the existence of glide and climb. In contrast to 2D simulations, which are confined to one slip plane, the extension of discrete dislocation simulations to three dimensions implies the occurrence of climb [Raa98]. However, nonconservative motion of edge dislocation segments generally occurs by the generation of intrinsic point defects. This leads to an under- or oversaturation of such defects, that

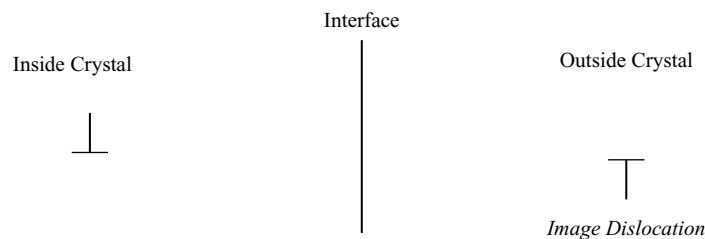


FIGURE 8-7 Construction for the calculation of the image force in two dimensions in a simple case where the infinite dislocation line is parallel and the Burgers vector perpendicular to the interface.

is, to a deviation from chemical equilibrium. The force that results from the tendency of the system to return to thermal equilibrium is quantified in terms of an osmotic force \mathbf{F}^{osm} . Following the original concept of Nabarro [Nab47b], Bardeen and Herring [BH52], Weertman [Wee65], and Balluffi and Granato [BG79], this contribution can enter the equation of motion through an expression of the form

$$\mathbf{F}^{\text{osm}} = -\frac{\mu}{\Omega} \frac{(\mathbf{b} \times \mathbf{t})}{|\mathbf{b} \times \mathbf{t}|} = -\frac{k_B T}{\Omega} \frac{(\mathbf{b} \times \mathbf{t})}{|\mathbf{b} \times \mathbf{t}|} \ln\left(\frac{c(\mathbf{r})}{c_0}\right) \quad (8.155)$$

where \mathbf{t} is the dislocation tangential vector and μ the chemical potential. This term couples the equation of motion with diffusion.

The Phonon Drag Force It is a characteristic feature of dislocations that they dissipate energy during their motion. This is due to the velocity-dependent electron and phonon drag [Lei50, Nab51, RHBJ68]. The dominant contribution to friction arises from phonon–dislocation interactions $\mathbf{F}^{\text{Pdrag}}$. In this context two mechanisms have been discussed, namely, the nonlinearity mechanism, which is imposed by the anharmonicity of the potential [SE68, Bra72], and the fluttering mechanism [Nin70], which is due to inelastic scattering between the dislocation and a phonon. A comparison of both approaches which provides very similar predictions except at very low temperatures was given by Suzuki [Suz91]. Following the approach of [Nin70], that is, assuming a fluttering mechanism, the temperature-dependent portion of the friction force can be written

$$\begin{aligned} \mathbf{F}^{\text{Pdrag}}(T) &= B_p(T) \dot{\mathbf{x}} = \frac{k_B T \omega_D^2}{\pi^2 c_i^3} \dot{\mathbf{x}} & \text{at } T > 0.5 \Theta \\ \mathbf{F}^{\text{Pdrag}}(T) &= B_p(T) \dot{\mathbf{x}} = \frac{14.4 k_B T \omega_D^2}{\pi^2 c_i^3} \left(\frac{T}{\Theta}\right)^2 \dot{\mathbf{x}} & \text{at } T < 0.5 \Theta \end{aligned} \quad (8.156)$$

where $B_p(T)$ is the temperature-dependent friction coefficient, Θ the Debye temperature, ω_D the Debye frequency, and c_i the transverse or longitudinal velocity of sound, respectively.

The Electron Drag Force The frictional force due to conduction electrons with the coefficient B_e was assessed by Tittman and Bömmel [TB66] using standard perturbation theory, and by Kravchenko [Kra66] and Brailsford [Bra69] using the Boltzmann equation. According to the analysis of these papers given by Suzuki [Suz91] the contribution of electrons to friction is practically independent of the temperature. For edge-type segments the electronic portion of the friction amounts to

$$\begin{aligned} \mathbf{F}^{\text{Edrag}} = B_e \dot{\mathbf{x}} &= \left(\frac{1-2\nu}{1-\nu}\right)^2 \frac{n_0 m_e v_F b^2 q_D}{96} \frac{1}{2} \\ &\quad \left(\frac{1}{1 + \left(\frac{q_D}{q_{\text{TF}}}\right)^2} + \left(\frac{q_{\text{TF}}}{q_D}\right) \tan^{-1}\left(\frac{q_D}{q_{\text{TF}}}\right) \right) \dot{\mathbf{x}} \end{aligned} \quad (8.157)$$

where ν is Poisson's ration, n_0 the equilibrium value of the electron density, m_e the mass of the quasi-free electron, v_F the Fermi velocity, q_D the radius of a Debye sphere, and q_{TF} the reciprocal of the Thomas–Fermi screening length. For screw dislocations the electronic frictional force is negligible.

The Inertia Force When the dislocation movement does not take place in the overdamped regime, that is, at very high applied forces, inertial effects \mathbf{F}^{iner} must be taken into consideration. This contribution renders the equation of motion into a second-order differential equation. The effective mass m_{eff} per length of a dislocation can be calculated according to

$$m_{\text{eff}} = m'_{\text{eff}} \frac{1}{\sqrt{1 - \left(\frac{\dot{x}}{c_i}\right)^2}} \quad m'_{\text{eff}} = \frac{\mu b^2}{2 c_i^2} \quad (8.158)$$

where μ is the shear modulus, c_i the velocity of sound, and m'_{eff} the rest mass of the dislocation. The mass point assigned to a dislocation segment of length $L = |\mathbf{r}'_{\mathbf{A}} - \mathbf{r}'_{\mathbf{B}}|$ then amounts to $\Delta m = m_{\text{eff}} L$.

The Cross-Slip Force While edge dislocations generally have a well-defined slip plane, this is not the case for screw dislocations. A particular glide plane for screw dislocations can only be defined in terms of the present maximum orientation factor, which would force an ideal screw dislocation into a certain plane, or through the occurrence of dissociation of the original complete dislocation into partial dislocations. The tendency to dissociate is inversely weighted by the stacking fault energy of the particular material. Thus, dislocation cross-slip is obstructed by the enthalpy that is required to recombine the dissociated portions of the dislocation. This gives rise to a cross-slip drag force $\mathbf{F}^{\text{cross}}$.

The Thermal Force The contributions discussed previously represent relevant line forces that must be considered for simulating athermal dislocation dynamics. However, in order to allow simulations at temperatures other than 0 K, thermal forces $\mathbf{F}^{\text{therm}}$ must be included in the model. This concept is particularly important for simulating thermal activation processes [Lei57, RSP93, MR96, Moh96].

From a thermodynamic point of view dislocations in thermodynamic equilibrium with the crystal can be regarded as a canonical ensemble, the energy of which fluctuates in time. Thermal fluctuations, which are also referred to as thermal noise, can be described formally in terms of many uncorrelated tiny perturbances, each of which changes the state variables of the system in a small but random fashion. Irrespective of the thermal fluctuations, the macroscopic mean value of the energy is a well-defined function of temperature, which is prescribed by the heat bath, that is, by the temperature of the crystal.

Due to the drag force terms, the moving dislocation continuously dissipates energy as a function of its velocity. Since the dislocation is assumed to be in thermodynamic equilibrium with the crystal, the energy lost through dissipation must be compensated by applying thermal pulses. These pulses on a dislocation can be described in terms of a Langevin force in case of the Brownian motion [RSP93]. It is assumed that the average of this force over the ensemble should vanish, that is,

$$\langle \mathbf{F}^{\text{therm}}(t) \rangle = 0 \quad (8.159)$$

since the equation of motion of the average velocity $\langle \dot{x} \rangle$ does not contain any probabilistic term. When multiplying the Langevin force term at two different times t and t' , one further assumes that the resulting average value vanishes if the time step $(t' - t)$ exceeds the duration time τ^c of a collision of the considered particle with the atoms:

$$\langle \mathbf{F}^{\text{therm}}(t), \mathbf{F}^{\text{therm}}(t') \rangle = 0 \quad \text{where} \quad |t - t'| \geq \tau^c \quad (8.160)$$

This amounts to stating that the Langevin force changes randomly in size and sign without correlation. This assumption is reasonable since it is likely that the collisions of the atoms with the small particle are approximately independent.⁶

Usually, the collision time τ^c is much smaller than the relaxation time of the velocity of the small particle, which amounts to $\tau^r = 1/B(T)$ where $B(T)$ is the friction coefficient. Thus, the approximation $\tau^c \rightarrow 0$ appears reasonable. This leads to the form

$$\langle \mathbf{F}^{\text{therm}}(t), \mathbf{F}^{\text{therm}}(t') \rangle = q \delta(t - t') \quad \text{where} \quad \tau^c \rightarrow 0 \quad (8.161)$$

The Dirac delta function appears in this equation because otherwise the energy of the particle considered cannot be finite as it should be in accord with equipartition law. According to Rönnpagel *et al* [RSP93], the strength of the thermal noise term is given by

$$q = \frac{2 B(T) k_B T}{m_{\text{eff}}^2} \quad (8.162)$$

where m_{eff} is the effective mass per length. Combining equations (8.162) and (8.156) substantiates the fact that the strength of the Langevin force is not simply linearly proportional to the temperature but also temperature-dependent through the phonon-dependent part of $B(T)$. In order to produce such thermal pulses, one must provide Gaussian-distributed random vectors $\Lambda(i, j)$ for each segment i at position $\mathbf{x}(i)$ and time step j :

$$f(\Lambda_k) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\Lambda_k^2}{2}\right) \quad (8.163)$$

where Λ_k is the k th component of the vector Λ . The thermal force $\mathbf{F}^{\text{therm}}(\mathbf{x}(i), t_j \leq t < t_j + \Delta t_j)$ imposed in the time between t_j and $t_j + \Delta t_j$ then amounts to

$$\mathbf{F}^{\text{therm}}(\mathbf{x}(i), t_j \leq t < t_j + \Delta t_j) = \Lambda(i, j) |\mathbf{b}| \sqrt{\frac{2 B(T) k_B T}{\Delta l(i) \Delta t_j b^2}} \quad (8.164)$$

where $\Delta l(i)$ is the length of the segment i [RSP93]. The portions of $\Lambda(i, j)$ that are parallel to the segment tangential vector can be neglected [Moh96].

Local heating of the slip plane is usually not considered. In metals it is generally assumed that the dissipation of heat due to friction is much faster as compared with the glide velocity of the dislocation. After each deformation step the temperature increase of the heat bath can be calculated from the dissipated power.

Solution of the Equation of Motion Incorporating all the previously mentioned contributions in equation (8.154) results, for *each* dislocation consisting of j sequentially arranged segments defined by $j + 1$ vectors, in a set of $3 \cdot j$ coupled, nonlinear, stochastic, partial differential equations of second order [RSP93, RRM96]. Such a set of equations must be solved for appropriate initial-value conditions, that is, for given initial positions \mathbf{x}_0^j and initial velocities $\dot{\mathbf{x}}_0^j$ by using finite difference techniques, for example, a higher-order Runge–Kutta or predictor–corrector method (see methods in Chapter 3).

⁶ It is not necessarily straightforward that this applies also to the interaction with moving dislocations. Furthermore, the phonon spectrum of crystals might be relevant in this context.

8.4.3 Viscous and Viscoplastic Dislocation Dynamics

Introduction

When dislocation simulations are conducted to mimic dynamics in the overdamped high-stress regime, the differential equations that were presented in the preceding section can be simplified. For instance, by dropping the inertia term in equation (8.154) the system can be turned into a set of first-order differential equations that is easier and faster to solve. In a second step, the treatment of thermal activation can be simplified by using a statistical exponential expression [KAA75, WW80, Nad88] in conjunction with one single Monte Carlo step instead of a Langevin term. Furthermore, the discrete consideration of self-interaction can be replaced by a discrete line tension approximation [FGC96].

Most phenomenological simulations have been conducted in two dimensions with infinite straight edge dislocations [LK87, GA89, GSLL89, Amo90, GH92, GH93, LBN93, vN95, WL95, RR96, RRG96, RRM96, ZSL97]. Edge dislocation glide is typically modeled in such approaches as successive periods of resting in front of obstacles and viscous glide between them.

Accounting for both effects, the dislocation velocity component in the direction of glide can be expressed according to

$$\dot{x}_{\text{glide}} = \frac{\lambda}{t_w + t_g} \quad (8.165)$$

where λ is the obstacle spacing, t_w the waiting time *in front of* obstacles, and t_g the time for glide *between* obstacles. When no particles or precipitates are considered, the obstacle spacing λ is determined exclusively by the forest dislocation density. In cases where additional obstacles such as precipitates are included, the effective obstacle spacing can be computed according to

$$\frac{1}{\lambda} = \sum_i \frac{1}{\lambda_i} \quad (8.166)$$

where λ_i is the respective spacing of lattice defects i that interact with moving dislocations. If only forest dislocations are considered with a density proportional to the total dislocation density, one can write

$$\lambda \propto \frac{1}{\sqrt{\varrho}} \quad (8.167)$$

where ϱ is the total dislocation density. Cutting of forest dislocations is a thermally activated process. Therefore, the cutting frequency ν_{cut} can be expressed in the following manner:

$$\begin{aligned} \nu_{\text{cut}} &= 2\nu_0 \exp\left(-\frac{Q}{k_B T}\right) \sinh\left(\frac{\sigma V}{k_B T}\right) = \\ &= 2\nu_0 \exp\left(-\frac{Q}{k_B T}\right) \sinh\left(\frac{F_{\text{glide}} A}{k_B T}\right) \end{aligned} \quad (8.168)$$

where Q is the activation energy, $\sigma V = F_{\text{glide}} A$ the mechanical work that results from the net local stress, and ν_0 the attack frequency. It is typically assumed that the activation area A is proportional to the spacing of the forest dislocations. The \sinh function results from the consideration of both forward and backward jumps.

The time spent resting in front of an obstacle is then simply given by

$$t_w = \frac{1}{\nu_{\text{cut}}} \quad (8.169)$$

Between the obstacles viscous glide is assumed. Hence, the *local* glide velocity for unimpeded motion, \dot{x}_{glide}^u , is proportional to the force F_{glide} , so that t_g can be calculated as

$$t_g = \frac{\lambda}{\dot{x}_{\text{glide}}^u} = \frac{\lambda}{c F_{\text{glide}}} \quad (8.170)$$

where c is a constant. From equations (8.165)–(8.170) the net velocity in the direction of glide can be derived:

$$\dot{x}_{\text{glide}} = \frac{\lambda}{t_w + t_g} = \frac{2 \nu_0 c \lambda F_{\text{glide}} \sinh\left(\frac{F_{\text{glide}} A}{k_B T}\right)}{c F_{\text{glide}} \exp\left(\frac{Q}{k_B T}\right) + 2 \nu_0 \sinh\left(\frac{F_{\text{glide}} A}{k_B T}\right)} \quad (8.171)$$

In contrast to this glide law, the nonconservative climb process is controlled by diffusion. The velocity of dislocation climb \dot{x}_{climb} is thus directly proportional to the respective force component with the diffusion coefficient D as constant, namely,

$$\dot{x}_{\text{climb}} = B F_{\text{climb}} = \frac{D}{k_B T} F_{\text{climb}} = \frac{F_{\text{climb}}}{k_B T} D_0 \exp\left(-\frac{H}{k_B T}\right) \quad (8.172)$$

where H is the activation enthalpy for vacancy formation and migration.

Once the dislocation velocities are determined in accord with the respective local stress fields, it has to be ensured that possible resulting dislocation reactions are considered by the simulation prior to the movement of the dislocations. Therefore, the time increment of each simulation step is chosen dynamically. For this purpose the minimum spacing between two dislocations is determined. The distance traveled by the fastest dislocation can then be limited to half this value. After the time increment has been determined, all dislocations are then moved according to their individual velocities.

It is clear that 2D simulations of edge dislocation dynamics do not account for dislocation multiplication. This shortcoming is due to the fact that all dislocations are parallel. This means that new dislocations have to be introduced by some artificial argument. This can be done by inserting a new mobile dislocation into the system whenever a dislocation is immobilized, or by explicitly relating the production rate to the glide activity in each glide system.

Phenomenological dislocation dynamics simulations are also possible in three dimensions [DC92, Dev96]. In these approaches the dislocation lines are typically subdivided into piecewise straight, sequentially arranged segments of either pure screw- or edge-type character [KCC⁺92]. Thus, in contrast to the 2D simulations, in three dimensions the mobility of the two different segment types must be distinguished.

Following the preceding concept, the glide velocity of an edge segment in kink motion, \dot{x}_{kink} , can be written

$$\dot{x}_{\text{kink}} = \frac{C_k \omega_0 \Lambda F_{\text{kink}} \sinh\left(\frac{F_{\text{kink}} A}{k_B T}\right)}{\frac{F_{\text{kink}}}{B_k} \exp\left(\frac{Q_k}{k_B T}\right) + C_k \Gamma_0 \sinh\left(\frac{F_{\text{kink}} A}{k_B T}\right)} \quad (8.173)$$

where ω_0 is the Debye frequency, Λ the local flight path of the segment, F_{kink} the net glide force acting on the kink segment, Q_k the activation energy, C_k a constant, and B_k the drag coefficient.

For situations where either the waiting or the flying time dominates, simpler phenomenological laws can be used, namely,

$$\dot{x}_{\text{kink}} = \tilde{C}_k \left(\frac{\tau}{\tau_0} \right)^m \exp\left(-\frac{Q_k}{k_B T}\right) \quad (8.174)$$

for the former, and

$$\dot{x}_{\text{kink}} = \frac{\tau b}{\tilde{B}_k} \quad (8.175)$$

for the latter case, where \tilde{C}_k and \tilde{B}_k are constants, τ_0 the reference shear stress,⁷ and m a variable which quantifies the stress dependence of the kink velocity.

For the screw portions similar glide expressions apply, although furnished with somewhat different constants, which reflect the influence of the dislocation core on the segment mobility. However, screw dislocations in 3D simulations can also leave their actual glide plane. For describing this cross-slip probability p in discrete calculations, Fivel *et al.* [FGC96] suggested using an expression of the form

$$p \propto \exp\left(\frac{(\tau_f - \tau_{\text{III}})V}{k_B T}\right) \quad (8.176)$$

where τ_f represents the effective friction stress, τ_{III} the stage-III stress, and V the activation volume of the process. The nonconservative movement of jogs is climb-controlled and can be described in accord with equation (8.172)

$$\dot{x}_{\text{climb}} = B F_{\text{climb}} = \frac{D}{k_B T} F_{\text{climb}} = \frac{F_{\text{climb}}}{k_B T} D_0 \exp\left(-\frac{H}{k_B T}\right) \quad (8.177)$$

where H is the activation enthalpy of self-diffusion in the bulk material. For considering the influence of the local stress on the jog climb rate, equation (8.177) can be rewritten

$$\begin{aligned} \dot{x}'_{\text{climb}} &= \frac{F_{\text{climb}}}{k_B T} D_0 \exp\left(-\frac{H}{k_B T}\right) = \frac{F_{\text{climb}}}{k_B T} D_0 \exp\left(-\frac{(H^f + H^m)}{k_B T}\right) \\ &= \frac{F_{\text{climb}}}{k_B T} D_0 \exp\left(-\frac{H^m}{k_B T}\right) \exp\left(-\frac{(H^f - \sigma^h \Omega)}{k_B T}\right) \end{aligned} \quad (8.178)$$

where Ω is the atomic volume, which can be approximated by $|b|^3$, σ^h the local hydrostatic stress component, H^f the enthalpy of formation of vacancies, and H^m their enthalpy of migration [Fri64, HL68, Suz91].

The influence of osmotic forces which can gradually oppose jog climb is usually neglected in phenomenological dislocation dynamics simulations. The omission of such chemical effects is often justified by the assumption that jog drag-stimulated vacancy fluxes mutually compensate each other by short-range diffusional processes among jogs with opposite sign. The self-interaction of such phenomenological dislocation simulations is typically considered by including the analytical line tension approximation or some numerical criteria derived therefrom [DC92].

⁷ The reference shear stress is not identical to the critical resolved shear stress.

8.5 Kinematics of Discrete Dislocation Dynamics

Dislocation dynamics can be used for the calculation of stress–strain curves and lattice reorientations. By tracing all distances traveled by the individual dislocations during each time step it is straightforward to use the Orowan equation for the calculation of the shear rate during a particular time step $\dot{\gamma}$ according to

$$\dot{\gamma} = \rho_m b \bar{x}_{\text{glide}} \quad (8.179)$$

where ρ_m is the density of mobile dislocations, b the magnitude of their Burgers vector, and \bar{x}_{glide} their average glide velocity.

In the case where N individual dislocations are considered in a 2D simulation, the right-hand side of equation (8.179) can be substituted by a sum over all dislocations. The shear rate in the x_1 -direction can then be calculated by

$$\dot{\gamma}_{x_1} = \frac{1}{A} \sum_i^N b_{x_1}^i \dot{x}_{x_1}^i \quad (8.180)$$

where $b_{x_1}^i$ is the x_1 -component of the Burgers vector of the i th dislocation and $\dot{x}_{x_1}^i$ its velocity in the x_1 -direction during the considered time interval. A is the area of the simulation cell. The shear rate in the x_2 -direction can be calculated analogously, that is,

$$\dot{\gamma}_{x_2} = \frac{1}{A} \sum_i^N b_{x_2}^i \dot{x}_{x_2}^i \quad (8.181)$$

The shear rate $\dot{\epsilon}_{x_1 x_2}$ can be calculated according to

$$\dot{\epsilon}_{x_1 x_2} = \frac{1}{2} (\dot{\gamma}_{x_1} + \dot{\gamma}_{x_2}) \quad (8.182)$$

This leads to a strain increment per simulation step Δt which amounts to $\Delta \epsilon_{x_1 x_2} = \dot{\epsilon}_{x_1 x_2} \Delta t$.

In three dimensions the strain rate tensor during a numerical experiment can be calculated by applying Orowan's equation to *all* segments in the array. Following equation (8.180) the shear rate in the three spatial directions can be calculated by

$$\dot{\gamma}_{x_1} = \frac{1}{V} \sum_i^N b_{x_1}^i \dot{x}_{x_1}^i \Delta s \quad \dot{\gamma}_{x_2} = \frac{1}{V} \sum_i^N b_{x_2}^i \dot{x}_{x_2}^i \Delta s \quad \dot{\gamma}_{x_3} = \frac{1}{V} \sum_i^N b_{x_3}^i \dot{x}_{x_3}^i \Delta s \quad (8.183)$$

where Δs the segment length and V the volume of the simulation box. The normal components of the strain rate tensor are then determined by

$$\dot{\epsilon}_{x_1 x_1} = \dot{\gamma}_{x_1} \quad \dot{\epsilon}_{x_2 x_2} = \dot{\gamma}_{x_2} \quad \dot{\epsilon}_{x_3 x_3} = \dot{\gamma}_{x_3} \quad (8.184)$$

and its shear components by

$$\dot{\epsilon}_{x_1 x_2} = \frac{1}{2} (\dot{\gamma}_{x_1} + \dot{\gamma}_{x_2}) \quad \dot{\epsilon}_{x_1 x_3} = \frac{1}{2} (\dot{\gamma}_{x_1} + \dot{\gamma}_{x_3}) \quad \dot{\epsilon}_{x_2 x_3} = \frac{1}{2} (\dot{\gamma}_{x_2} + \dot{\gamma}_{x_3}) \quad (8.185)$$

The strain increments per simulation step Δt amount to

$$\Delta\varepsilon_{x_1x_1} = \dot{\gamma}_{x_1} \Delta t \quad \Delta\varepsilon_{x_2x_2} = \dot{\gamma}_{x_2} \Delta t \quad \Delta\varepsilon_{x_3x_3} = \dot{\gamma}_{x_3} \Delta t \quad (8.186)$$

and its shear components to

$$\Delta\varepsilon_{x_1x_2} = \frac{\Delta t}{2} (\dot{\gamma}_{x_1} + \dot{\gamma}_{x_2}) \quad \Delta\varepsilon_{x_1x_3} = \frac{\Delta t}{2} (\dot{\gamma}_{x_1} + \dot{\gamma}_{x_3}) \quad \Delta\varepsilon_{x_2x_3} = \frac{\Delta t}{2} (\dot{\gamma}_{x_2} + \dot{\gamma}_{x_3}) \quad (8.187)$$

The rotation rate of the lattice affected, $\dot{\omega}_{x_ix_j}^{\text{latt}}$, which results from the shears on the individual slip systems, can be computed from the rigid-body rotation rate, that is, from the skew symmetric portion of the discretized crystallographic velocity gradient tensor, $\dot{\omega}_{x_ix_j}^{\text{spin}}$, and from the antisymmetric part, $\dot{u}_{x_i,x_j}^{\text{anti}}$, of the externally imposed macroscopic velocity gradient tensor, $\dot{u}_{x_i,x_j}^{\text{ext}}$:

$$\dot{\omega}_{x_ix_j}^{\text{latt}} = \dot{u}_{x_i,x_j}^{\text{anti}} - \dot{\omega}_{x_ix_j}^{\text{spin}} = \frac{1}{2} (\dot{u}_{x_i,x_j}^{\text{ext}} - \dot{u}_{x_j,x_i}^{\text{ext}}) - \frac{1}{2} (\dot{\gamma}_{x_i} - \dot{\gamma}_{x_j}) \quad (8.188)$$

8.6 Dislocation Reactions and Annihilation

In the preceding sections it was mainly long-range interactions between the dislocation segments that were addressed. However, strain hardening and dynamic recovery are essentially determined by short-range interactions, that is, by dislocation reactions and by annihilation, respectively.

Using a phenomenological approach that can be included in continuum-type simulations, one can differentiate between three major groups of short-range hardening dislocation reactions [FBZ80]: the strongest influence on strain hardening is exerted by sessile reaction products such as Lomer–Cottrell locks. The second strongest interaction type is the formation of mobile junctions. The weakest influence is naturally found for the case in which junctions are formed.

Two-dimensional dislocation dynamics simulations usually account for annihilation and the formation of sessile locks. Mobile junctions and the Peach–Koehler interaction occur naturally among parallel dislocations. The annihilation rule is straightforward. If two dislocations on identical glide systems but with opposite Burgers vectors approach more closely than a certain

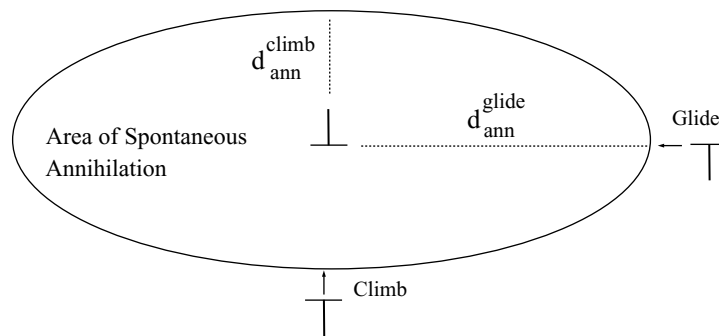


FIGURE 8-8 Annihilation ellipse in 2D dislocation dynamics. It is constructed by the values for the spontaneous annihilation spacing of dislocations that approach by glide and by climb.

minimum allowed spacing, they spontaneously annihilate and are removed from the simulation. Current 2D simulations [RRG96] use different minimum distances in the direction of glide ($d_{\text{ann}}^g \approx 20|b|$) and climb ($d_{\text{ann}}^c \approx 5|b|$), respectively [EM79] (Figure 8-8).

Lock formation takes place when two dislocations on different glide systems react to form a new dislocation with a resulting Burgers vector which is no longer a translation vector of an activated slip system. In the 2D simulation this process can be realized by the immobilization of dislocations on different glide systems when they approach each other too closely (Figure 8-9). The resulting stress fields of the sessile reaction products are usually approximated by a linear superposition of the displacement fields of the original dislocations before the reaction.

Dislocation reactions and the resulting products can also be included in 3D simulations. Due to the larger number of possible reactions, two aspects require special consideration, namely, the magnitude and sign of the latent heat that is associated with a particular reaction, and the kinematic properties and the stress field of the reaction product.

The first point addressed can be solved without using additional analytical equations. For investigating whether a particular reaction between two neighboring segments will take place or not, one subtracts the total elastic and core energy of all initial segments that participate in the reaction from that of the corresponding configuration after the reaction. If the latent heat is negative, the reaction takes place. Otherwise, the segments pass each other without reaction. In face-centered cubic materials 2 dislocations can undergo 24 different types of reactions. From this number only 12 entail sessile reaction products. Assuming simple configurations, that is, only a small number of reacting segments, the corresponding latent heat data can be included in the form of a reference table.

The short-range back-driving forces that arise from cutting processes are calculated from the corresponding increase in line energy. For either of the cutting defects, the increase in dislocation line amounts to the Burgers vector of the intersecting dislocation. Although this short-range interaction does not impose the same immediate hardening effect as a Lomer–Cottrell lock, it subsequently gives rise to the so-called jog drag effect, which is of the utmost relevance to the mobility of the dislocations affected.

The treatment of annihilation is also straightforward. If two segments have a spacing below the critical annihilation distance [EM79] the reaction takes place spontaneously. However, the subsequent reorganization of the dislocation segment vectors is not simple and must be treated with care.

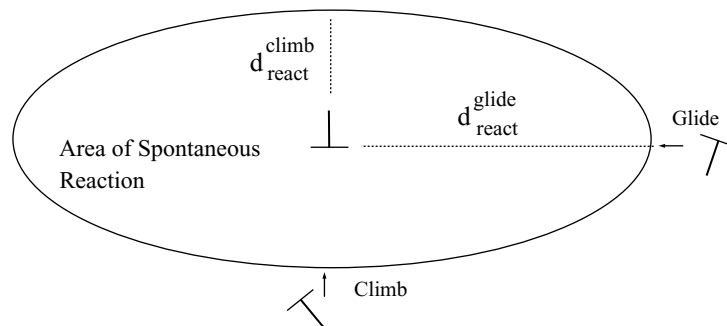


FIGURE 8-9 Reaction ellipse in 2D dislocation dynamics. It is constructed by the values for the spontaneous reaction spacing of dislocations that approach by glide and by climb.

The stress and mobility of glissile dislocation junctions can be simulated by using a simple superposition of the segments involved. Unfortunately, this technique does not adequately reflect the behavior of Lomer–Cottrell locks. Such sessile junctions must therefore be artificially rendered immobile.

Bibliography

- [AB74] R. J. Asaro and D. M. Barnett. *J. Phys. F. Lett.*, 4:L103, 1974.
- [ABH⁺87] S. I. Andersen, J. B. Bilde-Sørensen, N. Hansen, T. Leffers, H. Lilholt, O. B. Pedersen, and B. Ralph. Proceedings 8th RISØ International Symposium on Materials Science. RISØ National Laboratory, Roskilde, 1987.
- [ABHH80] M. F. Ashby, R. Bullough, C. S. Hartley, and J. P. Hirth. *Dislocation Modeling of Physical Systems*. Pergamon, Oxford, 1980.
- [AHBL73] R. J. Asaro, J. P. Hirth, D. M. Barnett, and J. Lothe. *Phys. Stat. Sol. (B)*, 60:261, 1973.
- [Amo90] R. J. Amodeo. *Res. Mech.*, 30:5, 1990.
- [Arg75] A. S. Argon. *Constitutive Equations in Plasticity*. MIT Press, Cambridge, 1975.
- [BAG⁺72] D. M. Barnett, R. J. Asaro, S. D. Gavazza, D. J. Bacon, and R. O. Scattergood. *J. Phys. F*, 2:854, 1972.
- [Bar72] D. M. Barnett. *Phys. Stat. Sol. (B)*, 49:741, 1972.
- [Bar96] D. M. Barnett. Private communication, 1996.
- [BBS79a] D. J. Bacon, D. M. Barnett, and R. O. Scattergood. *Prog. Mater. Sci.*, 23:51, 1979.
- [BBS79b] D. J. Bacon, D. M. Barnett, and R. O. Scattergood. *Phil. Mag. A*, 39:231, 1979.
- [BG79] R. W. Balluffi and A. V. Granato. Vol. 4 *Dislocations in Solids*. Elsevier Science, Amsterdam; Ser. ed.: F. R. N. Nabarro, 1979.
- [BH52] J. Bardeen and C. Herring. *Imperfections in Nearly Perfect Crystals*, Vol. ed.: W. Shockley, Wiley, New York, 1952.
- [BKS73] D. J. Bacon, U. F. Kocks, and R. O. Scattergood. *Phil. Mag.*, 28:1241, 1973.
- [Bou85] J. Boussinesq. *Application des Potentiels à l'Étude de l'Équilibre et du Mouvement des Solides Élastiques*. Gauthier-Villars, Paris, 1985.
- [Bra69] A. D. Brailsford. *Phys. Rev.*, 186:959, 1969.
- [Bra72] A. D. Brailsford. *J. Appl. Phys.*, 43:1380, 1972.
- [Bro67] L. M. Brown. *Phil. Mag.*, 15:363, 1967.
- [Cot53] A. H. Cottrell. *Dislocations and Plastic Flow in Crystal*. Oxford University Press/Clarendon Press, London, 1953.
- [Cot64] A. H. Cottrell. *Theory of Crystal Dislocation*. Gordon and Breach, New York, 1964.
- [DC92] B. Devincere and M. Condat. *Acta Metall.*, 40:2629, 1992.
- [de 60] R. de Wit. *Sol. St. Phys.*, p. 249, 1960.
- [de 67] R. de Wit. *Phys. Stat. Sol.*, 20:567, 1967.
- [Dev95] B. Devincere. *Solid State Commun.*, 93:875, 1995.
- [Dev96] B. Devincere. In *Proceedings of NATO ASI on Computer Simulation in Materials Science*, Vol. eds.: H. O. Kirchner, L. P. Kubin, and V. Pontikis, Vol. 308 of *NATO Advanced Science Institutes Series, Series E : Applied Sciences*, p. 309. Kluwer Academic in cooperation with NATO Science Division, Dordrecht, 1996.
- [DHZ92] I. Demir, J. P. Hirth, and H. M. Zbib. *Mech. Res. Comm.*, 19:369, 1992.
- [dK59] R. de Wit and J. S. Koehler. *Phys. Rev.*, 116:1113, 1959.
- [DK94] B. Devincere and L. P. Kubin. *Mod. Sim. Mater. Sci. Eng.*, 2:559, 1994.
- [EK86] Y. Estrin and L. P. Kubin. *Acta Metall.*, 34:2455, 1986.
- [EM79] U. Essmann and H. Mughrabi. *Phil. Mag. A*, 40:731, 1979.
- [EM84] Y. Estrin and H. Mecking. *Acta Metall.*, 32:57, 1984.
- [EM91] Y. Estrin and P. G. McCormick. *Acta Metall.*, 39:2977, 1991.
- [ERS53] J. D. Eshelby, W. T. Read, and W. Shockley. *Acta Metall.*, 1:251, 1953.
- [Esh49a] J. D. Eshelby. *Proc. Phys. Soc. A*, 62:307, 1949.
- [Esh49b] J. D. Eshelby. *Proc. R. Soc. London A*, 197:396, 1949.

- [Esh57] J. D. Eshelby. *Proc. R. Soc. London A*, 241:376, 1957.
- [FBZ80] P. Franciosi, M. Berveiller, and A. Zaoui. *Acta Metall.*, 28:273, 1980.
- [FGC96] M. C. Fivel, T. J. Gosling, and G. R. Canova. *Mod. Sim. Mater. Sci. Eng.*, 4:581, 1996.
- [FM66] A. J. E. Foreman and M. J. Makin. *Phil. Mag.*, 14:911, 1966.
- [Fra49] F. C. Frank. *Proc. Phys. Soc. A*, 62:131, 1949.
- [Fri56] J. Friedel. *Les Dislocations*. Gauthier-Villars, Paris, 1956.
- [Fri64] J. Friedel. *Dislocations*. Pergamon, Oxford, 1964.
- [GA87] G. Gottstein and A. S. Argon. *Acta Metall.*, 35:1261, 1987.
- [GA89] N. M. Ghoniem and R. J. Amodeo. *Phys. Rev. B*, 41:6958, 1989.
- [GH92] A. N. Gulluoglu and C. S. Hartley. *Mod. Sim. Mater. Sci. Eng.*, 1:1, 1992.
- [GH93] A. N. Gulluoglu and C. S. Hartley. *Mod. Sim. Mater. Sci. Eng.*, 1:383, 1993.
- [GK70] P. P. Gillis and J. Kratochvil. *Phil. Mag.*, 21:425, 1970.
- [GL56] A. V. Granato and K. Lücke. *J. Appl. Phys.*, 27:583, 1956.
- [GLSL90] A. N. Gulluoglu, R. LeSar, D. J. Srolovitz, and P. S. Lomdahl. In *Simulation and Theory of Evolving Microstructures*, Vol. eds.: M. P. Anderson and A. D. Rollett, p. 239. The Minerals, Metals and Materials Society, TMS Publication, 1990.
- [GLST64] A. V. Granato, K. Lücke, J. Schlipf, and L. J. Teutonico. *J. Appl. Phys.*, p. 9, 1964.
- [GSL89] A. N. Gulluoglu, D. J. Srolovitz, R. LeSar, and P. S. Lomdahl. *Scr. Metall.*, 23:1347, 1989.
- [GW94] T. J. Gosling and J. R. Willis. *J. Mech. Phys. Sol.*, 42:1199, 1994.
- [GZ86] J. Gittus and J. Zarka. *Modeling Small Deformations of Polycrystals*. Elsevier Applied Science, London, 1986.
- [HL68] J. P. Hirth and J. Lothe. *Theory of Dislocations*, 1st Ed. McGraw-Hill Series in Materials Science and Engineering, New York, 1968.
- [HL82] J. P. Hirth and J. Lothe. *Theory of Dislocations*, 2nd Ed. Wiley, New York, 1982.
- [Hok63] J. L. Hokanson. *J. Appl. Phys.*, 34:2337, 1963.
- [IL92] V. L. Indenbom and J. Lothe. *Elastic Strain Fields and Dislocation Mobility*, Vol. 31 of *Modern Problems in Condensed Matter Sciences, Ser. eds.: V. M. Agranovich and A. A. Maradudin*. Elsevier Science, Amsterdam, 1992.
- [IO67] V. L. Indenbom and S. S. Orlov. *Sov. Phys.—JETP*, 6:274, 1967.
- [KAA75] U. F. Kocks, A. S. Argon, and M. F. Ashby. In *Prog. Mater. Sci.*, Ser. eds.: B. Chalmers, J. W. Christian, and T. B. Massalski, Vol. 19. Pergamon, Oxford, 1975.
- [KCC+92] L. P. Kubin, G. R. Canova, M. Condat, B. Devincere, V. Pontikis, and Y. J. M. Bréchet. In *Non-Linear Phenomena in Materials Science II*; Vol. eds.: G. Martin and L. P. Kubin, Vol. 23–24 of *Solid State Phenomena*, p. 455. Trans Tech Publication, CH-Aedermannsdorf, 1992.
- [KK96] A. S. Krausz and K. Krausz. *Unified Constitutive Laws of Plastic Deformation*. Academic Press, London, 1996.
- [KL86] H. O. Kirchner and J. Lothe. *Phil. Mag. Lett. A*, 53:L7, 1986.
- [Koc66] U. F. Kocks. *Phil. Mag.*, 13:541, 1966.
- [Koc87] U. F. Kocks. *Unified Constitutive Equations for Creep and Plasticity*; Vol. ed.: A. K. Miller. Elsevier Science, Amsterdam, 1987.
- [Kra40] H. A. Kramers. *Physica*, 7:264, 1940.
- [Kra66] V. Y. Kravchenko. *Sov. Phys.—Solid State*, 8:740, 1966.
- [Krö65] E. Kröner. In *Fehlstellen, Plastizität, Strahlenschäden und Elektronentheorie*, Vol. ed.: A. Seeger, Vol. 1 of *Moderne Probleme der Metallphysik*. Springer-Verlag, Berlin, 1965.
- [Kub93] L. P. Kubin. Chapter 4: *Dislocation Patterning*, Vol. 6; Vol. ed.: H. Mughrabi: *Plastic Deformation and Fracture of Materials of Materials Science and Technology—A Comprehensive Treatment*, Ser. eds.: R. W. Cahn, P. Haasen, and E. J. Krämer. VCH, Weinheim, 1993.
- [LBN93] V. A. Lubarda, J. A. Blume, and A. Needleman. *Acta Metall.*, 41:652, 1993.
- [Lei50] G. Leibfried. *Z. Phys.*, 127:344, 1950.
- [Lei57] G. Leibfried. In *Dislocations and the Mechanical Properties of Crystals*, ed.: J. C. Fisher. Wiley, New York, 1957.
- [Li65] J. C. M. Li. *Acta Metall.*, 13:1097, 1965.
- [LK87] J. Lépinoux and L. P. Kubin. *Scr. Metall.*, 21:833, 1987.

- [MK81] H. Mecking and U. F. Kocks. *Acta Metall.*, 29:1865, 1981.
- [Moh96] V. Mohles. *Thermisch aktivierte Versetzungsbewegung in Kristallen auf der Grundlage von Simulationstechnungen*. Dissertation. Technische Universität Braunschweig, Institut für Metallphysik und Nukleare Festkörperphysik, 1996.
- [MR96] V. Mohles and D. Rönnpagel. In *Comput. Mater. Sci., Proceedings 5th International Workshop on the Computational Modeling of the Mechanical Behaviour of Materials, Aachen, 1996* Vol. eds.: S. Schmauder and D. Raabe, Vol. 7 of *Comput. Mater. Sci.*, Ser. eds.: U. Landman and R. Nieminen, p. 89. Elsevier Science, Amsterdam, 1996.
- [Mug80] H. Mughrabi. In *Proceedings of the 5th International Conference on the Strength of Metals and Alloys IC SMA 5*, Vol. eds.: P. Haasen, V. Gerold, and G. Kostorz, Vol. 3, p. 1615. Pergamon, Oxford, 1980.
- [Mug83] H. Mughrabi. *Acta Metall.*, 31:1367, 1983.
- [Mug93] H. Mughrabi. *Plastic Deformation and Fracture of Materials*; Vol. ed.: H. Mughrabi, Vol. 6 of *Materials Science and Technology—A Comprehensive Treatment*, Ser. eds.: R. W. Cahn, P. Haasen, and E. J. Krämer. VCH, Weinheim, 1993.
- [Mur63] T. Mura. *Phil. Mag.*, 8:843, 1963.
- [Mur87] T. Mura. *Micromechanics of Defects in Solids*. Martinus Nijhoff, Dordrecht, 1987.
- [Nab47a] F. R. N. Nabarro. *Proc. Phys. Soc.*, 59:256, 1947.
- [Nab47b] F. R. N. Nabarro. *Report on the Conference on the Strength of Solids*. Dislocations in Solids. Physical Society, London, 1947.
- [Nab51] F. R. N. Nabarro. *Proc. R. Soc. London A*, 209:278, 1951.
- [Nab89] F. R. N. Nabarro. *Dislocations in Solids*, Ser. ed.: F. R. N. Nabarro. Elsevier Science, Amsterdam, 1989.
- [Nad88] E. Nadgornyi. In *Prog. Mater. Sci.*, Vol. eds. J. W. Christian, P. Haasen, and T. B. Massalski, Vol. 31, Pergamon, Oxford, 1988.
- [Nin70] T. Ninomiya. *Fundamental Aspects of Dislocation Theory*, Vol. eds.: J. A. Simmons, R. de Wit, and R. Bullough, 1:315, 1970.
- [NNH93] S. Nemat-Nasser and M. Hori. *Micromechanics: Overall Properties of Heterogeneous Materials*. North-Holland Series in Applied Mathematics and Mechanics. North-Holland, Amsterdam, 1993.
- [Oro34] E. Orowan. *Z. Phys.*, 89:634, 1934.
- [PA84] F. B. Prinz and A. S. Argon. *Acta Metall.*, 32:1021, 1984.
- [Pei40] R. E. Peierls. *Proc. Phys. Soc.*, 52:34, 1940.
- [Pol34] M. Polanyi. *Z. Phys.*, 89:660, 1934.
- [Raa96a] D. Raabe. *Phil. Mag. A*, 73:1363, 1996.
- [Raa96b] D. Raabe. *Z. Metallk.*, 87:493, 1996.
- [Raa98] D. Raabe. *Phil. Mag. A*, 77:751, 1998.
- [RBC04] D. Raabe, F. Roters, F. Barlat, and L. Q. Chen. *Continuum Scale Simulation of Engineering Materials: Fundamentals—Microstructures—Process Applications*. Wiley-VCH, Weinheim, 2004.
- [RHBj68] A. R. Rosenfield, G. T. Hahn, A. L. Bement, and R. I. Jaffee. *Dislocation Dynamics*. McGraw-Hill, New York, 1968.
- [RHZ94] M. Rhee, J. P. Hirth, and H. M. Zbib. *Acta Metall.*, 42:2645, 1994.
- [Rön87] D. Rönnpagel. In *Proceedings 8th RISØ International Symposium on Materials Science*, eds.: S. I. Andersen, J. B. Bilde-Sørensen, T. Lorentzen, O. B. Pedersen and N. J. Sørensen, p. 503. RISØ National Laboratory, Roskilde, 1987.
- [RR96] F. Roters and D. Raabe. *Mater. Sci. Technol.*, 12:281, 1996.
- [RRG96] F. Roters, D. Raabe, and G. Gottstein. In *Proceedings 5th International Workshop on the Computational Modeling of the Mechanical Behaviour of Materials, Aachen, 1996* Vol. eds.: S. Schmauder and D. Raabe, Vol. 7 of *Comput. Mater. Sci.*, Ser. eds.: U. Landman and R. Nieminen, p. 56. Elsevier Science, Amsterdam, 1996.
- [RRM96] D. Raabe, F. Roters, and V. Marx. *Text. Microstruct.*, 26–27:611, 1996.
- [RSP93] D. Rönnpagel, T. Streit, and T. Pretorius. *Phys. Stat. Sol. (a)*, 135:445, 1993.
- [Sch80] G. Schöck. Chapter 10, Vol. 3 of *Dislocations in Solids*, Ser. ed.: F. R. N. Nabarro. Elsevier Science, Amsterdam; 1980.
- [SdB70] J. A. Simmons, R. de Wit, and R. Bullough. *Fundamental Aspects of Dislocation Theory*. National Bureau of Standards, Washington, D. C., 1970.

- [SDMR57] A. Seeger, J. Diehl, S. Mader, and H. Rebstock. *Phil. Mag.*, 2:323, 1957.
- [SE68] A. Seeger and H. Engelke. In *Dislocation Dynamics*, Vol. eds.: A. R. Rosenfield, G. T. Hahn, A. L. Bement, and R. I. Jaffee, p. 623. McGraw-Hill, New York, 1968.
- [See54] A. Seeger. *Z. Naturforsch.*, 9a:758, 1954.
- [See57] A. Seeger. In *Dislocations and the Mechanical Properties of Crystals*, ed.: J. C. Fisher. Wiley, New York, 1957.
- [Ste73] J. W. Steeds. *Introduction to Anisotropic Elasticity Theory of Dislocations*. Oxford University Press/Clarendon Press, Oxford, 1973.
- [Str54] A. N. Stroh. *Proc. R. Soc. London A*, 67:427, 1954.
- [Str58] A. N. Stroh. *Phil. Mag.*, 3:625, 1958.
- [Suz91] T. Suzuki. *Dislocation Dynamics and Plasticity*, Vol. 12 of *Springer Series in Materials Science*. Springer-Verlag, Berlin, 1991.
- [Tay34] G. I. Taylor. *Proc. R. Soc. London A*, 145:362, 1934.
- [TB66] B. R. Tittmann and H. E. Bömmel. *Phys. Rev.*, 151:178, 1966.
- [Teo82] C. Teodosiu. *Elastic Models of Crystal Defects*. Springer-Verlag, Berlin, 1982.
- [vN95] E. van der Giessen and A. Needleman. *Mod. Sim. Mater. Sci. Eng.*, 3:689, 1995.
- [Voi10] W. Voigt. *Lehrbuch der Kristallphysik*. Teubner Verlag, Berlin, 1910.
- [Vol07] D. Volterra. *Ann. Ecole Norm. Sup.*, (3) XXIV:400, 1907.
- [Wee65] J. Weertman. *Phil. Mag.*, 11:1217, 1965.
- [WL95] H. Y. Wang and R. LeSar. *Phil. Mag. A*, 71:149, 1995.
- [WW80] J. Weertman and J. R. Weertman. Chapters 8, 9, Vol. 3 of *Dislocations in Solids*, Ser. ed.: F. R. N. Nabarro, Elsevier Science, Amsterdam; 1980.
- [Yof60] E. H. Yoffe. *Phil. Mag.*, 5:161, 1960.
- [ZSL97] N. Zacharopoulos, D. J. Srolovitz, and R. LeSar. *Acta Mater.*, 45:3745, 1997.

9 Finite Elements for Microstructure Evolution

—Dierk Raabe

9.1 Fundamentals of Differential Equations

9.1.1 Introduction to Differential Equations

Many of the laws encountered in materials science are most conveniently formulated in terms of differential equations. Deriving and solving differential equations are thus among the most common tasks in modeling material systems.

Differential equations are equations involving one or more scalar or tensorial dependent variables, independent variables, unknown functions of these variables, and their corresponding derivatives. Equations which involve unknown functions that depend on only one independent variable are referred to as ordinary differential equations. If the equations involve unknown functions that depend on more than one independent variable they are referred to as partial differential equations. The “order” of a differential equation is the highest order of any of the derivatives of the unknown functions in the equation. Equations involving only the first derivatives are referred to as first-order differential equations. Equations involving the second derivatives are referred to as second-order differential equations. Second- and higher-order differential equations such as

$$\frac{d^2u(t)}{dt^2} = f(u, t) \quad (9.1)$$

can be transformed into a coupled set of lower-order equations by substitution:

$$\frac{dv(t)}{dt} = f(u, t) \quad v = \frac{du(t)}{dt} \quad (9.2)$$

In these equations u is the state variable that is a function of the independent time variable t , v is the first time derivative of u , and f a function of u and v , respectively. For instance, the frequently occurring problem of (one-dimensional) motion of a particle or dislocation segment of effective mass m under a force field $f(x, t)$ in the x -direction is described by the second-order differential equation

$$m \frac{d^2x(t)}{dt^2} = f(x, t) \quad (9.3)$$

If one defines the particle momentum

$$p(x, t) = m \frac{dx(t)}{dt} \quad (9.4)$$

equation (9.3) becomes the two coupled first-order equations (Hamilton's equations)

$$\frac{dx(t)}{dt} = \frac{p}{m} \quad \frac{dp(x, t)}{dt} = f(x, t) \quad (9.5)$$

Differential equations which contain only linear functions of the independent variables are called "linear differential equations." For these equations the superposition principle applies. That means linear combinations of solutions which satisfy the boundary conditions are also solutions to the differential equation satisfying the same boundary conditions. Differential equations which involve nonlinear functions of the independent variables are denoted as "nonlinear differential equations." For such equations the superposition principle does not apply.

Most problems in computational materials science lead in their mathematical formulation to "partial differential equations", which involve both space and time as independent variables. Usually, one is interested in particular solutions of partial differential equations, which are defined within a certain range of the independent variables and which are in accord with certain initial-value and boundary-value conditions. In this context it is important to emphasize that a problem that is in the form of a differential equation and boundary conditions must be well posed. That means only particular initial and boundary conditions transform a partial differential equation into a solvable problem.

Partial differential equations can be grouped according to the type of additional conditions that are required in formulating a well-posed problem. This classification scheme will be outlined in the following for the important group of linear second-order partial differential equations with two independent variables, say x_1 and x_2 . The general form of this equation is

$$A \frac{\partial^2 u}{\partial^2 x_1} + B \frac{\partial^2 u}{\partial x_1 \partial x_2} + C \frac{\partial^2 u}{\partial^2 x_2} + D \frac{\partial u}{\partial x_1} + E \frac{\partial u}{\partial x_2} + F u + G = 0 \quad (9.6)$$

where $A = A(x_1, x_2)$, $B = B(x_1, x_2)$, $C = C(x_1, x_2)$, $D = D(x_1, x_2)$, $E = E(x_1, x_2)$, $F = F(x_1, x_2)$, and $G = G(x_1, x_2)$ are given functions of the independent variables x_1 and x_2 . It is stipulated that the functions $A(x_1, x_2)$, $B(x_1, x_2)$, and $C(x_1, x_2)$ never be equal to zero at the same point (x_1, x_2) . In analogy to the classification of higher-order curves in analytical geometry that are described by

$$a x_1^2 + b x_1 x_2 + c x_2^2 + d x_1 + e x_2 + f = 0, \quad a^2 + b^2 + c^2 \neq 0 \quad (9.7)$$

equation (9.6) can for given values \hat{x}_1 , \hat{x}_2 , of the variables x_1 and x_2 assume hyperbolic, parabolic, or elliptic character. Roughly speaking, hyperbolic differential equations involve second-order derivatives of opposite sign when all terms are grouped on one side; parabolic differential equations involve only a first-order derivative in one variable, but have second-order derivatives in the remaining variables; and elliptic differential equations involve second order derivatives in each of the independent variables, each of the derivatives having equal sign when grouped on the same side of the equation.

hyperbolic partial differential equation	$4 A(\hat{x}_1, \hat{x}_2) C(\hat{x}_1, \hat{x}_2) < B^2(\hat{x}_1, \hat{x}_2)$
parabolic partial differential equation	$4 A(\hat{x}_1, \hat{x}_2) C(\hat{x}_1, \hat{x}_2) = B^2(\hat{x}_1, \hat{x}_2)$
elliptic partial differential equation	$4 A(\hat{x}_1, \hat{x}_2) C(\hat{x}_1, \hat{x}_2) > B^2(\hat{x}_1, \hat{x}_2)$

In that context it must be considered that, since $A(x_1, x_2)$, $B(x_1, x_2)$, and $C(x_1, x_2)$ depend on independent variables, the character of the differential equation may vary from point to point. The approach to group differential equations according to the character of their discriminant ($4AC - B^2$) is due to its importance in substituting mixed derivatives by new independent variables. The fundamental classification scheme outlined here for second-order partial differential equations can be extended to coupled sets of nonlinear higher-order partial differential equations with more than two independent variables.

Classical examples of the three types of differential equations are the wave equation for the hyperbolic class, the heat or diffusion equation and the time-dependent Schrödinger equation for the parabolic class, and the Laplace and time-independent Schrödinger equation for the elliptic class. In three dimensions and rectangular coordinates then can be written:

$$\begin{aligned} \text{wave equation} \quad & \frac{\partial^2 u}{\partial t^2} - c^2 \left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2} \right) = 0 \\ \text{diffusion equation} \quad & \frac{\partial u}{\partial t} - D \left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2} \right) = 0 \\ \text{Laplace equation} \quad & \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2} = 0 \end{aligned}$$

where x_1 , x_2 , and x_3 are the spatial variables, t the temporal variable, u the state variable, D the diffusion coefficient, which is assumed to be positive and independent of the concentration, and c the propagation velocity of the wave. The assumption that the diffusion coefficient is independent of the concentration applies of course only for certain systems and very small concentrations. In real materials the value of the diffusion coefficient is, first, a tensor quantity and, second, highly sensitive to the concentration.

It is worth mentioning that for stationary processes where $\partial u / \partial t = 0$, the diffusion (heat) equation changes into the Laplace equation. In cases where under stationary conditions sinks or sources appear in the volume being considered, the diffusion equation changes into the Poisson equation:

$$\text{Poisson equation} \quad \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2} - f(x_1, x_2, x_3) = 0$$

which in two dimensions is identical to the differential equation for the description of the transverse displacement of a membrane. An important differential equation similar to the Poisson equation is the Helmholtz equation, which contains both the dependent function itself and its second spatial derivative:

$$\text{Helmholtz equation} \quad \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2} + \alpha u = 0 \quad \alpha = \text{const.}$$

Using the more general Laplace operator $\Delta = \nabla^2$ instead of rectangular coordinates and \dot{u} and \ddot{u} for the first- and second-order time derivatives, respectively, the preceding equations can be rewritten in a more compact notation:

$$\begin{aligned} \ddot{u} - c^2 \Delta u &= 0 \\ \dot{u} - D \Delta u &= 0 \\ \Delta u &= 0 \\ \Delta u - f &= 0 \\ \Delta u + \alpha u &= 0 \end{aligned}$$

Hyperbolic and parabolic partial differential equations typically describe nonstationary, that is, time-dependent problems. This is indicated by the use of the independent variable t in the corresponding equations. For solving nonstationary problems one must define initial conditions. These are values of the state variable and its derivative, which the solution should assume at a given starting time t_0 . These initial conditions could amount to $u(x_1, x_2, x_3, t_0)$ and $\dot{u}(x_1, x_2, x_3, t_0)$ for the wave equation and $u(x_1, x_2, x_3, t_0)$ for the diffusion or heat equation. If no constraints are given to confine the solutions to particular spatial coordinates, that is, $-\infty < x_1, x_2, x_3 < +\infty$, the situation represents a pure initial-boundary problem.

In cases where additional spatial conditions are required, such as $u(x_{10}, x_{20}, x_{30}, t)$ for the wave equation, and $u(x_{10}, x_{20}, x_{30}, t)$ or $(\partial u/\partial x_1)(x_{10}, x_{20}, x_{30}, t)$, $(\partial u/\partial x_2)(x_{10}, x_{20}, x_{30}, t)$, and $(\partial u/\partial x_3)(x_{10}, x_{20}, x_{30}, t)$ for the diffusion equation, or a combination of both, one speaks of a “boundary-initial-value problem.”

Models that are mathematically described in terms of elliptic partial differential equations are typically independent of time, thus describing *stationary* situations. The solutions of such equations depend only on the boundary conditions, that is, they represent pure boundary-value problems. Appropriate boundary conditions for the Laplace or stationary heat and diffusion equation, respectively, $\Delta u = 0$, can be formulated as Dirichlet boundary conditions or as Neumann boundary conditions. Dirichlet boundary conditions mean that solutions for the state variable u are given along the spatial boundary of the system. Neumann boundary conditions mean that solutions for the first derivative $\partial u/\partial x_n$ are given normal to the spatial boundary of the system. If both the function and its normal derivative on the boundary are known, the border conditions are referred to as Cauchy boundary conditions.

9.1.2 Solution of Partial Differential Equations

The solution of partial differential equations by use of analytical methods is only possible in a limited number of cases. Thus, one usually has to resort to numerical methods [Coh62, AS64, BP83, EM88]. In the following sections a number of techniques are presented that allow one to obtain approximate numerical solutions to initial- and boundary-value problems.

Numerical methods to solve complicated initial-value and boundary-value problems have in common the discretization of the independent variables (typically time and space) and the transformation of the continuous derivative into its discontinuous counterpart, that is, its finite difference quotient. Using these discretization steps amounts to recasting the continuous problem expressed by differential equations with an infinite number of unknowns, that is, function values, into a discrete algebraic one with a finite number of unknown parameters which can be calculated in an approximate fashion.

Numerical methods to solve differential equations which are essentially defined through *initial* rather than *boundary* values, that is, which are concerned with time derivatives, are often referred to as finite difference techniques. Most of the finite difference simulations addressed in this book are discrete not only in time but also in space. Finite difference methods approximate the derivatives that appear in differential equations by a transition to their finite difference counterparts. This applies for the time and the space derivatives. Finite difference methods do not use polynomial expressions to approximate functions.

Classical textbooks suggest a substantial variety of finite difference methods [Coh62, AS64, BP83, EM88]. Since any simulation must balance optimum calculation speed and numerical precision, it is not reasonable to generally favor one out of the many possible finite difference solution techniques for applications in computational materials science. For instance, parabolic large-scale bulk diffusion or heat transport problems can be solved by

using a simple central difference Euler method, while the solution of the equations of particle motion in molecular dynamics is usually achieved by using the Verlet or the Gear predictor–corrector method. In most cases, it is useful to select a discretization method with respect to the properties of the underlying differential equations, particularly to the highest occurring order of derivative.

A second group of numerical means of solving differential equations comprises the various finite element methods. These methods are designed to solve numerically both complex boundary-value and initial-value problems. They have in common the spatial discretization of the area under consideration into a number of finite elements, the temporal discretization in cases where time-dependent problems are encountered, and the approximation of the true spatial solutions in the elements by polynomial trial functions. These features explain why they are referred to as finite element techniques.

Although both the finite difference and the finite element techniques can handle space and time derivatives, the latter approach is more sophisticated in that it uses trial functions and a minimization routine. Thus, the finite difference techniques can be regarded as a subset of the various more general finite element approximations [ZM83, ZT89].

Many finite difference and particularly most finite element methods are sometimes intuitively associated with the solution of large-scale problems. Although this association is often true for finite element methods which prevail at solving meso- and macroscale boundary-value problems in computational materials science, it must be underlined that such general associations are inadequate. Finite difference and finite element methods represent mathematical approximation techniques. They are generally *not* intrinsically calibrated to any physical length or timescale. Scaling parameters are introduced by the physics of the problem addressed but not by the numerical scheme employed to solve a differential equation.

9.2 Introduction to the Finite Element Method

This section deals with the simulation of materials properties and microstructures at the mesoscopic and macroscopic levels. Of course, a strict subdivision of the numerous methods that exist in computational materials science according to the length scales that they address is to a certain extent arbitrary and depends on the aspect under investigation.

The finite element method is a versatile numerical means of obtaining approximate solutions to boundary and initial-value problems. Its approach consists in subdividing the sample of interest into a number of subdomains and by using polynomial functions to approximate the true course of a state function in a piecewise fashion over each subdomain [Cou43, ZM83]. Hence, finite element methods are *not* intrinsically calibrated to some specific physical length or timescale. The application of the finite element method in materials science, though, lies particularly in the field of macro- and mesoscale simulations where averaging empirical or phenomenological constitutive laws can be incorporated that describe the actual material behavior in a statistical fashion. Emphasis of the finite element method is often placed on problems in computational mechanics, particularly when the considered shapes are complicated, the material response is nonlinear, or the applied forces are dynamic [RSHP91, NNH93]. All three features are typically encountered in the calculation of large-scale structures and elastic–plastic deformation. The extension of classical computational solid mechanics to microstructure mechanics or computational micromechanics requires a scale-dependent physical formulation of the underlying constitutive behavior that is admissible at the level addressed, and a detailed incorporation of microstructure [GZ86, ABL⁺94, KK96].

This task reveals a certain resemblance to molecular dynamics, where the equations of motion are solved for a large number of interacting particles. These calculations require some approximate formulation of the interatomic potential. It is clear that the accuracy of the underlying potential determines the reliability of the predictions. Similar arguments apply for the use of computational solid mechanics in materials science. The validity of the constitutive laws and the level at which the microstructure is incorporated in the finite element grid determine the predictive relevance of the simulation. As a rule the accuracy of solid mechanics calculations can be increased by decreasing the level at which the required microstructural and constitutive data are incorporated. Advanced finite element methods have more recently also been used on a very fine microstructural scale.

In addition to the previously mentioned field of solid-states mechanics and its treatment with finite element methods, many materials problems exist, which must be formulated as initial-value rather than as boundary-value problems. Such time-dependent simulations are characterized by the presence of time derivatives in the governing differential equations and the prescription of initial conditions at the time origin. Methods to numerically integrate equations which involve time derivatives are provided by the various finite difference approaches [FW60, RM67, Det69, Mar76, DeV94, BP95]. Finite difference methods comprise a number of general numerical means for solving initial-value problems. Typical examples in materials science are the solution of the diffusion or heat equation, of the atomic equations of motion in molecular dynamics, or of the equations of motion in dislocation dynamics. As is apparent from these examples, the use of finite difference algorithms is also *not* confined to any particular scale. The present section gives a discussion of the potential of the finite difference method in meso- and macroscale materials science.

Numerical methods to solve initial- and boundary-value problems have in common the discretization of the independent variables, which are usually time and space, and the transformation of the continuous derivatives into finite difference quotients. Performing these discretizations amounts to recasting the continuous problem expressed by differential equations with an infinite number of unknowns, that is, function values, into an algebraic one with a finite number of unknown parameters that can be calculated in an approximate fashion. Although both the finite difference and the finite element methods can essentially be used to solve boundary- and initial-value problems, the latter technique represents the more general approach since it uses polynomial shape functions and a minimalization procedure. According to Zienkiewicz and Morgan [ZM83], the finite difference techniques can thus be regarded as a subset of the finite element approximation.

9.3 Finite Element Methods at the Meso- and Macroscale

9.3.1 Introduction and Fundamentals

This section is devoted to discussing particular applications of the method for the simulation of materials problems at the meso- and macroscale with special emphasis on large-strain plasticity.

The finite element technique is a numerical method for obtaining approximate solutions to boundary- and initial-value problems by using polynomial interpolation functions. In contrast to analytical techniques, finite elements are also applicable to complicated shapes. The basic characteristic of the finite element method is the discretization of the domain of interest, which may have nearly arbitrary geometry, into an assembly of relatively simply shaped elements that are connected.

The finite element method approximates the real course of the state variables considered within each element by ansatz or interpolation polynomials. This approach of interpolating the variable within each cell amounts to assuming a piecewise polynomial solution over the entire domain under consideration. In the case of elastic and large-strain plastic materials response, it is usually the displacement that is the unknown state variable. The polynomials usually serve also as shape functions to update the form of the finite elements. The coordinate transformation associated with the updating of this mesh, for instance during a simulated large-strain plastic deformation process, is often referred to as the most important component of a successful finite element solution. This problem is of special relevance for simulations at the meso- and macroscale.

In order to derive the governing equations for the resulting displacements in a mechanically consistent manner, the finite element method incorporates certain additional criteria which are usually derived from the equilibrium of the forces. In the field of solid-state deformation this is mostly achieved by minimizing corresponding energy functionals, with the displacement field variable entering in the form of ansatz polynomials. Depending on the character of the material response to external and internal loads, three criteria prevail in solid mechanics: namely, the strong form of the differential equation of motion, the weak form of the virtual work principle, and the stable equilibrium quantified by the minimum potential energy or minimum mechanical energy. While the variational virtual displacement principle is frequently used for nonlinear material response, the use of the minimum potential energy principle prevails for linear material response. The latter method is also referred to as the Dirichlet variational principle.

For obtaining approximate solutions, the polynomial trial functions for the unknown state variables (e.g., displacement) are substituted into the integral functionals for the energy calculation. This technique does not necessarily lead to a minimization of possible deviations at particular nodes but to a minimization of the total, that is, of the integral deviation. Thus, the finite element method focuses on achieving the best possible correspondence between the correct and the approximate solution over the entire system. The incorporation of the trial functions into the integral functionals results in a system of equations that represents a discretized counterpart of the underlying continuous partial differential equations and whose solution is the discrete solution at the nodes [Cou43, Liv83, ZM83]. While most of the early finite element methods used in computational materials science postulated isotropic, homogeneous, linear, and continuous materials properties [ZT89], a number of advanced methods consider material heterogeneity, crystal anisotropy, nonlinear material response, and nonlinear geometrical aspects [GZ86, MAS93, DBM94, BDMK95, BMK96, SD96a, SW96].

In deterministic systems there exists a fundamental difference between spatial coordinates and time, namely, symmetry breaking. While space is isotropic, time always passes in the same direction, that is, it is irreversible. From this qualitative difference in the independent system coordinates it follows that, unlike space-dependent tasks, time-dependent problems must be formulated as initial-value rather than as boundary-value problems. Consequently, time-dependent initial-value simulations are characterized by the presence of time derivatives in the governing differential equations and the prescription of initial conditions at the time origin. Typical examples of such time-dependent differential equations are Newton's equation of motion, the wave equation, the heat equation, and the diffusion equation. The heat and diffusion equations contain first-order and the wave equation second-order time derivatives.

Since it is very tedious or sometimes even impossible to obtain exact solutions to these equations by analytical integration, the employment of numerical approaches is often indispensable. The finite difference method represents a general technique for integrating equations which

involve time derivatives. However, it can also be employed to solve boundary-value or mixed problems [FW60, RM67, Det69, Mar76].

Focusing on applications in large-scale materials simulations, finite difference algorithms are frequently encountered in solving heat flow and bulk diffusion problems. Furthermore, they represent standard techniques for the time integration of the equations of motion in the field of molecular dynamics and dislocation dynamics. Typical variants in these domains are the Verlet and the Gear predictor–corrector algorithm.

Although much of the material presented in this section is relevant to other engineering disciplines, emphasis will be placed here on technical details of the finite element (FE) and finite difference (FD) methods required for predictions in the fields of solid mechanics, heat flow, and solid-state diffusion.

9.3.2 The Equilibrium Equation in FE Simulations

Depending on the character of the material response to external and internal loads, the material dynamics are conveniently described in terms of the “strong form” of the differential equation of motion, the “weak form” of the virtual work principle, or the stable equilibrium quantified by the “minimum mechanical energy.”

A simple straightforward approach to deriving the equations for displacement-based finite element codes starts from the general principle of virtual work. This is the work done by arbitrary small virtual displacements due to the forces and moments acting on a solid body in accord with continuity and displacement boundary constraints. For the most general case this principle can be written

$$\delta\widehat{W} = \iiint_V \sigma_{ij} \delta\widehat{\varepsilon}_{ij} dV = \iiint_V P_j \delta\widehat{u}_j dV + \iint_S T_j \delta\widehat{u}_j dS + \sum F_j \delta\widehat{u}_j \quad (9.8)$$

where $\delta\widehat{W}$ is the virtual work that results from the strain due to the virtual displacements $\delta\widehat{\varepsilon}$ that act on the stresses σ . This work equals the sum of the virtual work that results from the virtual displacements $\delta\widehat{u}$ due to the body force P , to the tractions T , and to point forces F . S is the surface that encloses the volume V .

Equation (9.8) is generally valid for an arbitrary body. However, the finite element method decomposes the solid under investigation into a large number n of simply shaped volume elements that are connected at nodes. Thus, equation (9.8) applies for each individual segment under implicit consideration of equilibrium and compatibility. The course of the displacement is approximated in each finite element by interpolation polynomials that enter all n equations of the form of equation (9.8). This amounts to calculating the volume and surface integrals over each finite segment individually and subsequently summing over all elements. Assuming that the point forces are only applied at the nodal points, equation (9.8) may then be rewritten

$$\sum_n \iiint_V \sigma_{ij} \delta\widehat{\varepsilon}_{ij} dV = \sum_n \iiint_V P_j \delta\widehat{u}_j dV + \sum_n \iint_S T_j \delta\widehat{u}_j dS + \sum_n F_j \delta\widehat{u}_j \quad (9.9)$$

where S is the respective surface enclosing V , the individual element volume.¹

9.3.3 Finite Elements and Shape Functions

The original finite element techniques were based on assuming underlying constitutive rules for the material, which were based on isotropic, homogeneous, linear, and continuous behavior.

¹ Each finite element can have a different volume.

More sophisticated methods, which nowadays prevail in advanced computational materials science, increasingly consider more appropriate constitutive materials laws, which take into consideration material heterogeneity, crystal anisotropy, and nonlinear material response.

However, the finite element algorithm generally represents a continuum type of approach. As a rule, it does not exactly incorporate the genuine three-dimensional dynamics of single lattice defects such as dislocations for the description of crystal plasticity but typically uses averaging constitutive materials laws. This is usually realized by employing the apparatus of the state variable approach in conjunction with the use of implicit variables. This more phenomenological method is required for conducting simulations at the macro- and mesoscale within reasonable computation times. The continuum approach represents an acceptable simplification when the typical spacing of the underlying lattice defects is much shorter than the characteristic dimensions of the body.

In contrast to microscopic simulations where each lattice defect is considered individually, or analytical approaches where the surrounding of an element under investigation is averaged as effective medium, the finite element method has to discretize the solid being investigated into an assembly of simply shaped elements. Figure 9-1 shows some possible shapes of finite elements for two-dimensional simulations. The shapes and sizes of the finite elements are defined in terms of the spatial coordinates of their nodes.

The element nodes are those points where the state variable investigated, say the displacement field \mathbf{u} , is actually calculated by the finite element algorithm. In order to ensure strain compatibility, the finite elements are interconnected at a finite number of such nodes. This means that most nodes belong to more than one element. The values \mathbf{u}_i of the spatial state variable field $\mathbf{u}(x_1, x_2, x_3)$ at the i th node can serve as interpolation points for calculating the course of the variable anywhere in the element to which the node belongs. Elements that allow one the approximation of the coordinates and of the course of the field variable of any point inside the element as a function of the nodal coordinates and of the nodal values of the state variable by simple polynomial functions are referred to as isoparametric elements. The polynomials are usually designed to describe both the element shape and the course of the state variable inside the element. The approximation functions are therefore often referred to as polynomial form functions, shape functions, ansatz functions, or interpolation functions. The use of isoparametric elements has many computational and convergence advantages for accurate finite element solutions. Most finite element methods in computational material science therefore work with isoparametric elements.

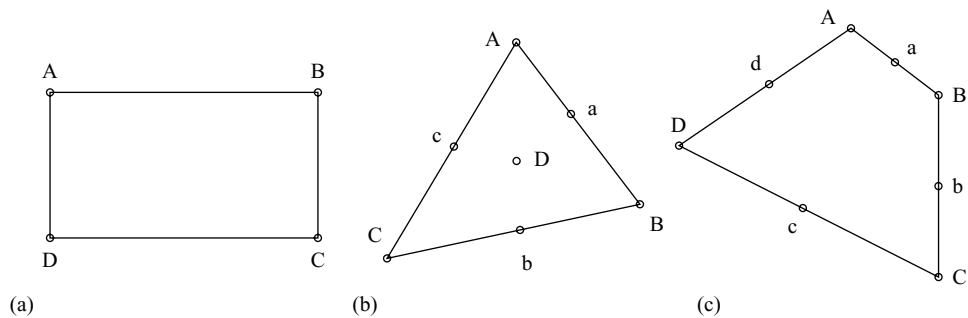


FIGURE 9-1 Some examples of element shapes for two-dimensional finite element calculations, (a) rectangular element, (b) triangular element, (c) quadrilateral element.

For instance, for a simple two-dimensional isoparametric element with four nodes the following set of linear shape functions can be used:

$$\begin{aligned} K_1(\zeta, \varkappa) &= \frac{1}{2} (1 + \zeta) (1 + \varkappa) & K_2(\zeta, \varkappa) &= \frac{1}{2} (1 + \zeta) (1 - \varkappa) \\ K_3(\zeta, \varkappa) &= \frac{1}{2} (1 - \zeta) (1 + \varkappa) & K_4(\zeta, \varkappa) &= \frac{1}{2} (1 - \zeta) (1 - \varkappa) \end{aligned} \quad (9.10)$$

The variables ζ and \varkappa lie within the range $-1 \leq \zeta, \varkappa \leq +1$. The values of the considered field variable $u(\zeta, \varkappa)$ in the element can then be calculated by

$$u(\zeta, \varkappa) = \sum_{i=1}^{n=4} K_i(\zeta, \varkappa) u_i \quad (9.11)$$

where n is the number of nodes and u_i are the values of the field variable at the nodes. The form functions are used to transform the algebraic strain–displacement relations into a more convenient matrix notation. More precisely, the application of shape functions to state variables allows one to map each element into a master generic element with a fixed length by using an isoparametric transformation. Figure 9-2 shows the transformation of a two-dimensional triangular and a two-dimensional quadrilateral element from physical space to mapped space.

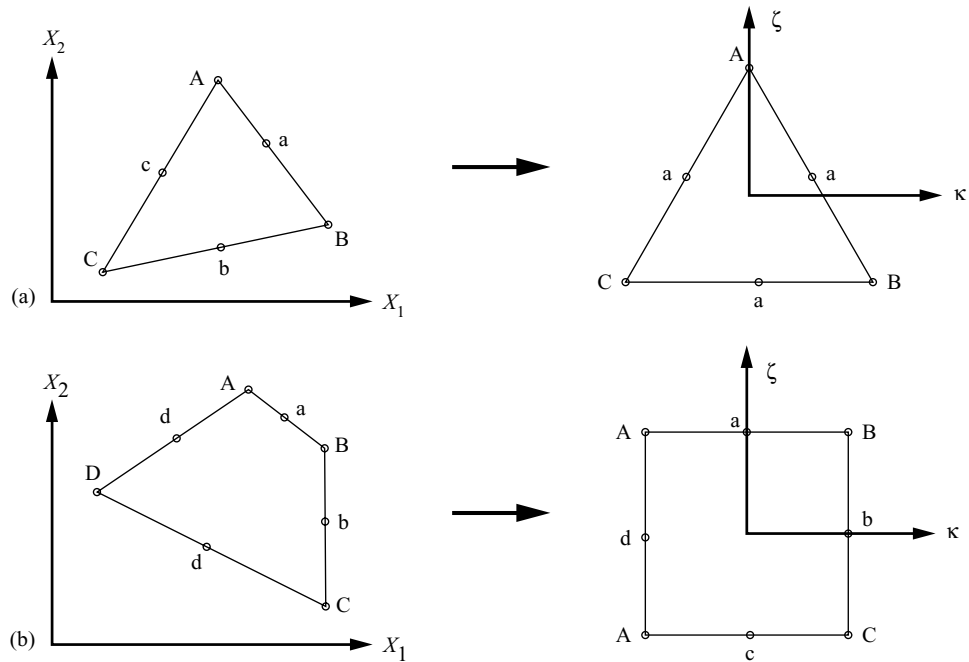


FIGURE 9-2 Transformation of a two-dimensional triangular (a) and a two-dimensional quadrilateral (b) element from the physical coordinate system (x_1, x_2) to the mapped coordinate system.

9.3.4 Assemblage of the Stiffness Matrix

This section describes the assemblage of the element and system stiffness matrix and the consideration of elastic and elastic–plastic constitutive materials response. By defining the spatial derivatives $K_{i,j} = \partial K_i / \partial x_j$ along x_j , $j = 1, 2, 3$, of the n different ansatz functions $i = 1, 2, \dots, n$, the matrix \mathbf{B} with rank $(3, n)$ can be written

$$\mathbf{B} = \begin{pmatrix} K_{1,1} & 0 & 0 & \dots & K_{i,1} & 0 & 0 & \dots & K_{n,1} & 0 & 0 \\ 0 & K_{1,2} & 0 & \dots & 0 & K_{i,2} & 0 & \dots & 0 & K_{n,2} & 0 \\ 0 & 0 & K_{1,3} & \dots & 0 & 0 & K_{i,3} & \dots & 0 & 0 & K_{n,3} \\ K_{1,2} & K_{1,1} & 0 & \dots & K_{i,2} & K_{i,1} & 0 & \dots & K_{n,2} & K_{n,1} & 0 \\ 0 & K_{1,3} & K_{1,2} & \dots & 0 & K_{i,3} & K_{i,2} & \dots & 0 & K_{n,3} & K_{n,2} \\ K_{1,3} & 0 & K_{1,1} & \dots & K_{i,3} & 0 & K_{i,1} & \dots & K_{n,3} & 0 & K_{n,1} \end{pmatrix} \quad (9.12)$$

The infinitesimal strain tensor can be written as a vector:

$$\boldsymbol{\varepsilon}^T = \left(\frac{\partial u_1}{\partial x_1}, \frac{\partial u_2}{\partial x_2}, \frac{\partial u_3}{\partial x_3}, \frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2}, \frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1}, \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \quad (9.13)$$

where u_i and x_j are the displacements and the three orthogonal directions, respectively. Accordingly, the $3n$ components of the displacement vector can be written

$$\mathbf{u}^T = \left(u_{x_1}^1, u_{x_2}^1, u_{x_3}^1, u_{x_1}^2, u_{x_2}^2, u_{x_3}^2, \dots, u_{x_1}^n, u_{x_2}^n, u_{x_3}^n \right) \quad (9.14)$$

where n is the number of element nodes. The relation between strain and displacement can then be compactly presented in the form

$$\varepsilon_{(i=1,\dots,6)} = B_{(i=1,\dots,6) (j=1,\dots,3n)} u_{(j=1,\dots,3n)} \quad (9.15)$$

where the index notation shows the ranks of the matrices. From the \mathbf{B} matrix the stiffness matrices of the elements can be obtained by integration according to

$$\begin{aligned} \mathbf{K}^{\text{elem}} &= \int_{V_{\text{elem}}} \mathbf{B}^T \mathbf{C}^{\text{El}} \mathbf{B} \, dV_{\text{elem}} \\ \mathbf{K}^{\text{elem}} &= \int_{V_{\text{elem}}} \mathbf{B}^T \mathbf{C}^{\text{El,P1}} \mathbf{B} \, dV_{\text{elem}} \end{aligned} \quad (9.16)$$

where \mathbf{C}^{El} is the elastic stiffness tensor and $\mathbf{C}^{\text{El,P1}}$ the elastic–plastic stiffness tensor, each of which describes the constitutive material behavior. The rank of the element stiffness matrix \mathbf{K}^{elem} corresponds to the degrees of freedom per element. The total stiffness matrix \mathbf{K} is obtained by a summation of all element stiffness matrices \mathbf{K}^{elem} . It describes both the actual geometry and the material behavior of the entire system. The rank of \mathbf{K} maps the degrees of freedom of the entire finite element system.

For an isotropic linear elastic material response, the matrix $\mathbf{C}^{\text{El,P1}}$ reduces to \mathbf{C}^{El} and can be written

$$\mathbf{C}^{\text{El}} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} (1-\nu) & \nu & \nu & 0 & 0 & 0 \\ \nu & (1-\nu) & \nu & 0 & 0 & 0 \\ \nu & \nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{(1-2\nu)}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{(1-2\nu)}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{(1-2\nu)}{2} \end{pmatrix} \quad (9.17)$$

using the reduced suffix notation. The complete description requires a fourth-rank tensor notation.

If the material is deformed both elastically and plastically, the matrix $\mathbf{C}^{\text{El,P1}}$ must be used instead of \mathbf{C}^{El} . A typical approach for $\mathbf{C}^{\text{El,P1}}$ has the form

$$\mathbf{C}^{\text{El,P1}} = \mathbf{C}^{\text{El}} - \mathbf{D}(\boldsymbol{\Gamma}) \quad (9.18)$$

One possible approach to write the backstress matrix $\mathbf{D}(\boldsymbol{\Gamma})$ is

$$D_{ij} = \Gamma_i \Gamma_j \frac{3}{2 k_f^2} \left(1 + \frac{2 E E_T (1 + \nu)}{3 E (E - E_T)} \right)^{-1} \quad (9.19)$$

where Γ_n , $n = 1, \dots, 6$ denotes the k th component of the stress deviator written in vector form, E the elastic modulus, E_T the strain hardening coefficient $d\sigma/d\varepsilon$, and k_f the flow stress. Equation (9.19) seems to differ from a typical metallurgical constitutive description since it subtracts the plastic portion from the elastic one rather than adding it. However, this description is consistent with conventional approaches, since it simply extrapolates the elastic stress linearly into the plastic regime beyond the actual yield surface and then subtracts the overestimated portion to end up with a final value which lies exactly on the yield surface. More detailed presentations of the incorporation of nonlinear materials response have been published by Gittus and Zarka [GZ86], Stouffer and Dame [SD96b], and Krausz and Krausz [KK96].

The forces at the element nodes are now computed from the actual stresses by integration:

$$\mathbf{F}^{\text{elem}} = \int_{V_{\text{elem}}} \mathbf{B} \boldsymbol{\sigma} dV_{\text{elem}} \quad (9.20)$$

The total internal force vector \mathbf{F} is calculated by a summation. The integrations in equations (9.16) and (9.20) can be carried out using the Gauss integral formula.

The final infinitesimal displacements at the nodes Δu are found by solving the equation system

$$\begin{pmatrix} K_{1,1} & \dots & K_{1,m} \\ \cdot & \dots & \cdot \\ \cdot & \dots & \cdot \\ \cdot & \dots & \cdot \\ K_{m,1} & \dots & K_{m,m} \end{pmatrix} \begin{pmatrix} u_1 \\ \cdot \\ \cdot \\ \cdot \\ u_m \end{pmatrix} = \begin{pmatrix} \mathcal{F}_1 - F_1 \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{F}_m - F_m \end{pmatrix} \quad (9.21)$$

where $m = 3n$ is the number of degrees of freedom of the system, that is, the number of nodes multiplied by the dimension (2D, 3D), \mathcal{F} the vector of the externally imposed nodal loads,

\mathbf{F} the internal force vector arising from the internal stresses among the elements, and $\mathcal{F} - \mathbf{F}$ the minimization criterion imposed by the requirement for equilibrium.² Since the total stiffness matrix depends implicitly on the node displacements, equation (9.21) is a nonlinear equation that must be solved by iteration.

9.3.5 Solid-State Kinematics for Mechanical Problems

For describing finite deformation kinematics associated with complex loading situations, two basic approaches are conceivable. The first method, referred to as the Lagrange presentation, describes the movement of each elementary unit as a function of time and space. It is thus particularly suitable for describing nonstationary processes. If the equation of motion for each element and deformation step is described in terms of the initial sample coordinate system, the method is referred to as the total Lagrangian description. If the field variables are expressed in terms of the respective preceding state, the approach is denoted as the updated Lagrangian description. The second method, referred to as the Euler description, considers material flow through a given section. Thus, it is a suitable technique for describing stationary processes. Although the Lagrange method is less flexible than the Euler method, its application prevails in the description of nonstationary deformation processes. The Euler method prevails in the investigation of deformation processes with relatively simple material flow kinematics, such as wire drawing or homogeneous cold rolling.

For measuring incremental deformations it is pertinent to define two tensor quantities, namely, the deformation gradient $\mathbf{F}(\mathbf{x}, t)$, and the related displacement gradient $\mathbf{H}(\mathbf{x}, t)$. Following the notation commonly used in continuum mechanics, bold capital letters are used to indicate second-rank tensor quantities while bold lowercase letters indicate first-rank tensor quantities.

The spatial coordinates, $\mathbf{r}(\mathbf{x}, t)$, of each material point at time $t > t_0$ can be described by its preceding coordinates, $\mathbf{x}(t_0)$, at a time $t = t_0$ and the space-dependent displacement $\mathbf{u}(\mathbf{x}, t)$, that is,

$$\mathbf{r}(\mathbf{x}, t) = \mathbf{x}(t_0) + \mathbf{u}(\mathbf{x}, t) \quad (9.22)$$

The deformation gradient tensor \mathbf{F} is then defined by

$$F_{ij} = \frac{\partial r_i}{\partial x_j} \quad (9.23)$$

and the displacement gradient tensor \mathbf{H} by³

$$H_{ij} = \frac{\partial u_i}{\partial x_j} \quad (9.24)$$

The relation between both tensors is expressed by

$$\mathbf{H} = \mathbf{F} - \mathbf{I} \quad (9.25)$$

where \mathbf{I} is the unit tensor

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (9.26)$$

² In full, $\mathcal{F} - \mathbf{F}$ would be written $\mathcal{F}(t + \Delta t) - \mathbf{F}(t)$.

³ In elasticity theory one often uses the abbreviation $u_{i,j}$ for the displacement gradient tensor, where the comma indicates the spatial derivative.

Equation (9.25) indicates that the temporal derivative of the deformation gradient tensor and of the displacement gradient tensor must be equal:

$$\dot{\mathbf{H}} = \dot{\mathbf{F}} \quad (9.27)$$

These derivatives are termed the deformation gradient rate tensor, velocity gradient tensor, or displacement gradient rate tensor. The deformation gradient must remain strictly positive and bounded in order to prevent any self-penetration or cracking of the material. If a volume element undergoes pure translation, the deformation gradient effectively remains equal to 1 and the displacement gradient equal to 0. However, this would no longer be the case if the element were to undergo pure rigid-body rotation. This means that the deformation and the displacement gradient tensor are not objective measures of deformation, since they do filter translation but not rigid-solid rotation. For obtaining an adequate measure that can distinguish between strain and rotation, one uses a linear decomposition of the deformation gradient tensor \mathbf{F} into a symmetric portion \mathbf{U} or \mathbf{W} , and an orthonormalized antisymmetric portion \mathbf{R} :

$$\mathbf{F} = \mathbf{R}\mathbf{U} = \mathbf{W}\mathbf{R} \quad (9.28)$$

This operation, which is referred to as the polar decomposition theorem, implies that each deformation can be decomposed into a strain and a rigid-body rotation. For an arbitrary deformation, \mathbf{W} is not necessarily identical to \mathbf{U} . In other words, the symmetric part of the deformation gradient tensor is a path-dependent quantity and cannot be treated as a state function. By using these definitions it is now possible to introduce tensors that *are* objective measures of deformation, that is, that are independent of the rigid-body rotation. For instance, the product

$$\mathbf{F}^T \mathbf{F} = \mathbf{U}^T \mathbf{R}^T \mathbf{R} \mathbf{U} = \mathbf{U}^2 \quad (9.29)$$

where

$$\mathbf{R}\mathbf{R}^T = \mathbf{I} \quad (9.30)$$

is independent of the rotation \mathbf{R} . The identity of \mathbf{U} and \mathbf{U}^T is straightforward since \mathbf{U} is symmetric. Equation (9.30) follows from the fact that \mathbf{R} is orthogonal and normalized. Thus, \mathbf{U} can be transformed as a pure strain to its principal axes; this corresponds to the solution of an eigenvalue problem. Expressing \mathbf{U} in terms of its eigenvectors \mathbf{n}_i , the transposed eigenvectors \mathbf{n}_i^T and its eigenvalues λ_i yields

$$\mathbf{U} = \sum_{i=1}^3 \lambda_i \mathbf{n}_i \mathbf{n}_i^T \quad (9.31)$$

so that \mathbf{U}^2 can be written

$$\mathbf{U}^2 = \sum_{i=1}^3 \lambda_i^2 \mathbf{n}_i \mathbf{n}_i^T \quad (9.32)$$

Hence, by knowing \mathbf{F} [equation (9.29)], one can calculate the eigenvalues and the rotation associated with the prescribed deformation state. The eigenvalues λ_i represent the incremental rotation-free deformation parallel to the principal axes \mathbf{n}_i . The strain tensor \mathbf{E} can be written

$$\mathbf{E} = \sum_{i=1}^3 \varepsilon_i \mathbf{n}_i \mathbf{n}_i^T \quad (9.33)$$

where ε_i are the strain components parallel to the principal axes \mathbf{n}_i . Typically, the strain components are nonlinear functions of the eigenvalues, that is, $\varepsilon_i = f(\lambda_i)$. Two relations between strain and eigenvalue are typically in use, namely, the logarithmic form, $\varepsilon_i^L = \ln(\lambda_i)$, and the Green–Lagrange form, $\varepsilon_i^G = 1/2(\lambda_i^2 - 1)$. In three dimensions this can be written

$$\begin{aligned}\mathbf{E}^L &= \frac{1}{2} \ln(\mathbf{F}\mathbf{F}^T) \\ \mathbf{E}^G &= \frac{1}{2} (\mathbf{F}\mathbf{F}^T - \mathbf{I})\end{aligned}\quad (9.34)$$

Both tensors represent objective measures of deformation. Using the relation between \mathbf{F} and \mathbf{H} as given in equation (9.25), they can be rewritten

$$\begin{aligned}\mathbf{E}^L &= \frac{1}{2} \ln((\mathbf{H} + \mathbf{I})(\mathbf{H} + \mathbf{I})^T) = \frac{1}{2} \ln((\mathbf{H} + \mathbf{I})(\mathbf{H}^T + \mathbf{I})) \\ &= \frac{1}{2} \ln(\mathbf{H} + \mathbf{H}^T + \mathbf{H}\mathbf{H}^T + \mathbf{I}) \approx \frac{1}{2} \ln(\mathbf{H} + \mathbf{H}^T + \mathbf{I})\end{aligned}\quad (9.35)$$

$$\begin{aligned}\mathbf{E}^G &= \frac{1}{2} ((\mathbf{H} + \mathbf{I})(\mathbf{H} + \mathbf{I})^T - \mathbf{I}) = \frac{1}{2} ((\mathbf{H} + \mathbf{I})(\mathbf{H}^T + \mathbf{I}) - \mathbf{I}) \\ &= \frac{1}{2} (\mathbf{H} + \mathbf{H}^T + \mathbf{H}\mathbf{H}^T) \approx \frac{1}{2} (\mathbf{H} + \mathbf{H}^T)\end{aligned}\quad (9.36)$$

These expressions provide convenient rotation-free measures of deformation.

However, there are four main reasons for employing rate tensor equations instead of the displacement gradient or deformation gradient tensor form. First, the temporal derivative of \mathbf{H} is equal to that of \mathbf{F} , equation (9.27); second, the constitutive behavior of materials is often related not only to the *strain* but to the *strain rate*; third, in work-hardening materials it makes only sense to use incremental strains;⁴ and fourth, many industrial forming processes prescribe the strain rate but not necessarily the strain.

Using the Green–Lagrange form, equation (9.36), one can calculate the symmetric and anti-symmetric portions of the velocity gradient tensor according to

$$\begin{aligned}\dot{\mathbf{E}} &= \frac{1}{2} (\dot{\mathbf{H}} + \dot{\mathbf{H}}^T) = \frac{1}{2} (\dot{\mathbf{F}} + \dot{\mathbf{F}}^T) \\ \dot{\mathbf{\Omega}} &= \frac{1}{2} (\dot{\mathbf{H}} - \dot{\mathbf{H}}^T) = \frac{1}{2} (\dot{\mathbf{F}} - \dot{\mathbf{F}}^T)\end{aligned}\quad (9.37)$$

where $\dot{\mathbf{E}}$ is the strain rate tensor, $\dot{\mathbf{\Omega}}$ the spin, and $\dot{\mathbf{H}} = \dot{\mathbf{F}}$ the velocity gradient tensor. While the $\dot{\mathbf{E}}$ tensor is symmetric, the $\dot{\mathbf{\Omega}}$ tensor is antisymmetric.

9.3.6 Conjugate Stress–Strain Measures

From the objective strain measures derived in the preceding section, so-called conjugate stress quantities can be found. In general, an increment of the energy density associated with a Cauchy stress tensor $\boldsymbol{\sigma}$ and a strain increment $d\boldsymbol{\varepsilon}$ is calculated by $dW = \sigma_{ij} d\varepsilon_{ij}$, where the Einstein summation rule applies. A stress tensor that is combined with an objective strain tensor increment to give the same increment of the energy density dW is then referred to as energetically conjugate to that strain.

⁴ The shape and the size of the yield surface can change after each strain increment due to work-hardening and crystal anisotropy. The strain increment can then be divided by the simulation time step to give a strain rate.

The stress tensor that is conjugate to the Green–Lagrange strain tensor \mathbf{E}^G is denoted as the second Piola–Kirchhoff stress tensor $\mathbf{\Lambda}$:

$$dW = \left[J (\mathbf{F}^{-1})^T \boldsymbol{\sigma} \mathbf{F}^{-1} \right]_{ij} dE_{ij}^G = \mathbf{\Lambda} d\mathbf{E}^G \quad (9.38)$$

where \mathbf{F} is the deformation gradient tensor and J the volume dilatation. Since the volume dilatation in most metals is equal to 1, the preceding equation can be rewritten

$$dW = \left[(\mathbf{F}^{-1})^T \boldsymbol{\sigma} \mathbf{F}^{-1} \right]_{ij} dE_{ij}^G = \left[\mathbf{R}^T \boldsymbol{\sigma} \mathbf{R} \right]_{ij} dE_{ij}^G = \mathbf{\Lambda} d\mathbf{E}^G \quad (9.39)$$

where the polar decomposition theorem has been applied [see equation (9.28)].

Accordingly, the Kirchhoff stress tensor $\boldsymbol{\Sigma}$, which is conjugate to the logarithmic strain \mathbf{E}^L , is defined through

$$dW = \left[\mathbf{R}^T (\boldsymbol{\Sigma}) \mathbf{R} \right]_{ij} dE_{ij}^L \quad (9.40)$$

It is often pertinent to dismantle stress tensor increments into a part that depends on the rigid-body rotation, and a part that reflects the stress change as enforced by the constitutive material behavior. In general form such a decomposition can, for a given tensor \mathbf{A} , be written

$$\dot{\mathbf{A}} = \overset{\nabla}{\mathbf{A}} + \boldsymbol{\Omega} \mathbf{A} + \mathbf{A} \boldsymbol{\Omega}^T \quad (9.41)$$

where $\overset{\nabla}{\mathbf{A}}$ is referred to as the Jaumann derivative.

Bibliography

- [ABL⁺94] S. I. Andersen, J. B. Bilde-Sørensen, T. Lorentzen, O. B. Pedersen, and N. J. Sørensen. Proceedings 15th RISØ International Symposium on Materials Science. RISØ National Laboratory, Roskilde, 1994.
- [AS64] M. Abramovitz and I. A. Segun. *Handbook of Mathematical Functions*. Dover, New York, 1964.
- [BDMK95] A. J. Beaudoin, P. R. Dawson, K. K. Mathur, and U. F. Kocks. *Int. J. Plast.*, 11:501, 1995.
- [BMK96] A. J. Beaudoin, H. Mecking, and U. F. Kocks. *Phil. Mag. A*, 73:1503, 1996.
- [BP83] J. F. Botha and G. F. Pinder. *Fundamental Concepts in the Numerical Solution of Differential Equations*. Wiley, New York, 1983.
- [BP95] N. Bellomo and L. Preziosi. *Modeling Mathematical Methods and Scientific Computation*. CRC Press, Boca Raton, FL, 1995.
- [Coh62] A. M. Cohen. *Numerical Analysis*. McGraw-Hill, London, 1962.
- [Cou43] R. Courant. *Bull. Am. Math. Soc.*, 49:1, 1943.
- [DBM94] P. R. Dawson, A. J. Beaudoin, and K. K. Mathur. In *Proceedings 15th RISØ International Symposium on Materials Science*, eds.: S. I. Andersen, J. B. Bilde-Sørensen, T. Lorentzen, O. B. Pedersen, and N. J. Sørensen, p. 33. RISØ National Laboratory, Roskilde, 1994.
- [Det69] J. W. Dettman. *Mathematical Methods in Physics and Engineering*. Wiley, New York, 1969.
- [DeV94] P. L. DeVries. *A First Course in Computational Physics*. Wiley, New York, 1994.
- [EM88] G. Engeln-Müllges. *Formelsammlung zur Numerischen Mathematik mit Standard-FORTRAN 77-Programmen*. BI Wissenschaftsverlag, Mannheim, Wien, Zürich, 1988.
- [FW60] G. E. Forsythe and W. R. Wasow. *Finite Difference Methods for Partial Differential Equations*. Wiley, New York, 1960.
- [GZ86] J. Gittus and J. Zarka. *Modeling Small Deformations of Polycrystals*. Elsevier Applied Science, London, 1986.

- [KK96] A. S. Krausz and K. Krausz. *Unified Constitutive Laws of Plastic Deformation*. Academic Press, London, 1996.
- [Liv83] R. K. Livesley. *Finite Elements—An Introduction for Engineers*. Cambridge University Press, Cambridge, 1983.
- [Mar76] D. Marsal. *Die numerische Lösung partieller Differentialgleichungen*. BI-Wissenschaftsverlag, Bibliographisches Institut Mannheim, Leipzig, Wien, Zürich, 1976.
- [MAS93] P. E. McHugh, R. J. Asaro, and C. F. Shih. *Acta Metall.*, 41:1461, 1993.
- [NNH93] S. Nemat-Nasser and M. Hori. *Micromechanics: Overall Properties of Heterogeneous Materials*. North-Holland Series in Applied Mathematics and Mechanics. North-Holland, Amsterdam, 1993.
- [RM67] R. D. Richtmyer and K. W. Morton. *Difference Methods for Initial Value Problems*. Wiley (Interscience), New York, 1967.
- [RSH91] G. W. Rowe, C. E. N. Sturgess, P. Hartley, and I. Pillinger. *Finite Element Plasticity and Metal Forming Analysis*. Cambridge University Press, Cambridge, 1991.
- [SD96a] G. B. Sarma and P. R. Dawson. *Acta Metall.*, 44:1937, 1996.
- [SD96b] D. C. Stouffer and L. T. Dame. *Inelastic Deformation of Metals*. Wiley, New York, 1996.
- [SW96] S. Schmauder and D. Weichert. *Proceedings 4th International Workshop on the Computational Modeling of the Mechanical Behaviour of Materials, Lille, 1995*; Vol. eds.: S. Schmauder and D. Weichert: Vol. 5 of *Comput. Mater. Sci.*; Ser. eds.: U. Landman and R. Nieminen. Elsevier Science, Amsterdam, 1996.
- [ZM83] O. C. Zienkiewicz and K. Morgan. *Finite Elements and Approximation*. Wiley, New York, 1983.
- [ZT89] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method*, Vol. 1. McGraw-Hill, London, 1989.

This page intentionally left blank

Index

A

Abnormal grain growth, 72, 98
Acceleration
 Dislocation, 267
Activation energy
 Diffusion, 152, 157
Activity, 25, 165
 Chemical, 25
Activity coefficient, 25, 27, 165
Affinity, 16
Airy stress function, **283**
Airy's scalar stress function, 284
Alloy, 220
Anisotropic
 Elastic field, 278
 Elasticity, **278**
Anisotropic energies, 68
Anisotropic grain growth, 95
Anisotropy, 224
Annihilation, 312
Ansatz
 Function, 325
 Polynomial, 323
Antisymmetric part
 Displacement gradient tensor, 272
Approximate solution, 323
Athermal dislocation dynamics, 306
Atomic mobility, see diffusion, mobility
Average
 Description, 322
Avogadro constant, 20
Avrami, 112
 Coefficient, 183, 184

 Exponent, 183–185

Avrami, M., 181

B

Backstress, **328**
Boltzmann diffusion equation, 162
Boltzmann, L., 19
Boundary condition, 231, 318
 Cauchy, **320**
 Dirichlet, **320**
 Neumann, **320**
Boundary conditions, 51, 52
Boundary energy, 68
Boundary mobility, 72
Boundary-initial-value problem, 320
Boundary-value problem, 318, 320–322
Bow-out criterion, 300
Bravais, 1
Bulk free energy density, 224

C

Calculation speed, 321
Calibration
 Models, 321
CALPHAD, 7, 33, 37
Carnot cycle, 13
Carnot, Nicolas, 13
Cauchy
 Boundary condition, **320**
 Stress tensor, 331
Cellular automata
 Curvature
 Kink template, 124

- Definition, 109
- Grain growth, 123
- Hybrid, 136
- Inverse updating, 135
- Irregular, 131
- One-dimensional, 110
- Probabilistic, 113
- Random, 130
- Recrystallization, 116
- Shapeless, 131
- State change rule, 110
- State transformation function, 110
- Central
 - Difference method, **321**
- Characteristic
 - Polynom, 273
- Checkerboarding, 85
- Chemical driving force, 28
- Chemical potential, 11, 12, 17, 25, 165, 242
 - Equilibrium, 26
- ChemSage, 8
- Clausius, Rudolf, 10
- Coarsening
 - Grain, 31
 - Precipitate, 31
- Compatibility, **274**
- Complex
 - Loading situation, 329
- Compositional fluctuation, 15
- Conjugate stress–strain measures, **331**
- Constitutive
 - Law, 322
 - Empirical, 322
 - Phenomenological, 322
- Continuous
 - Derivative, 320
- Coordination number, 22
- Critical nucleation energy, 188
- Cross-slip force, 302, **306**
- CSL, 97
- Cubic lattice, 50
- Cubic symmetry, 225
- Curie temperature, 36
- Curvature induced pressure, 30
- Cutting criterion, 300
- D**
- Deformation gradient tensor, 332
- Delta function, 281
- Dendrite, 220
- Dependent variable, 318, 320

- Difference
 - Equation, 320
- Differential equation, 281, 320, 321
 - Diffusion, 319
 - Discretization, **320**
 - Elliptic, **319, 320**
 - Elliptic, 320
 - Equilibrium, 273
 - Hyperbolic, **319, 320**
 - Introduction, **318**
 - Laplace, **319**
 - Linear, 318
 - Nonstationary, **320**
 - Nonhomogeneous, 281
 - Nonlinear, 318
 - Numerical solution, **320**
 - Parabolic, **319, 320**
 - Partial, 318, 320
 - Poisson, **319**
 - Stationary, 320
 - Wave, **319**
- Diffuse interface, 222
- Diffusion, 137, 151, 241, 321, 322
 - Coefficient, 319
 - Equation, **319, 320**
 - Coefficient, 160, 171, 200
 - Convective flux, 173
 - Coordinate system, 170
 - Couple, 168
 - Darken's equation, 170
 - Drift term, 170
 - Fick's first law, 160
 - Fick's second law, 161
 - Flux, 165
 - Generalized force, 171
 - Impurity diffusion, 167
 - Interdiffusion, 168, 170
 - Interstitial, 153
 - Intrinsic, 169, 171
 - Kirkendall effect, 168
 - Macroscopic, 160
 - Mechanism, 151, 168
 - Microscopic, 154
 - Mobility, 165, 171
 - Multicomponent, 171
 - Numerical solution, 174
 - Phenomenological laws, 160
 - Point source, 162
 - Random walk, 155
 - Reference frame, 168, 169
 - Self diffusion, 167

- Time reversal, 154
- Tracer diffusion, 167
- Vacancy flux, 168
- Vibration frequency, 157
- Dilute solution, 27
- Dirichlet boundary condition, **320**
- Dirichlet variational principle, 323
- Discretization, 229, **320**
 - Finite element simulation, **324**
 - Independent variable, **320**
- Discretization and interpolation in finite element simulation, **324**
- Dislocation, 268
 - Annihilation, 312
 - Finite, 284
 - Flexible, 312
 - Glide plane, 312
 - Infinite, 284
 - Lomer–Cottrell lock, 312
 - Mobile junction, 312
 - Mobility, 313
 - Segment, 268, 284, **300**
- Dislocation dynamics, **298, 299, 302**
 - 2D, 307, 308, 311, 312
 - 3D, 307, 308, 311, 312
 - Athermal, 305
 - Back-driving force, 313
 - Bow-out, 301
 - Bow-out criterion, 300
 - Canonical ensemble, 299
 - Constriction force, 302
 - Cross-slip force, 302, **306**
 - Cutting, 313
 - Cutting criterion, 300
 - Discrete dislocations, 308, 311
 - Dislocation reaction, **312**
 - Dislocation–dislocation force, 302, **302**
 - Dynamic equilibrium, 302
 - Electron drag, **304**
 - Electron drag force, 302
 - Equilibrium, 302
 - External force, 302, **303**
 - Force equilibrium, 302
 - Forces, **302, 304–306**
 - Image, **303**
 - Image force, 302
 - Inertia, **305**
 - Kinematics, **310**
 - Langevin force, 302
 - Line tension, 302
 - Linear elastic-limit criterion, 300
 - Local equilibrium, 302
 - Lomer–Cottrell lock, 313
 - Long-range interactions, 312
 - Mechanical equation of state, **301**
 - Mobility, 313
 - Newton’s equation of motion, 302
 - Newtonian, **299**
 - Obstacle force, 302, **304**
 - Orowan’s equation, 311
 - Osmotic force, 302, **304**
 - Overdamped high stress regime, 308
 - Peach–Koehler force, **302**
 - Peierls force, 302, **302**
 - Phenomenological, **308**
 - Phonon drag force, 302, **304**
 - Pressure dependence, 299
 - Quasi-harmonic approximation, 299
 - Segmentation, **300**
 - Self force, 302, **302**
 - Short-range interactions, 308, 311, 312
 - Solution of the equation of motion, **307**
 - Strain rate tensor, 311
 - Stress fluctuation criterion, 300
 - Temperature dependence, 299
 - Thermal, 306
 - Thermal activation, 306, 308
 - Thermal force, 302, **305**
 - Viscoplastic, **307**
 - Viscous, **307**
- Dislocation statics, **284–288**
 - 2D, **285, 286, 287, 289**
 - Anisotropic, **287**
 - Isotropic, **285**
 - 3D
 - Anisotropic, **292**
 - Field equation, **289**
 - Field equations, **292**
 - Isotropic, **289**
 - Infinite dislocation, **285, 286, 287**
 - Isotropic, **289**
- Dislocation–dislocation force, 302, **302**
- Displacement field, 269
- Displacement gradient tensor, 272
 - Antisymmetric part, 272
 - Symmetric part, 272
- Dissipation, 30
- Distribution
 - Gaussian, 163
 - Probability, 163
 - Second moment, 159
- Double well potential, 224

Drift velocity, 165
Driving force, 15, 17
 Chemical, see chem. driving force
 General, 16
Driving pressure, 112
Dynamic
 Equilibrium, 302
 Recovery, 312

E

EBSD, 89
Eden model, 135
Edge dislocation, **285**, **287**
Efficiency, 15
Einstein's equation, 159
Elastic
 Constants, **278**
 Energy, **280**
Elasticity
 Theory, **269**, 283, 284
Electron
 Drag force, **302**
Element, 323
Energetically conjugate, 331
Energy
 Internal, 8
 Mechanical, 13
 Thermal, 13
Energy balance, 245
Enthalpy, 9, 10, 194
 Mixing, 21, 23, 25
Entropy, 12
 Boltzmann, 20
 Configurational, 20
 Ideal, 21
 Mixing, 19
 Production, 12, 16
 Thermodynamic definition, 10
Entropy functional, 222, 241
Equation
 Diffusion, 319
 Heat, 319
 Helmholtz, 319
 Poisson, 319
 Wave, 319
Equation of motion, 329
Equilibrium, 16, 302
 Equation, **273**
 Rotational, 274
 Translational, 274
 Multiphase, 25

Error function, 164
Error function complement, 164
Euler
 Finite element method, 329
 Kinematics, **329**
 Method, 321
 Representation, **329**
Euler forward integration, 175
Euler method, 230
Explicit integration, 175
Extended volume, 182
Extensive property, 9
External force, 302

F

$F^*A^*C^*T$, 8
Fick's first law, see diffusion, Fick's first law
Fick's second law, see diffusion, Fick's second law
Field
 Equation, 278, 284
Finite deformation kinematics, 329
Finite difference method, 228, **320**, 321, 322, 324
 Scaling, 321
Finite difference quotient, **320**
Finite element method, 321–324, 329
 Discretization, 324
 Equation of motion, 329
 Euler representation, **329**
 Finite deformation kinematics, 329
 Interpolation, 324
 Interpolation function, **325**
 Isoparametric element, **325**
 Lagrange representation, **329**
 Scaling, 321
 Shape function, **324**
 Solid-state kinematics, **329**
Flexible dislocation, 312
Force
 Interatomic, 269
Form function, 322, **325**
Functional, 323

G

Galilean transformation, 172
Gear predictor–corrector method, 321
General solution, 25, 33
Gibbs energy, 11, 16
 Common tangent, 26
 Excess, 31, 33
 Magnetic ordering, 36

- Minimization, 7, 16, 40, 43
 - Molar, 11, 12, 22, 23
 - Gibbs, J. W., 18
 - Gibbs–Thomson equation, 33
 - Ginzburg–Landau functional, 243
 - Glauber, 66
 - Glauber dynamics, 49, 50
 - Glide plane, 312
 - Gradient entropy density, 224
 - Grain boundary, 2
 - Mobility, 112
 - Orientation
 - Plane normal, 133
 - Triple junction, 134
 - Grain growth, 67, 87, 88, 253
 - Grain structure, 220
 - Green’s function, 268, **280**, 281, 283
 - Tensor, 281
 - Time-dependent, 281
 - Time-independent, 281
 - Green–Lagrange
 - Strain tensor, 331
 - Tensor, 331
- H**
- Hamiltonian, 48, 49, 66, 69, 77, 91
 - Heat
 - Equation, **319**, 320
 - Heat capacity, see specific heat cap.
 - Heat flux, 15
 - Heat reservoir, 13
 - Heat transport equation, 239
 - Helmholtz
 - Equation, **319**
 - Henry’s law, 28
 - Hexagonal lattice, 50
 - Hydrostatic stress, 273
 - Hyperbolic partial differential equation, 320
- I**
- Ideal gas, 9
 - Ideal solution, 21
 - Image force, 302
 - Impingement, 181
 - Implicit time integration, 177
 - Impurity diffusion, 167
 - Incubation time, 189
 - Independent state parameter, 211
 - Independent variable, 318, 320
 - Discretization, **320**
 - Inertia force, 302
 - Infinite dislocation, 268, 284
 - Initial condition, 318
 - Initial-value problem, 318, 320, 322
 - Intensive property, 9
 - Interatomic
 - Potential, 321
 - Interatomic bonding, 194
 - Interdiffusion, 168, 247
 - Interface, 179
 - Interface curvature, 30
 - Interface pressure, 30
 - Interfacial energy, 186, 194
 - Internal energy, 8
 - Internal processes, 15
 - Internal variable, 16, 17
 - Interpolation function, 325
 - Intrinsic diffusion, see diffusion, intrinsic
 - Ising model, 48, 54, 55
 - Isoparametric element, **325**
 - Isotropic
 - Elasticity, **280**
 - Limit, **278**
- J**
- Jaumann derivative, 332
 - Johnson, W., 181
 - Jump frequency, 159
- K**
- Kawasaki, 66
 - Kawasaki dynamics, 49, 53
 - Kinematics, **329**
 - Kinetic coefficient, 248
 - Kirchhoff stress tensor, 332
 - Kirkendall drift, 169
 - Kirkendall effect, 168
 - Kirkendall velocity, 170
 - KJMA theory, 181
 - Kocks, U. F., 308
 - Kolmogorov, A. N., 181
- L**
- Lagrange
 - Representation, **329**
 - Lagrange multiplier, 43, 248
 - Laplace
 - Equation, **319**, 320
 - Operator, 319
 - Large-scale
 - Problem, 321

- Levi–Civita operator, 272
- Linear
 - Elastic-limit criterion, 300
 - Elasticity, **268**
 - Shape function, 325
- Local
 - Equilibrium, 302
- Logarithmic strain, 332
- Lomer–Cottrell lock, 312
- LSW–theory, 208

- M**
- Macroscale, 322
 - Boundary-value problem, 321
- Magnetic entropy, 36
- Magnetic excess energy, 36
- Magnetic ordering, 36, 38
- Mass balances, 245
- Mass conservation, 162
- MatCalc, 40
- Material
 - Plasticity, 323
- Mathematical
 - Approximation technique, 321
- Maximum entropy production, 212
- Maxwell’s dilemma, 155
- MDF, 89
- Mechanical
 - Equation of state, **301**
- Mechanical mixture, 18
- Mehl, R. F., 169, 181
- Membrane equation, 320
- Mesoscale, 322
 - Boundary-value problem, 321
- Metropolis, 66, 69
- Metropolis function, 49
- Metropolis probability transition function, 72, 93
- Microstructure, 2, 220
 - Homogenization, 321
- Evolution
 - One-dimensional, 111
- Minimal surfaces, 53
- Minimum potential energy principle, 323
- Mobile junction, 312
- Mobility coefficient, 246
- Mohr circle, 273
- Molar quantity, 11
- Mole fraction, 41
- Mole fractions, 11
- Molecular dynamics, 321

- Monte Carlo, 66
- Monte Carlo method, 49
- Monte Carlo time step, 50
- Morphology, 220
- Motion by curvature, 54
- Moving boundary problem, 179
- MTData, 8
- Multiobstacle potential, 245
- Multiwell potential, 244
- Multicomponent, 241
- Multiphase, 241
- Mura, T., 283

- N**
- Nearest-neighbor broken-bond model, 194
- Nearest-neighbor
 - Bonds, 22
- Neighborhood
 - Corrected Moore, 120
 - Margolus, 118
 - Moore, 117
 - Von Neumann, 117
- Neumann boundary condition, **320**
- Newton’s equation of motion, 302
 - Dislocation dynamics, **299**
- Nonstationary, 320
 - Process, **329**
- Nonhomogeneous differential equation, 281
- Nonlinear
 - Materials response, 328
- Nucleation, 111
 - Classical nucleation theory, 186
 - Continuous, 113
 - Macroscopic treatment, 186
 - Probabilistic, 114
 - Site saturated, 111
 - Solid state, 185
- Nucleation barrier, 188
- Nucleation rate, 189
- Nucleus composition, 191
- Numerical
 - Precision, 321
- Numerical Kampmann–Wagner model, 199, 206

- O**
- Obstacle force, 302
- Onsager relations, 247
- Onsager, L., 212
- Orowan, E., 311
- Orthoequilibrium composition, 193
- Osmotic force, 302

Ostwald ripening, 31, 208
 Overdamped
 High-stress dislocation dynamics, 308

P
 PANDAT, 8
 Paraequilibrium composition, 193
 Parabolic partial differential equation, 320
 Partial differential equation, 280, 281, 283, 284, 320, **320**
 Green's function, **280**
 Introduction, **318**
 Numerical solution, **320**
 Solution, 280
 Pattern formation, 220
 Peach–Koehler force, **302**
 Peierls force, **302**
 Phase transformation, 220
 Impingement, see impingement
 Microstructure, 180
 Precipitation, see precipitation
 Sharp-interface
 Phase-field equation, 224, 246
 Phase-field model, 241
 Phonon drag force, 302
 Piecewise polynomial solution, 323
 Pinning, 104
 Pinning systems, 75
 Piola–Kirchhoff stress tensor, 331
 Poisson equation, **319**, 320
 Polar decomposition theorem, 332
 Polycrystal, 241
 Polynomial
 Form function, 322
 Function, 322
 Trial function, 321
 Potts boundary-site algorithm, 81
 Potts model, 64, 80
 Potts model applications, 87
 Potts N -fold way algorithm, 82
 Potts parallel algorithm, 84
 Potts temperature, 72
 Precipitate distribution, see precipitation, size distribution
 Precipitation, 179, 197
 Coarsening, 197
 Diffusion-controlled, 197
 Interface-controlled, 197
 Mean-field approach, 210
 Numerical Kampmann–Wagner model, 206
 SFFK model, 209

Size distribution, 207
 Principal stress, 273
 Principle of time reversal, 154

Q
 Quadrijunctions, 68, 70

R
 Radioactive isotopes, 167
 Random walk, 155
 Mean displacement, 156
 Root mean square displacement, 156, 158
 Raoult's law, 28
 Reaction/interaction criterion, 300
 Read–Shockley, 97, 100
 Read–Shockley function, 72
 Read–Shockley relation, 92
 Recrystallization, 80, 102, 111
 Redlich–Kister polynomial, 33, 39
 Reference state, 8
 Regular solution, 22, 27
 Rigid-body rotation, 271, 332
 Ring mechanism, 152
 Rotation, 272
 Rotational equilibrium, 274
 Roughening temperature, 62
 Rönnpagel, D., 306

S
 Scale, 321
 Parameter, 321
 Screw dislocation, **286**, **288**
 Self-diffusion, 167
 Self force, 302
 Self-similarity, 69
 SGTE, 37
 Shape function, **325**
 Simulation
 Finite difference, 322
 Finite element, 322
 Macroscale, 322
 Mesoscale, 322
 Sink, 319
 Site fraction, 41
 Soap bubble, 67
 Soap bubbles, 64
 Solid-state
 Kinematics, **329**
 Mechanics, 323
 Solidification, 220

- Solution
 - General, see general solution
 - Ideal, see ideal solution
 - Regular, see regular solution
- Solution thermodynamics, 18
- Source, 319
- Space derivative, 321
- Specific heat capacity, 10, 36
- Speed-up algorithms, 80
- Spontaneous process, 15
- Square lattice, 50
- State change rule, 109
- State function, 9
- State transformation function, 109
- State variable, 9, 322, 323
- Static equilibrium, 273, 274
- Stationary, 320
 - Diffusion equation, 320
 - Heat equation, 320
 - Process, 319, 329
- Stefan, J., 179
- Stored energy, 77
- Strain
 - Field, **269**
 - Green–Lagrange, 331
 - Hardening, 312
 - Rate tensor, 331
- Stress
 - Cauchy tensor, 331
 - Deviator, 273
 - Field, **272**
 - Fluctuation criterion, 300
 - Function
 - Airy’s, **283**
 - Piola–Kirchhoff tensor, 331
 - Strain
 - Conjugate, **331**
- Structure
 - Evolution equation, 307
- Sublattice model, 34, 41
- Superposition principle, 318
- Symmetric, 66
- Symmetric function, 49
- Symmetric part
 - Displacement gradient tensor, 272
- Invariants, 273
- Strain, 272
- Stress, 273
- Stress deviator, 273
- Ternary alloy, 253
- Texture, 89, 96
- Thermal
 - Activation, 308
 - Force, 302, **306**
- Thermal activation, 157
- Thermal fluctuation, 15
- ThermoCalc, 8, 37
- Thermodynamic
 - Database, 37
 - Equilibrium, 16, 40
 - Extensive property, 9
 - Factor, 166
 - Intensive property, 9
 - Software, 8
 - State function, 9
 - State variables, 9
- Thermodynamic extremal principle, 30, 202, 212
- Thermodynamics, 7
 - Equilibrium, 306
 - Computational, 7
 - Equilibrium, 8
 - First law, 8
 - Irreversible, 13
 - Of solutions, 18
 - Reversible, 8
 - Second law, 12, 15
- Time
 - Derivative, 321
- Tracer diffusion, 167
- Traction, 273
- Trail functions, 323
- Transformation, 321, 322
- Transient nucleation rate, 191
- Translational
 - Equilibrium, 274
- Trial function, 321
- Triple point, 65
- Triple point mobility, 93
- Triple points, 67
- T**
- Taylor expansion, 273
- Taylor, R. L., 321
- Tensor
 - Displacement gradient, 272
- V**
- Vacancy
 - Enthalpy of formation, 153
 - Equilibrium density, 153
 - Interstitial, 35

- Random walk, 154
- Substitutional, 153
- Vacancy exchange mechanism
- Vacancy wind, 168
- Vacancy-assisted diffusion, 153
- Vanilla algorithm, 53, 54
- Variable
 - Dependent, 318
- Variational
 - Potential energy principle, 323
 - Principle, 323
 - Dirichlet, 323
 - Virtual displacement, 323
 - Virtual displacement principle, 323
- Velocity
 - Gradient tensor, 331
- Verlet
 - Method, 321
- Virtual displacement principle, 323
- Viscoplastic
 - Dislocation dynamics, 308

- Viscous
 - Dislocation dynamics, 308
- Volterra, D., 283
- Volume
 - Dilatation, 332
- Volume energy, 131
- von Mises, R., 273
- Voronoi, 130

W

- Wave
 - Equation, **319**, 320
- Weertman, J., 308

Y

- Yield surface, 275, 328

Z

- Zener anisotropy ratio, 278
- Zener pinning, 75, 103
- Zienkiewicz, O. C., 321

This page intentionally left blank