



Technische Universität Graz  
Institut für Festigkeitslehre  
Institut für Werkstoffkunde  
und Schweißtechnik  
Kopernikugasse 24/I  
8010 Graz

**Peter Loidolt**

Rechnerische Beschreibung von Vorstadien der  
Rekristallisation in Metallen

**Masterarbeit**

zur Erlangung des akademischen Grades eines  
Diplom-Ingenieurs

Technische Universität Graz  
Fakultät Maschinenbau

Studienrichtung:  
Wirtschaftsingenieurwesen-Maschinenbau

Betreuer: Dipl.-Ing. Dr.techn. Manfred Ulz  
Beurteiler: Univ.-Prof. Dipl.-Ing. Dr.techn. Christof Sommitsch

## Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, \_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

## Eidesstattliche Erklärung<sup>1</sup>

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am \_\_\_\_\_  
Datum

\_\_\_\_\_  
Unterschrift

---

<sup>1</sup>Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

# Inhaltsverzeichnis

<b>Deklaration</b>	<b>I</b>
<b>Abbildungsverzeichnis</b>	<b>IV</b>
<b>Abkürzungsverzeichnis</b>	<b>VI</b>
<b>Danksagung</b>	<b>VII</b>
<b>Kurzfassung/Abstract</b>	<b>VII</b>
<b>I. Theorie</b>	<b>1</b>
<b>1. Metallphysik</b>	<b>1</b>
1.1. Aufbau von Metallen . . . . .	1
1.1.1. Nulldimensionale Fehlstellen . . . . .	2
1.1.2. Eindimensionale Fehlstellen . . . . .	3
1.1.3. Zweidimensionale Fehlstellen . . . . .	5
1.1.4. Dreidimensionale Fehlstellen . . . . .	6
1.2. Kaltverformung . . . . .	6
1.3. Erholung . . . . .	11
1.4. Rekristallisation . . . . .	12
1.4.1. Keimbildung . . . . .	14
1.5. Particle Stimulated Nucleation PSN . . . . .	16
1.5.1. Mechanismus von PSN . . . . .	16
1.5.2. Textur nach PSN . . . . .	18
<b>2. Kontinuumsmechanik</b>	<b>19</b>
2.1. Deformationsgradient . . . . .	19
2.2. Verzerrungen . . . . .	20
2.3. Verformungsraten . . . . .	21
2.4. Multiplikative Zerlegung des Deformationsgradienten . . . . .	22
2.5. Gleichgewicht . . . . .	22
2.6. Spannung . . . . .	24
2.7. FE-Ansatz . . . . .	24
2.8. Plastizitätsmodelle . . . . .	25
2.9. Numerische Lösung . . . . .	26
<b>3. Modellierung</b>	<b>27</b>
3.1. Subkornrotation . . . . .	28
3.2. FE-Modell . . . . .	35
3.2.1. Geometrischer Aufbau des FE-Modells . . . . .	36
3.2.2. Vernetzung . . . . .	38

3.2.3. Materialkennwerte . . . . .	39
<b>II. Ergebnis und Diskussion</b>	<b>41</b>
<b>4. Ergebnisse</b>	<b>41</b>
4.1. Dehnungen und Rotationen . . . . .	41
4.2. Subkornbildung durch Erholung . . . . .	46
4.3. Rotation nach Subkornbildung . . . . .	48
4.4. Missorientierung benachbarter Subkörner . . . . .	54
<b>5. Diskussion</b>	<b>58</b>
<b>6. Ausblick</b>	<b>64</b>
<b>III. Anhang</b>	<b>67</b>
<b>7. Beschreibung der Erholung im Nahbereich eines harten Einschlusses</b>	<b>67</b>
7.1. Modell zur Beschreibung der lokalen Erholung . . . . .	67
7.2. Vereinfachtes Modell . . . . .	68
7.3. Simulation . . . . .	70
7.3.1. Lösung für homogene Verformung . . . . .	70
7.3.2. Lösung für heterogene Verformung . . . . .	75
7.4. Zusammenfassung . . . . .	78
<b>8. Python Skript für die Berechnung der Subkornrotationen</b>	<b>80</b>
<b>Literaturverzeichnis</b>	<b>94</b>

## Abbildungsverzeichnis

1.	Kristallgitter [1] . . . . .	2
2.	Punktdefekte [1] . . . . .	3
3.	Versetzungen [1] . . . . .	3
4.	Burgersumlauf [1] . . . . .	4
5.	Aufspaltung Burgersvektor [1] . . . . .	4
6.	Korngrenzen [1] . . . . .	5
7.	kfz-Gleitsysteme [1] . . . . .	8
8.	Schmidsche Schubspannung [1] . . . . .	8
9.	Spannungsverteilung Stufenversetzung [2] . . . . .	9
10.	Kraftwirkung Versetzungen [1] . . . . .	10
11.	Verfestigung Einkristall [1] . . . . .	11
12.	Polygonisieren – theoretisches Modell [3] . . . . .	12
13.	Polygonisieren – real [3] . . . . .	12
14.	Resultierender Druck an Grenzfläche [3] . . . . .	13
15.	Mobilität als Funktion der Subkornmissorientierung [4] . . . . .	13
16.	HVEM Untersuchung von PSN in einer Al-Si-Legierung [3] . . . . .	16
17.	Subkornwachstum in der Nähe eines harten Teilchens während in-situ Glühversuch [3] . . . . .	17
18.	Keimbildung als Funktion der Dehnung und des Teilchendurchmessers [3] . . . . .	17
19.	111 Polfigur eines teilweise rekristallisierten Ni-SiO <sub>2</sub> Einkristalls [3] . . . . .	18
20.	Mittlere maximale Missorientierung an Teilchen [3] . . . . .	28
21.	Orientierungsgradient 1D . . . . .	29
22.	Maximale Missorientierung Subkorn 1D . . . . .	30
23.	Wahrscheinliche Missorientierung Subkorn 1D . . . . .	31
24.	Orientierungsgradient 2D . . . . .	32
25.	Maximale Missorientierung 2D . . . . .	33
26.	Wahrscheinliche Missorientierung 2D . . . . .	34
27.	Rotation des Matrixmaterials im Nahbereich eines Einschlusses . . . . .	36
28.	Randbedingungen . . . . .	37
29.	Modellierung Subkörner . . . . .	38
30.	Vernetzung . . . . .	39
31.	Fließkurven Johnson-Cook . . . . .	40
32.	Maximale Rotation an einem kugelförmigen Einschluss . . . . .	41
33.	Lokale Dehnung und Rotation als Funktion des Verfestigungsexponenten . . . . .	42
34.	Lokale Dehnung und Rotation als Funktion der Teilchenform . . . . .	44
35.	Maximale lokale Dehnung und Rotation als Funktion der Dehnung . . . . .	45
36.	Lokale Rotation als Funktion der Teilchenform . . . . .	46
37.	Subkornorientierung als Funktion der Erholung . . . . .	47
38.	Missorientierung als Funktion der Erholung . . . . .	48
39.	Rotation eines Subkornes an einem $d_p = 5\mu\text{m}$ Teilchen als Funktion der globalen Dehnung und der Subkorngröße . . . . .	49
40.	Rotation eines Subkornes mit Durchmesser $4\mu\text{m}$ an Teilchen verschiedener Größe . . . . .	50

41.	Rotation eines Subkornes bestimmter Größe als Funktion von Dehnung und Teilchendurchmesser . . . . .	51
42.	Rotation eines Subkornes als Funktion des Verfestigungsexponenten . . . . .	52
43.	Rotation eines $2\mu\text{m}$ Subkornes an einem rotationselliptischen Teilchen mit Hauptachsenlänge $A=5\mu\text{m}$ . . . . .	53
44.	Rotation eines Subkornes an rotationselliptischen Teilchen . . . . .	54
45.	4 Subkörner an $10\mu\text{m}$ Teilchen . . . . .	55
46.	Missorientierung des zentralen Subkornes an einem $10\mu\text{m}$ Teilchen zu seinen Nachbarn . . . . .	56
47.	Missorientierung des zentralen Subkornes radial nach außen zu Subkorn 4 . . . . .	57
48.	Rotationszonen in der Nähe eines Teilchens [5] . . . . .	58
49.	Rotationszonen schematisch in der Nähe von Teilchen [5] . . . . .	58
50.	GROD-Karte in der Nähe von Teilchen nach Kaltverformung $\varepsilon \approx 0,5$ [6] . . . . .	59
51.	Bereiche mit großen Rotationsgradienten in der Simulation . . . . .	59
52.	GROD-map Übersicht [6] . . . . .	60
53.	GROD-map in der Nähe von kleinen Teilchen nach Kaltverformung $\varepsilon \approx 0,5$ [6] . . . . .	62
54.	Mögliches Anwachsen der Subkörner über den Quadranten hinaus . . . . .	63
55.	Keimbildung als Funktion der Teilchengröße und der Dickenreduktion . . . . .	64
56.	Substrukturen mit diskreten Versetzungen [7] . . . . .	66
57.	Defektdichte nach 10min Glühzeit . . . . .	72
58.	Temperaturverlauf . . . . .	72
59.	Versetzungsdichte nach Kaltverformung $\varepsilon = 0,6$ und Glühen nach Abb.58 . . . . .	73
60.	Leerstellenkonzentration nach Kaltverformung $\varepsilon = 0,6$ und Glühen nach Abb.58 . . . . .	73
61.	Fließspannung nach Kaltverformung $\varepsilon = 0,6$ und Glühen nach Abb.58 . . . . .	73
62.	Defektdichten nach unterschiedlichen Dehnungen . . . . .	74
63.	Lokale plastische Dehnung $\varepsilon_{global} = 0,5$ . . . . .	75
64.	Lokale Leerstellenkonzentration zu unterschiedlichen Zeitpunkten . . . . .	76
65.	Lokale Versetzungsdichten zu unterschiedlichen Zeitpunkten . . . . .	77
66.	Detail lokale Versetzungsdichten . . . . .	78
67.	Lokale Versetzungsdichten zu unterschiedlichen Zeitpunkten für $K_{VD} = 1$ . . . . .	78

## Abkürzungsverzeichnis

$\varepsilon$	Dehnung
$\gamma$	Scherung
$\nu$	Querdehnzahl
$\rho$	Versetzungsdichte
$\sigma$	Normalspannung
$\tau$	Schubspannung
$c_V$	Konzentration Leerstellen
$G$	Schubmodul
GND	Geometrisch notwendige Versetzungen
kfz	kubisch flächenzentriert
PSN	Particle Stimulated Nucleation
$v$	Geschwindigkeit
$D$	Dehnrates
$F$	Deformationsgradient
$L$	Geschwindigkeitsgradient
$R$	Rotationstensor
$V$	Verzerrungstensor
$W$	Rotationsgeschwindigkeit
$\lambda$	Dehnungsverhältnis
FEM	Finite Elemente Methode
$\Delta\Theta$	Missorientierung an Grenzflächen
$\Theta$	Rotation bzw. Missorientierung bezüglich Ausgangszustand
$d_p$	Teilchendurchmesser
$d_{SG}$	Subkorndurchmesser
A,B	Ellipsenhauptachse, Ellipsennebenachse
$c$	Exponent für Johnson-Cook Materialmodell
PEEQ	Plastische äquivalente Dehnung
CPFE	Kristallplastische Finite Elemente
EBS	Elektronenrückstreubeugung
GROD	Grain Rotation Orientation Deviation

## Danksagung

Die vorliegende Arbeit entstand im Sommer- und Wintersemester 2013 am Institut für Werkstoffkunde und Schweißtechnik der TU Graz in Zusammenarbeit mit dem Institut für Festigkeitslehre der TU Graz. Ich möchte dem Institutsleiter Professor Sommitsch für die Ermöglichung dieser Arbeit sowie deren Betreuung danken. Weiters gilt mein Dank Dr. Manfred Ulz vom Institut für Festigkeitslehre für die fachliche Unterstützung auf dem Bereich der FE-Simulation. Ohne diese Hilfe wäre die rechnerische Umsetzung der betrachteten Modelle nicht möglich gewesen. Ein besonderer Dank gilt Dozent Dr. Werner Mitter für die anregenden Diskussionen auf dem Bereich der Metallphysik, die zur Entwicklung der Modelle beigetragen haben.

## Kurzfassung

Ziel dieser Diplomarbeit war die Entwicklung und die rechnerische Umsetzung eines Modells zur Beschreibung der Keimbildung für Rekristallisation, wobei das Augenmerk auf die Keimbildung in der Nähe von harten Einschlüssen gelegt wird. Es werden dabei zwei verschiedene Ansätze verfolgt. Zum einen wird die Verteilung der Versetzungsdichte und der Leerstellenkonzentration in der Nähe eines harten Teilchens nach der Kaltverformung und nach einer Glühbehandlung berechnet. Dabei handelt es sich um eine zweidimensionale Simulation eines Modells, deren theoretische Grundlagen und eindimensionale Lösung von Mitter und Sommitsch in [8] veröffentlicht wurden. Da sich die Ergebnisse dieser Simulation schwer mit dem Hauptteil dieser Arbeit verknüpfen lassen, werden sie im Anhang angefügt.

Der Hauptteil dieser Arbeit beschäftigt sich mit abnormalem Subkornwachstum als Keimbildungsmechanismus. Missorientierungen von Subkörnern sind in diesem Zusammenhang von großer Bedeutung. In der Nähe von harten Teilchen kommt es bei plastischer Verformung zu gleichsinnig gerichteten Orientierungsgradienten mit großen Reichweiten, wodurch die Bildung von Subkörnern mit großen Missorientierungen möglich ist. Aus diesem Grund wird die plastische Dehnung und die Rotation des Materials im Nahbereich des Teilchens mit Hilfe der FE-Software ABAQUS berechnet. Dabei wird auf einfache isotrope Materialmodelle zurückgegriffen. Da es für das hier betrachtete Problem keine angemessene Simulationsmethode zur Beschreibung der Subkornbildung gibt, werden für die Subkorngröße realistische Werte angenommen. Für diese Subkörner wird die Rotation im Bezug auf den Ausgangszustand vor der Verformung berechnet. Dazu wird eine Methode eingeführt, mit der die Rotation der Subkörner aus dem Ergebnis der FE-Simulation berechnet werden kann. Der Unterschied der Rotation zweier benachbarter Subkörner, also deren Missorientierung, wird als Keimbildungskriterium definiert. Es wird eine Reihe von Simulationen durchgeführt um die Keimbildung als Funktion der globalen Dehnung, des Teilchendurchmessers und des Subkorndurchmessers vorausszusagen. Letztendlich werden die Simulationsergebnisse mit Versuchsergebnissen verglichen.



## Abstract

The objective of this master thesis was the development and numerical solution of a model which describes the nucleation of recrystallization. The focus is on the formation of nuclei near to hard particles. Next to that, the distribution of dislocation density and vacancy concentration was calculated for the cold deformed state as well as for the annealed state. This is the two-dimensional simulation of a model, which is described and solved in one dimension by Mitter and Sommitsch in [8]. The solution of this simulation is attached in the appendix.

The main part of this work deals with abnormal subgrain growth as a mechanism for nucleation. The misorientations of subgrains are of great interest. Plastic deformation leads to long range orientation gradients in the vicinity of hard particles, which enables the formation of highly misoriented subgrains. This is the motivation to use the FE software ABAQUS to simulate the plastic strain and the material rotations near to a particle. Simple material models are used for this simulation. Since there is no adequate simulation method for the description of the subgrain formation according to the problem considered here, realistic values for the subgrain size are assumed. The rotation of the subgrains with respect to the undeformed state is calculated by introducing a novel method, which uses the results of the FE simulation as a basis. The difference of rotation of two subgrains is their misorientation and is defined as the nucleation criteria. A number of simulations are performed to predict the formation of nuclei as a function of the global strain, the particle diameter and the subgrain size. Finally the numerical results are compared with experimental observations.

# Teil I.

## Theorie

In diesem Teil werden die theoretischen Grundlagen erörtert. Dabei wird zuerst auf die physikalischen Gesetzmäßigkeiten eingegangen, die für das grundlegende Verständnis von Rekristallisation notwendig sind.

Als nächster Schritt wird die FE-Methode als Werkzeug zur Beschreibung von plastischer Verformung erklärt.

Als letzter Schritt wird ein Modell zur Beschreibung der Substruktur in der Nähe von harten Teilchen sowie der damit einhergehenden Keimbildung für Rekristallisation eingeführt.

### 1. Metallphysik

Für das Verständnis von Rekristallisation ist es notwendig, Grundlagen der Metallphysik zu erörtern. Zuerst wird auf den Aufbau von Metallen eingegangen. Danach wird beschrieben, was während der Verformung von Metallen passiert. Dies ist notwendig, da Rekristallisation nur bei verformten Metallen auftreten kann. Letztendlich werden die Phänomene Erholung und Rekristallisation beschrieben.

#### 1.1. Aufbau von Metallen

(Dieses Kapitel ist in Anlehnung an [1, Kapitel 3])

Metalle sind aus regelmäßig angeordneten Atomen aufgebaut. Diese Anordnung wird als Kristallgitter bezeichnet. Für Metalle wichtige Gitter sind das kubisch-raumzentrierte Gitter krz (*alpha*-Fe, Cr), das kubisch-flächenzentrierte Gitter kfz (*gamma*-Fe, Cu, Al, Ag, Au) und das hexagonal dichtest gepackte Gitter hdp (*alpha*-Ti, Co, Zn).

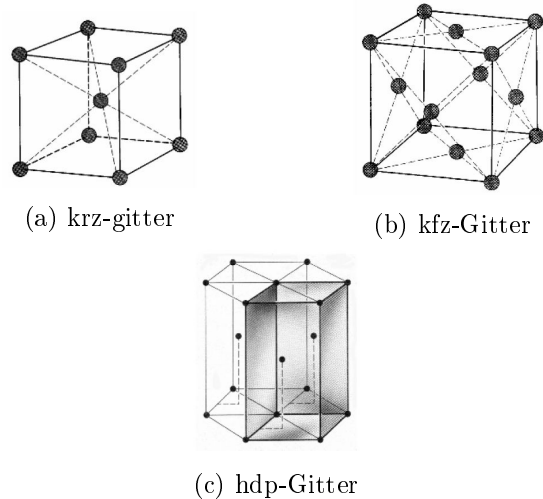


Abbildung 1: Kristallgitter [1]

Würde es einen idealen Einkristall geben, so könnte bei bekannter Einheitszelle die Lage jedes einzelnen Atoms im Körper durch Translation der Einheitszelle beschrieben werden. Tatsächlich beinhalten größere metallische Körper eine Vielzahl von Defekten.

### 1.1.1. Nulldimensionale Fehlstellen

Die kleinsten Fehlstellen sind die nulldimensionalen Fehlstellen. Sie haben abgesehen vom eigenen Atomvolumen keine räumliche Erstreckung und werden aus diesem Grund Punktdefekte genannt. Zu ihnen zählen die Leerstellen und Zwischengitteratome (Abbildung 2 a und b). D.h. an manchen Gitterstellen fehlt ein Atom bzw. an manchen Stellen zwischen den tatsächlichen Gitterstellen befinden sich Atome. Aus energetischer Sicht sollten keine Leerstellen bestehen, da das ungestörte Gitter die niedrigste Energie besitzt. Mit der Bildung von Leerstellen ist eine Steigerung der Entropie verbunden, wodurch die freie Enthalpie bei einer bestimmten Konzentration an Leerstellen minimal wird. Leerstellen sowie Zwischengitteratome sind also im thermodynamischen Gleichgewicht vorhanden und können mit Hilfe der freien Bildungsenthalpie  $G_B^L$  berechnet werden.

$$c_L^a = \exp\left(\frac{S_v^L}{k}\right) \exp\left(-\frac{H_B^L}{kT}\right) = \exp\left(-\frac{G_B^L}{kT}\right) \quad (1)$$

Wobei  $c_L^a$  die atomare Leerstellenkonzentration,  $S_v^L$  die Schwingentropie je Leerstelle,  $H_B^L$  die Enthalpie für die Bildung einer Leerstelle,  $k$  die Boltzmannkonstante und  $T$  die absolute Temperatur ist. Leerstellen sind essentiell für Diffusionsprozesse von Gitteratomen. Im idealen Gitter ist keine Diffusion möglich. Wie in Gleichung 1 ersichtlich steigt die Leerstellenkonzentration mit zunehmender Temperatur. Aus diesem Grund findet nennenswerte Diffusion erst bei erhöhter Temperatur statt. Dabei ist nicht die Absoluttemperatur sondern die relative Temperatur bezogen auf die Schmelztemperatur maßgeblich, da die freie Bildungsenthalpie mit zunehmender Schmelztemperatur steigt.

Weitere nulldimensionale Defekte sind Fremdatome, die sich als Substitutionsatom an einem Gitterplatz (Abbildung 2c) oder als interstitiell gelöste Atome auf Zwischengitterplätzen befinden. Für die weiteren Betrachtungen in dieser Arbeit haben diese gelösten Fremdatome keine besondere Relevanz. Aus diesem Grund werden im Folgenden unter dem Begriff Punktdefekte nur die Leerstellen und Zwischengitteratome verstanden.

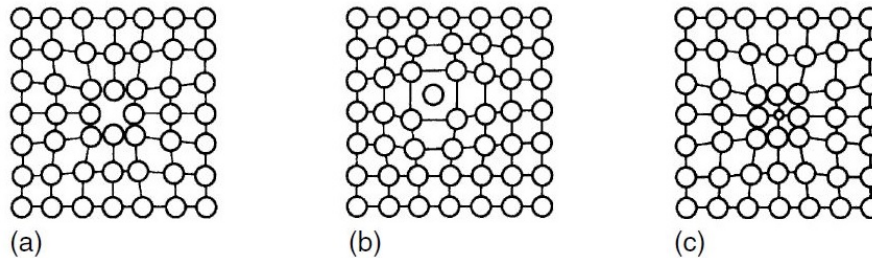


Abbildung 2: Punktdefekte [1]

### 1.1.2. Eindimensionale Fehlstellen

Zu den eindimensionalen Defekten gehören die Kristallversetzungen. Der am einfachsten vorstellbare Fall einer Versetzung ist eine Stufenversetzung. Dabei endet eine Kristallebene mitten im Kristall. Die Linie, entlang derer die Kristallebene endet, wird als Versetzungslinie bezeichnet. Der etwas kompliziertere Fall der Schraubenversetzung entsteht, wenn sich eine Gitterebene wie eine Schraubenfläche um die Versetzungslinie dreht.

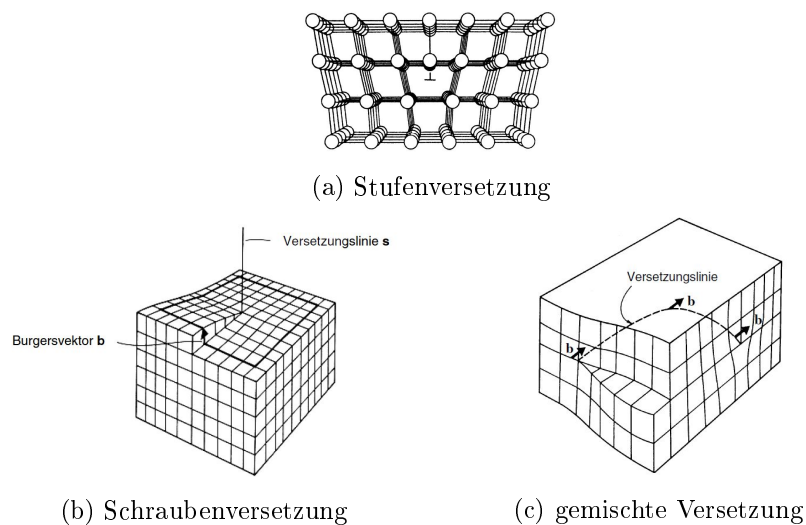


Abbildung 3: Versetzungen [1]

Zur Charakterisierung der Versetzungen wird der durch den Burgers-Umlauf bestimmte Burgersvektor (Abbildung 4) verwendet. Dabei wird die Versetzungslinie mit einem

Linienzug mit Sprüngen von Gitterplatz zu Gitterplatz umrundet. Die Differenz des Linienzuges für den gedachten fehlerfreien und des realen Kristalls wird als Burgersvektor bezeichnet. Im Fall der Stufenversetzung ist der Burgersvektor normal zur Versetzungslinie, bei der Schraubenversetzung parallel (Abbildung 3a und 3b). Der Charakter der Versetzung kann sich – wie in Abbildung 3c gezeigt – entlang der Versetzungslinie verändern. Dabei kann im Übergangsbereich der Burgersvektor in einen Anteil  $\vec{b}_s$  parallel zur Versetzungslinie (Schraubenversetzung) und einen Anteil  $\vec{b}_e$  normal dazu (Stufenversetzung) aufgeteilt werden (Abbildung 5).

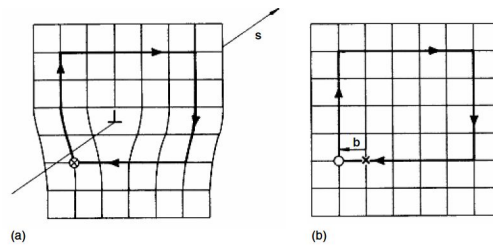


Abbildung 4: Burgersumlauf [1]

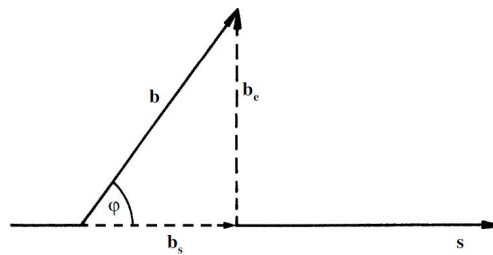


Abbildung 5: Aufspaltung Burgersvektor [1]

Versetzungen können sich bewegen und verursachen durch ihre Bewegung plastische Verformung. Bei der Bewegung verschieben sich die Versetzungslinien bevorzugt in dichtest gepackten Ebenen des Kristalls. Dabei unterscheiden sich Stufen- und Schraubenversetzungen darin, dass sich Stufenversetzungen nur in der Ebene bewegen können in der sie sich befinden, währenddessen Schraubenversetzungen durch Quergleiten die Gleitebene wechseln können. Die einzige Möglichkeit einer Stufenversetzung die Gleitebene zu wechseln ist die Anlagerung und Abspaltung von Punktdefekten an der Versetzungslinie. Dadurch kann durch Klettern die Gleitebene gewechselt werden. Die Anzahl der Versetzungen im Kristall spielt für die mechanischen Eigenschaften eine große Rolle. Aus diesem Grund wird die Versetzungsdichte als Länge aller Versetzungslinien pro Volumeneinheit  $[m/m^3]$  definiert. Bei regelmäßig parallel angeordneten Versetzungslinien kann die Versetzungsdichte  $\rho$  aus dem mittleren Versetzungsabstand  $d$  berechnet werden.

$$\rho = \frac{1}{d^2} \left[ \frac{m}{m^3} \right] \quad (2)$$

Beim Minimum der freien Enthalpie sollte die Versetzungsdichte null sein. In der Realität kann selbst bei der sorgfältigen Züchtung von Einkristallen nicht verhindert werden, dass Versetzungen gebildet werden. Versetzungen können sich in zwei Teilversetzungen aufspalten und führen dadurch zu Stapelfehlern. Die Stapelfehlerenergie SFE ist ein Maß dafür, wie weit sich die Versetzung dabei aufspalten kann. Je höher die Stapelfehlerenergie ist, desto geringer ist der Abstand zwischen den Teilversetzungen. Durch die Aufspaltung in zwei Teilversetzungen werden Schraubenversetzungen am Quergleiten gehindert, wodurch weniger dynamische Erholung möglich ist. Je geringer der Abstand zwischen den Teilversetzungen ist, desto eher können sie sich wieder vereinigen (einschnüren) wodurch Quergleiten wieder möglich ist. Aluminium ist z.B. einer der Werkstoffe mit der höchsten Stapelfehlerenergie und neigt daher stark zur dynamischen Erholung.

### 1.1.3. Zweidimensionale Fehlstellen

Technische metallische Werkstoffe bestehen in den allermeisten Fällen nicht aus einem einzigen Kristall, sondern aus einer Vielzahl verschieden orientierter Körner. Die Korngrenzen, also der Übergangsbereich zwischen zwei Körnern, werden zu den zweidimensionalen Kristallbaufehlern gezählt. Zur Charakterisierung von Korngrenzen muss der Orientierungsunterschied der beiden benachbarten Körner und die Lage der Grenzfläche zu den Körnern angegeben werden. Der Orientierungsunterschied kann durch eine Kombination aus Drehachse und Drehwinkel angegeben werden. Steht die Drehachse normal auf die Grenzfläche, so liegt der Idealtyp einer Drehkorngrenze vor (Abbildung 6a(a)). Liegt die Drehachse in der Grenzfläche, so handelt es sich um eine Kippkorngrenze (Abbildung 6a(b,c)).

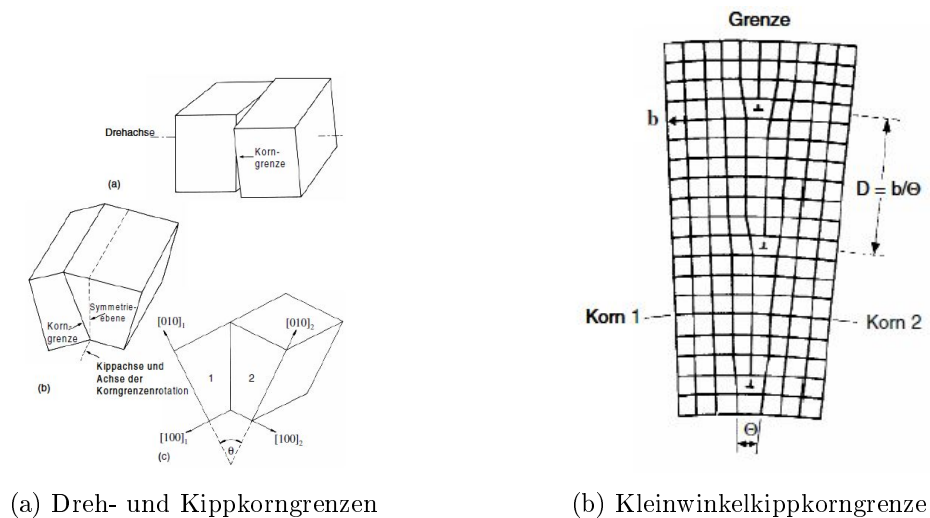


Abbildung 6: Korngrenzen [1]

Die Struktur von Korngrenzen kann bei kleinen Orientierungsunterschieden durch eine Schar von Versetzungen beschrieben bzw. auch als solche im Transmissionselektronenmi-

kroskop (TEM) beobachtet werden. Dabei ergibt sich die Missorientierung zwischen zwei Körnern zu (siehe Abbildung 6b)

$$\frac{b}{D} = 2 \sin \frac{\theta}{2} \approx \theta \quad (3)$$

Dieses Modell der Korngrenze verliert jedoch ab einem Orientierungsunterschied von  $15^\circ$  seine Gültigkeit, da sich ab dieser Missorientierung die Versetzungskerne überlappen. Aus diesem Grund werden Korngrenzen unter  $15^\circ$  als Kleinwinkelkorngrenzen und darüber als Großwinkelkorngrenzen definiert.

#### 1.1.4. Dreidimensionale Fehlstellen

Als letzter Kristallbaufehler sei der Einschluss einer zweiten Phase angeführt. Technische Metalle sind niemals rein. Ein Teil der Fremdatome ist im Matrixwerkstoff gelöst. Bei höheren Konzentrationen an Fremdatomen wird die Löslichkeitsgrenze überschritten und es wird neben der Matrixphase eine sogenannte Ausscheidungsphase gebildet. Dies kann entweder bei der Erstarrung aus der Schmelze (Primärausscheidungen) oder während des Abkühlvorganges in der festen Phase (Sekundärausscheidungen) passieren. Diese Ausscheidungen haben aufgrund ihrer chemischen Zusammensetzung und Struktur andere mechanische Eigenschaften als der Matrixwerkstoff. Feine Ausscheidungen können gezielt dazu verwendet werden, den Werkstoff zu verfestigen. Grobe Ausscheidungen können jedoch auch negative Auswirkungen wie eine verminderte Zähigkeit mit sich bringen. In dieser Arbeit sind große Ausscheidungen insofern von Interesse, da sie eine inhomogene Verformung der umliegenden Matrix verursachen.

### 1.2. Kaltverformung

(Dieses Kapitel ist in Anlehnung an [1, Kapitel 6])

Es sind die elastische und die plastische Verformung zu unterscheiden. Bei der elastischen Verformung werden die Atome durch die wirkenden Kräfte von ihrer Gleichgewichtslage ausgelenkt. Werden die Kräfte wieder weggenommen, so bewegen sich die Atome wieder in die ursprüngliche Lage zurück. Im Gegensatz dazu sind plastische Verformungen irreversibel, d.h. auch nach Wegnehmen der äußeren Kräfte wird die Ausgangslage nicht mehr erreicht. Versuche an Einkristallen ergeben, dass es bevorzugt an dichtest gepackten Ebenen des Kristallgitters zu Abgleitungen kommt. Theoretisch lässt sich berechnen bei welchen Schubspannungen die Gitterebenen aneinander abgleiten sollten,

$$\tau_{max} = \frac{G b}{2\pi d} \quad (4)$$

wobei  $G$  der Schubmodul,  $b$  der Abstand zweier Atome in einer Ebene und  $d$  der Ebenenabstand ist. Daraus ergeben sich theoretisch Schubspannungen von  $0,9 \cdot 10^9$ MPa für Aluminiumeinkristalle, was in etwa drei Größenordnungen größer als der gemessene Wert ist. Tatsächlich geschieht plastische Verformung jedoch nicht durch Abgleiten der Ebenen als Ganzes, sondern – wie schon im Kapitel 1.1 angedeutet – durch Bewegung von

Versetzungen. Ein fundamentaler Zusammenhang zwischen der mittleren Versetzungsgeschwindigkeit  $v$  und makroskopischer Scherrate  $\dot{\gamma}$  ist

$$\dot{\gamma} = \rho b v \quad (5)$$

Die auf eine Versetzung einwirkende Kraft je Längeneinheit  $\vec{K}$  kann mit der Peach-Koehler-Gleichung beschrieben werden.

$$\vec{K} = (\boldsymbol{\sigma} \cdot \vec{b}) \times \vec{s} \quad (6)$$

Dabei ist  $\boldsymbol{\sigma}$  ein beliebiger Spannungszustand und  $\vec{b}$  der Burgersvektor des Linienelements  $\vec{s}$ . Damit Versetzungen entlang einer Gitterebene verschoben werden können, muss eine kritische Schubspannung, die sogenannte Peierls-Spannung, überschritten werden.

$$\tau_p = \frac{2G}{1-\nu} \exp\left(-\frac{2\pi}{(1-\nu)} \frac{d}{b}\right) \quad (7)$$

Die kritische Schubspannung  $\tau_p$  hängt linear vom Schubmodul  $G$ , jedoch exponentiell vom Ebenenabstand  $d$  und dem Burgersvektor  $b$  ab ( $\nu$  ist die Poissonzahl). Für eine geringe kritische Schubspannung sollte der Ebenenabstand möglichst groß sein währenddessen der Burgersvektor möglichst klein sein soll. Der Ebenenabstand lässt sich aus den Miller-Indizes  $hkl$  und dem Gitterparameter  $a$  (also der Größe der Einheitszelle) berechnen. Die Miller-Indizes einer Netzebene sind die kleinsten ganzzahligen Reziprokwerte der Achsenabschnitte, die durch das Schneiden der Netzebene mit den Achsen der Einheitszelle erzeugt werden können. Für den Fall eines kubischen Kristalls entspricht dies der auf ganze Zahlen erweiterten Koordinaten der Ebenennormalen. Die Erklärung dazu befindet sich in [1, Kapitel 2.3].

$$d = \frac{a}{\sqrt{h^2 + k^2 + l^2}} \quad (8)$$

Für kfz-Kristalle ist der Ebenenabstand für die  $\{111\}$ -Ebenen am größten. Innerhalb der Ebenen ist der Burgersvektor in jenen Richtungen am geringsten, in denen sich die Atome berühren. In kfz-Kristallen sind dies die  $\langle 110 \rangle$ -Richtungen (Abbildung 7). Die Kombination aus Gleitebene und Gleitrichtung ergeben ein Gleitsystem. Da kfz-Kristalle 4  $\{111\}$ -Ebenen haben in denen drei  $\langle 110 \rangle$ -Richtungen vorhanden sind, ergibt dies 12 Gleitsysteme.



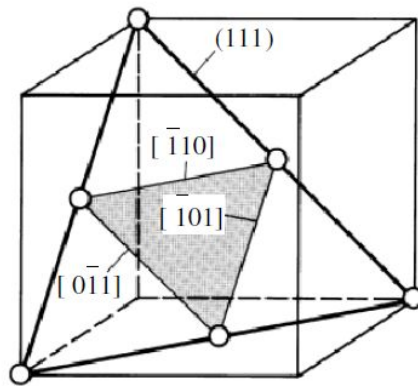


Abbildung 7: kfz-Gleitsysteme [1]

In Gleichung 7 wird die kritische Schubspannung ausgedrückt. Im Allgemeinen sind jedoch äußere Kräfte beliebig zu den Gleitsystemen des Kristalls orientiert. Für den Fall von Zugbelastung kann die resultierende Schubspannung berechnet werden durch

$$\tau = \sigma \cos \kappa \cos \lambda = m\sigma \quad (9)$$

wobei  $\kappa$  und  $\lambda$  die Winkel zwischen Zugrichtung und Ebenennormalen bzw. Gleitrichtung sind (Abbildung 8).

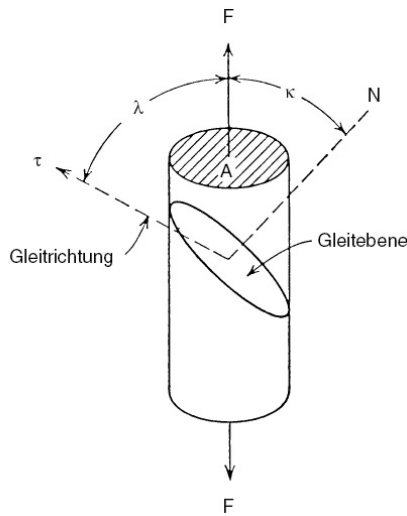


Abbildung 8: Schmid'sche Schubspannung [1]

Der Schmidfaktor  $m$  kann die Werte  $(-0,5 \leq m \leq 0,5)$  einnehmen. Sofern mehrere Gleitsysteme vorhanden sind, erhält jenes mit dem betragsmäßig höchsten Schmidfaktor die höchste Schubspannung. Aus diesem Grund ergibt sich ein anisotropes Verhalten von Einkristallen. Für die meisten technischen Betrachtungen spielt dies jedoch keine Rolle,

da in dem betrachteten Volumen eine Vielzahl von verschieden orientierten Körnern vorliegt und sich somit ein quasi-isotropes Verhalten ergibt.

Versetzungen verursachen in ihrer Umgebung eine Verzerrung des Kristallgitters. Diese Verzerrung führt dazu, dass selbst bei Abwesenheit einer makroskopischen Belastung ein Spannungsfeld vorhanden ist. Am Beispiel einer Stufenversetzung bedeutet dies, dass im Bereich, wo eine zusätzliche Halbebene vorhanden ist, eine Druckspannung herrscht, währenddessen auf der Seite mit fehlender Halbebene Zugspannung auftritt.

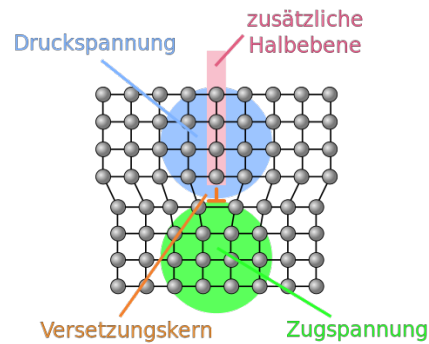


Abbildung 9: Spannungsverteilung Stufenversetzung [2]

Wie in der Peach-Koehler-Gleichung (Gleichung 6) beschrieben, wird in einem Spannungsfeld eine Kraft auf Versetzungen ausgeübt. Dies bedeutet, dass Versetzungen gegenseitig aufeinander einwirken ohne sich dabei berühren zu müssen. Die resultierende Kraft zwischen zwei Versetzungen ist von der Lage zueinander und von der Richtung des Burgersvektors der Versetzungen abhängig. Versetzungen mit parallelen Burgersvektoren haben das Bestreben sich übereinander bzw. unter  $45^\circ$  anzuordnen (Abbildung 10). Ordnen sich die Versetzungen übereinander an so überlagert sich Druckzone mit Zugzone, wodurch die gespeicherte elastische Energie minimiert wird. Würden sich die Versetzungen nebeneinander anlagern, so würde der umgekehrte Fall eintreten. Versetzungen mit antiparallelen Burgersvektoren verhalten sich umgekehrt und können sich nebeneinander anordnen, was dazu führt, dass sie sich gegenseitig auslöschen können.

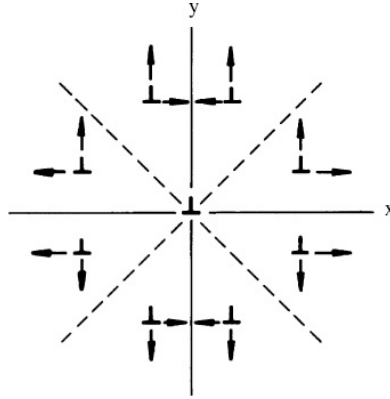


Abbildung 10: Kraftwirkung Versetzungen [1]

Die Wechselwirkungen zwischen den Versetzungen bewirken nicht nur eine bevorzugte Ausrichtung, sondern verursachen auch eine zusätzliche Kraft, wenn sich Versetzungen aneinander vorbeibewegen. Die daraus resultierende Spannung  $\tau_{pass}$  wird als Passierspannung bezeichnet,

$$\tau_{pass} = \alpha_1 \frac{Gb}{d} = \alpha_1 Gb\sqrt{\rho_p} \quad (10)$$

wobei  $d$  der mittlere Abstand von den parallelen Primärversetzungen ist. Zusätzlich gibt es jedoch auch Versetzungen auf anderen Gleitebenen, welche von den Primärversetzungen geschnitten werden müssen. Dies führt zu einer zusätzlichen Schneidespannung  $\tau_s$ , welche von der Versetzungsdichte dieser sogenannten Waldversetzungen abhängt.

$$\tau_s = \frac{1}{2} Gb\sqrt{\rho_w} \quad (11)$$

Geht man davon aus, dass die Primärversetzungen und Waldversetzungen in einem bestimmten Verhältnis stehen, so kann die gesamte Schubspannung zusammengefasst werden.

$$\tau_0 = \alpha_1 Gb\sqrt{\rho_p} + \frac{1}{2} Gb\sqrt{\rho_w} = \alpha Gb\sqrt{\rho} \quad (12)$$

Dabei sind  $\alpha_1$  und  $\alpha$  Geometriefaktoren. Der Geometriefaktor  $\alpha$  ist in der Größenordnung 0,5.

Bei Scherversuchen an Einkristallen sieht man, dass mit zunehmender Abscherung die Schubspannung ansteigt (Abbildung 11). Dabei können drei Bereiche unterschieden werden. Im Bereich 1 können die im Kristall vorhandenen Versetzungen einfach abgleiten (easy-glide-Bereich). Bereich 1 kommt jedoch nur dann vor, wenn der Kristall so ausgerichtet ist, dass nur ein Gleitsystem für die plastische Verformung notwendig ist. In polykristallinen Werkstoffen kommt aus diesem Grund der easy-glide-Bereich nicht vor. Im Bereich 2 kommt es zu einer starken Verfestigung, wobei sich die Versetzungsdichte stark erhöht. Eine Erklärung hierfür ist der Frank-Read-Mechanismus. Wird eine Versetzungslinie an beiden Enden durch ein Hindernis festgehalten und aufgrund einer Schubspannung ausgewölbt, so kann dies zur Bildung einer neuen Versetzung führen. Mögliche

Hindernisse sind z.B. Ausscheidungen einer zweiten Phase oder Versetzungslinien aus einer anderen Gleitebene. Im dritten Bereich flacht die Schubspannungskurve wieder ab. Aufgrund der erhöhten Schubspannung wird es für Schraubenversetzungen möglich ihre Gleitebene zu verlassen um etwaige Hindernisse zu umgehen. Außerdem ist es möglich, dass es auf anderen Ebenen Versetzungen mit antiparallelem Burgersvektor gibt mit denen eine gegenseitige Auslöschung vollzogen werden kann. Wenn dies während der Verformung geschieht, so wird dieser Vorgang dynamische Erholung genannt.

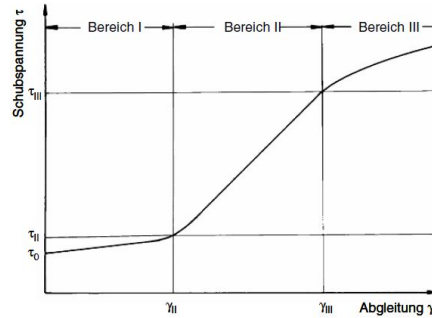


Abbildung 11: Verfestigung Einkristall [1]

### 1.3. Erholung

Nach der Kaltverformung stellt sich eine Mikrostruktur mit hoher Versetzungsdichte ein, in der ein Teil der Versetzungen statistisch verteilt und ein Teil in Zellwänden angeordnet ist. Der Anteil an Versetzungen in den Zellwänden steigt mit zunehmender Kaltverformung und der dabei auftretenden dynamischen Erholung. Die dynamische Erholung wird stark von der Stapelfehlerenergie beeinflusst. Werkstoffe mit einer hohen Stapelfehlerenergie wie Aluminium bilden wenige Stapelfehler, die das Quergleiten von Schraubenversetzungen behindern könnten, wodurch bereits nach der Kaltverformung eine ausgeprägte zellenförmige Mikrostruktur vorhanden ist. Bei einer nachfolgenden Glühung kommt es zur Steigerung der Leerstellenkonzentration, wodurch es nun auch den Stufenversetzungen möglich ist durch Klettern ihre Gleitebene zu verlassen und sich mit anderen Versetzungen auszulöschen. Jene Versetzungen, die sich nicht gegenseitig auslösen, lagern sich verstärkt in den Zellwänden an. Befindet sich im untersuchten Bereich ein Überschuss an Stufenversetzungen mit bestimmten Burgersvektoren, so führt dieser Erholungsvorgang zur Bildung von Kippsubkorn Grenzen wie in Abbildung 12 schematisch dargestellt. Erstreckt sich dieser Überschuss über einen größeren Bereich, so bedeutet das, dass ein gerichteter Orientierungsgradient innerhalb des untersuchten Bereiches vorhanden ist. In der schematischen Darstellung wird dies durch eine Krümmung der Gitterebene illustriert. Im Umkehrschluss bedeutet dies, dass, wenn das Gitter aufgrund mechanischer Belastung gekrümmt wird, zusätzliche Versetzungen gebildet werden müssen. Diese Versetzungen werden geometrisch notwendige Versetzungen genannt. Die Missorientierung von Subkörnern hängt also vom Orientierungsgradienten des Gitters ab. Die reale Darstellung in Abbildung 13 zeigt, wie sich die Struktur der Zellwände während der Glühung

von Aluminium bei 250°C für 2 Minuten verändert. Die Zellwände werden schärfer und es bilden sich Subkorn Grenzen.

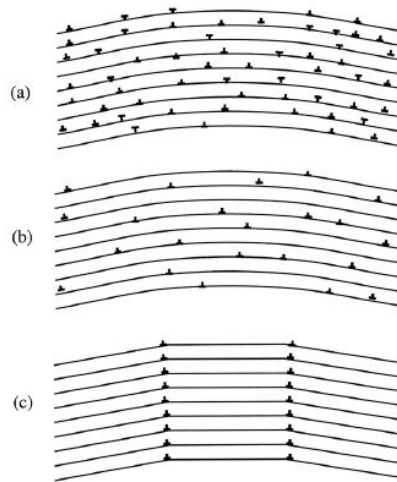


Abbildung 12: Polygonisieren – theoretisches Modell [3]

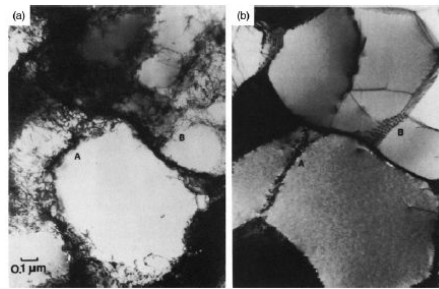


Abbildung 13: Polygonisieren – real [3]

#### 1.4. Rekristallisation

Nach Humphreys [3] ist Rekristallisation das Wandern von Grenzflächen aufgrund einer treibenden Kraft. Die treibende Kraft ist der Energieunterschied zwischen zwei benachbarten Teilvolumina des Mikrogefüges. Der Energieunterschied kommt aufgrund unterschiedlicher Versetzungsdichten und der damit einhergehenden gespeicherten elastischen Energie je Volumeneinheit  $G^i$  zustande. Dabei ist  $\rho_i$  die entsprechende Versetzungsdichte,  $G$  der Schubmodul und  $b$  der Burgersvektor des Materials.

$$G^i \approx 0,5\rho_i Gb^2 \quad (13)$$

Betrachtet man zwei Bereiche, welche durch eine Grenzfläche  $a$  getrennt sind, so ergibt sich bei Verschieben der Grenzfläche um die Strecke  $dx$  normal zur Grenzfläche ein

Gewinn an freier Enthalpie  $dG$ :

$$dG = dx(G^A - G^B)a \quad (14)$$

Die treibende Kraft ergibt sich zu  $-\frac{dG}{dx}$  und der Druck auf die Grenzfläche ist  $\frac{F}{a}$  und somit

$$P = -\frac{1}{a} \frac{dG}{dx} = G^B - G^A = \Delta G \left[ \frac{J}{m^3} \right] \quad (15)$$

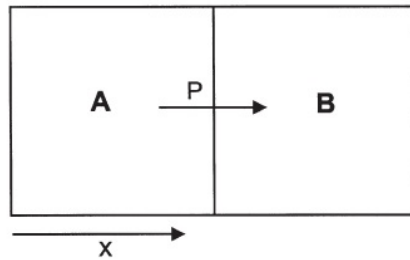


Abbildung 14: Resultierender Druck an Grenzfläche [3]

Durch diese treibende Druck kommt es zum Wandern der Grenzfläche mit der Geschwindigkeit [3]

$$v = M P \quad (16)$$

wobei M die Mobilität der Grenzfläche ist. Sie hängt von der Temperatur und der Struktur der Grenzfläche ab (siehe Abbildung 15). Zusammengefasst bedeutet das, dass für die Rekristallisation eine bewegliche Grenzfläche vorhanden sein muss, welche zwei Bereiche unterschiedlicher Versetzungsdichte trennt.

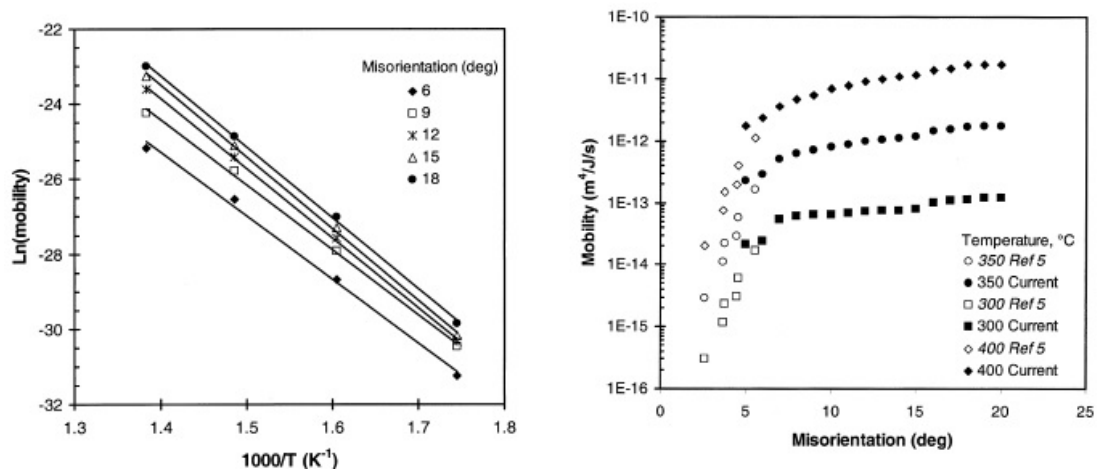


Abbildung 15: Mobilität als Funktion der Subkornmisorientierung [4]

Dies deckt sich auch mit der Aussage von Gottstein [1], wonach für die Keimbildung von Rekristallisation drei Kriterien erfüllt werden müssen:

- **Thermodynamische Instabilität**

Ähnlich wie bei der Erstarrung aus der Schmelze muss ein Keim mit einem kritischen Radius  $r_c$  vorhanden sein, wodurch ein zusätzliches Wachstum zu einer Verringerung der freien Enthalpie führt.

$$r_c = \frac{2\gamma}{P} = \frac{4\gamma}{\rho G b^2} \quad (17)$$

Die kritische Keimgröße hängt von der spezifischen Oberflächenenergie  $\gamma$  des Keimes und der treibenden Kraft  $P$ , und somit der Versetzungsdichte  $\rho$ , ab. Da die treibende Kraft im Vergleich zur Erstarrung sehr gering ist, reicht die thermische Fluktuation für die Keimbildung nicht aus. Stattdessen muss sich ein Keim durch vorangegangene Erholungsvorgänge gebildet haben.

- **Mechanische Instabilität**

Es muss ein lokales Ungleichgewicht der treibenden Kraft herrschen, damit die Korngrenze eine definierte Bewegungsrichtung hat. Diese Bedingung wird durch ungleiche Verteilung der Versetzungsdichte erfüllt.

- **Kinetische Instabilität**

Die Grenzfläche des Keims muss beweglich sein, was nur bei Großwinkelkorngrenzen der Fall ist.

Da die kritische Keimgröße von großer Bedeutung für diese Arbeit ist, soll Gleichung 17 exemplarisch für die Versetzungsdichte von kaltverformtem Aluminium ausgewertet werden. Es wird eine realistische Versetzungsdichte von  $10^{15} m^{-2}$  angesetzt. Der Schubmodul und der Burgersvektor von Aluminium sind 26 GPa bzw.  $2,86 \cdot 10^{-10} m$ . Für die spezifische Oberflächenenergie von Großwinkelkorngrenzen wird in etwa  $0,5 J/m^2$  in der Literatur angegeben [3] [1].

$$r_c = \frac{4 \cdot 0,5}{10^{15} \cdot 26 \cdot 10^9 (2,86 \cdot 10^{-10})^2} = 0,94 \cdot 10^{-6} m \approx 1 \mu m \quad (18)$$

Dabei handelt es sich nur um einen Richtwert, da die tatsächliche Versetzungsdichte variiert.

#### 1.4.1. Keimbildung

Es gibt bevorzugte Bereiche an denen alle drei Bedingungen für Rekristallisation erfüllt werden. Nach Humphreys [3] sind dies bevorzugt Stellen an denen Inhomogenitäten in der verformten Mikrostruktur auftreten. Dabei werden zwei Möglichkeiten unterschieden:

- **Keimbildung durch Bewegung von Korngrenzen**

Es kommt zum Ausbauchen der vorhandenen Korngrenze in Richtung jenes Kornes mit der höheren Versetzungsdichte. In diesem Fall hat der Keim die Orientierung des Kornes mit der niedrigeren Versetzungsdichte. Dieser Vorgang wird SIBM (strain induced boundary migration) genannt. Diese Art der Keimbildung spielt besonders bei geringen Verformungen die dominierende Rolle.

- **Erzeugte Keimstellen**

Die Keime werden während der Verformung erzeugt. Humphreys [3] sagt dazu:

It is now established beyond reasonable doubt that recrystallization originates from **dislocation cells or subgrains which are present after deformation**.

Humphreys [3] erklärt die Keimbildung anhand folgender Phänomene:

- Die Orientierung des Keims ist bereits in der verformten Struktur vorhanden. Es werden keine neuen Orientierungen gebildet.
- Keimbildung geschieht durch Wachsen von Subkörnern. Dabei wird beobachtet, dass die Erholung in der Nähe der Keimstellen wesentlich schneller als im restlichen Material vonstattengeht.
- Damit eine Großwinkelkorngrenze gebildet werden kann, muss ein Orientierungsgradient vorhanden sein. Dazu kommt, dass Bereiche mit einem Orientierungsgradienten eine hohe gespeicherte Energie haben, da zusätzliche geometrisch notwendige Versetzungen vorhanden sein müssen.

Die Schlussfolgerung von Humphreys [3] lautet daher

Nucleation of recrystallization in such a situation can therefore be considered as no more than discontinuous subgrain growth at sites of high strain energy and orientation gradient.

Keimbildung entsteht an Stellen von inhomogener Verformung. Die Keimbildungsstellen spielen für Werkstoffe insofern eine Rolle, als die daraus wachsenden Körner die Orientierung der Keime haben. Je nachdem wie die Keime orientiert sind, kann sich dadurch eine Textur des rekristallisierten Gefüges bilden oder auch nicht. Folgende Stellen der verformten Mikrostruktur können als Keimbildungsstellen agieren:

- **Korngrenzen**

Im Gegensatz zu SIBM können an Korngrenzen auch Keime mit einer zu den beiden Körnern unterschiedlichen Orientierung entstehen. Dies ist besonders bei größeren Verformungen der Fall. Da in den beiden Körnern die primären Gleitsysteme unterschiedlich orientiert sind, wäre eine Verformung durch einfaches Abgleiten nicht kompatibel und es müssen aus diesem Grund zusätzliche Gleitsysteme aktiviert werden. Dadurch kommt es in der Nähe der Korngrenzen zu Orientierungsänderungen.

- **Übergangsbänder**

Übergangsbänder trennen Bereiche unterschiedlicher Verformung innerhalb eines Kornes. In diesem Bereich befinden sich große Orientierungsgradienten, wodurch Keime gebildet werden können.

- **Scherbänder**

Scherbänder sind schmale Bereiche intensiver Scherung mit lokal hohen Dehnungen.



- **Harte Teilchen**

In der Nähe von großen harten Ausscheidungen bilden sich Verformungszonen, welche für die Keimbildung prädestiniert sind. Die Keimbildung an Teilchen wird PSN (particle stimulated nucleation) genannt. In weiterer Folge wird noch genauer auf diese Art der Keimbildung eingegangen.

## 1.5. Particle Stimulated Nucleation PSN

(Dieses Kapitel ist in Anlehnung an [3, Kapitel 9.3])

PSN wird in vielen wichtigen Legierungen wie Al, Fe, Cu und Ni beobachtet. Dabei kommt es zur Keimbildung an Teilchen ab einer Größe von rund  $1\mu\text{m}$  [3]. Da viele technische Materialien Teilchen von dieser Größe beinhalten ist PSN von großem technischem Interesse. Die Orientierung der Keime an Teilchen unterscheidet sich von denen an Stellen des Gefüges des von Teilchen unbeeinflussten Matrixwerkstoffes, wodurch die Textur des rekristallisierten Gefüges beeinflusst wird. Da die Anzahl und Größe der Teilchen modifizierbar sind, kann dadurch das rekristallisierte Gefüge gesteuert werden.

### 1.5.1. Mechanismus von PSN

Informationen über den Keimbildungsmechanismus erhält man durch in-situ Versuche mittels Hochspannungselektronenmikroskopie (HVEM). Untersuchungen an gewalzten Aluminiumproben ergaben, dass Rekristallisation an Subkörnern in der Nähe der Teilchen startet (Abbildung 16). Dabei kommt es zur schnellen Subkornwanderung. Der Ursprung muss nicht an der Oberfläche des Teilchens sein. Die Korngrenze kann aufhören zu wachsen, wenn die Verformungszone um das Teilchen rekristallisiert ist bzw. wenn es von einem schnelleren Korn umwachsen wird. Abbildung 16 zeigt ein Beispiel von Keimbildung an einem Teilchen wo dies der Fall ist.

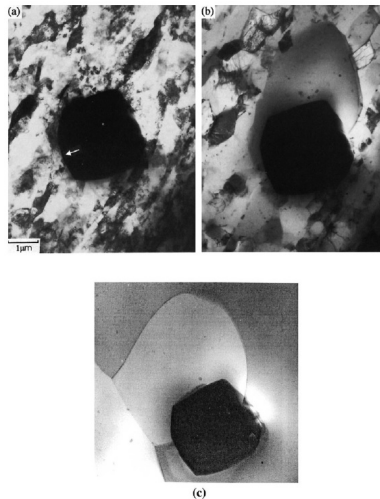


Abbildung 16: HVEM Untersuchung von PSN in einer Al-Si-Legierung [3]

Betrachtet man die Größe der Subkörner und die maximale Missorientierung innerhalb der Verformungszone, so erkennt man, dass die Subkörner mit zunehmender Glühzeit wachsen und die maximale Missorientierung zur Matrix abnimmt (Abbildung 17).

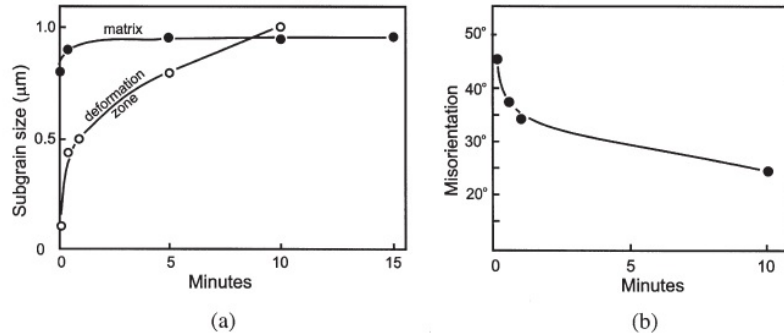


Abbildung 17: Subkornwachstum in der Nähe eines harten Teilchens während in-situ Glühversuch [3]

Obwohl die maximale Missorientierung relativ zur unrotierten Matrix innerhalb der Verformungszone abnimmt, steigt beim Subkornwachstum innerhalb eines Orientierungsgradienten die Missorientierung zwischen den einzelnen Subkörnern. Dadurch können Grenzflächen mit einer Missorientierung von  $15^\circ$  entstehen, was der Bildung von Korngrenzen entspricht. Ob diese Korngrenzen wachsen können, hängt von der thermischen Stabilität (Gleichung 17) ab. D.h. es muss eine kritische Keimgröße vorhanden sein, damit der Keim wachsen kann. Experimentelle Untersuchungen (Abbildung 18) ergeben einen Zusammenhang zwischen Teilchendurchmesser, Verformung und der Bildung von Keimen an Teilchen. Dabei wird klar, dass bei kleineren Teilchen mehr Verformung als bei großen Teilchen notwendig ist, damit Keimbildung stattfinden kann.

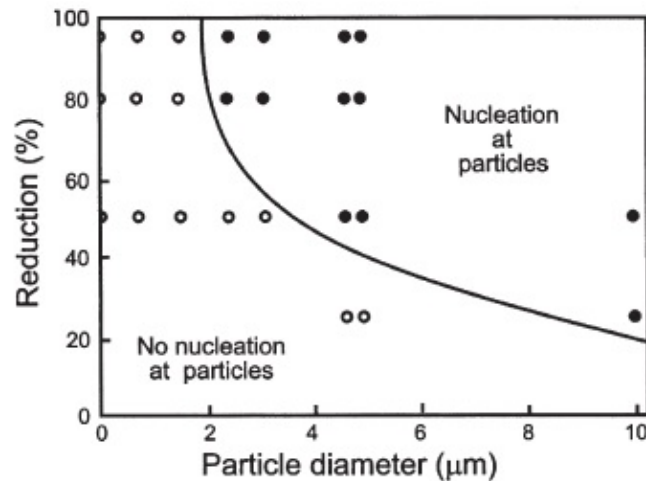


Abbildung 18: Keimbildung als Funktion der Dehnung und des Teilchendurchmessers [3]

### 1.5.2. Textur nach PSN

Nachdem sich die Orientierung neuer Körner auf Orientierungen der verformten Mikrostruktur beschränkt, müsste nach der Rekristallisation eine Textur vorhanden sein. Tatsächlich lässt sich an Versuchen mit Einkristallen nachweisen, dass die Orientierung der PSN Keime nicht zufällig ist. Abbildung 19 zeigt die Orientierungen von Matrix (M) und den PSN Keimen (quadratische Symbole). Obwohl die Orientierungsverteilung der Keime breiter als die der Matrix ist, ist trotzdem eine starke Textur vorhanden.

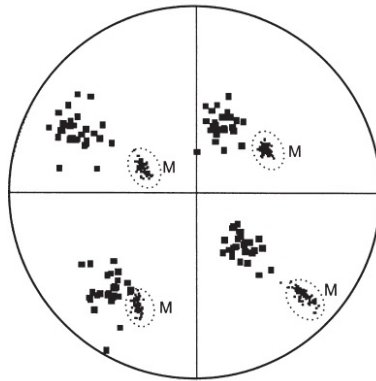


Abbildung 19: 111 Polfigur eines teilweise rekristallisierten Ni-SiO<sub>2</sub> Einkristalls [3]

Untersuchungen an polykristallinen Proben mit großen Teilchen ergeben, dass die Keime bzw. Körner schwach texturiert oder zufällig orientiert sind (z.B. in [9]). Das Verformungsverhalten von polykristallinen Proben ist komplizierter als bei Einkristallen, wodurch es bereits innerhalb des Kornes zu einer breiteren Verteilung der Orientierungen kommt. Wird weiter über alle Körner gemittelt, so wird eine gleichmäßige Verteilung der Keimorientierung erreicht.

## 2. Kontinuumsmechanik

(Dieses Kapitel ist in Anlehnung an [10, Theory Manual])

Um die Verformung eines Körpers zu beschreiben, bedient man sich der Kontinuumsmechanik. Dabei wird vom mikroskopischen Aufbau des Materials, wie er in Kapitel 1 erörtert wurde, abgesehen.

### 2.1. Deformationsgradient

Ziel ist es die zeitliche Änderung der Position  $\vec{x}$  eines Materialteilchens, welches sich ursprünglich an der Stelle  $\vec{X}$  befindet, in Abhängigkeit von der Belastung zu beschreiben. Dabei stellt  $\vec{X}$  die Position des Teilchens in der Ausgangskonfiguration und  $\vec{x}$  in der laufenden Konfiguration dar. Da die Belastung eine Funktion der Zeit ist, kann folgender Ausdruck gefunden werden:

$$\vec{x} = \vec{x}(\vec{X}, t) \quad (19)$$

Werden nun zwei benachbarte Materialteilchen der Ausgangskonfiguration an den Positionen  $\vec{X}$  und  $\vec{X} + d\vec{X}$  betrachtet, so findet man in der laufenden Konfiguration:

$$d\vec{x} = \frac{\partial \vec{x}}{\partial \vec{X}} d\vec{X} \quad (20)$$

wobei die Matrix

$$\mathbf{F} = \frac{\partial \vec{x}}{\partial \vec{X}} \quad (21)$$

als Deformationsgradient bezeichnet wird. Dieser Tensor ist eine lineare Abbildung von Vektoren aus der Ausgangskonfiguration in die laufende Konfiguration. Da das Materialverhalten jedoch nicht von der Starrkörperbewegung des Materialteilchens im Raum, sondern von der Dehnung abhängt, müssen diese beiden Teile der Bewegung aufgespalten werden. Wird eine Teststrecke  $d\vec{X}$  betrachtet, so kann dessen Länge mit

$$dL^2 = d\vec{X}^T d\vec{X} \quad \text{und} \quad dl^2 = d\vec{x}^T d\vec{x} \quad (22)$$

in der Ausgangs- bzw. in der laufenden Konfiguration berechnet werden. Damit kann das Dehnungsverhältnis der betrachteten Teststrecke definiert werden.

$$\lambda = \frac{dl}{dL} = \sqrt{\frac{d\vec{x}^T d\vec{x}}{d\vec{X}^T d\vec{X}}} \quad (23)$$

Bei reiner Starrkörperbewegung ist  $\lambda$  gleich eins. Unter Verwendung des Deformationsgradienten kann die Länge der Teststrecke in der laufenden Konfiguration ausgedrückt werden.

$$d\vec{x}^T d\vec{x} = d\vec{X}^T \mathbf{F}^T \mathbf{F} d\vec{X} \quad (24)$$

$$\lambda^2 = \frac{d\vec{X}^T}{\sqrt{d\vec{X}^T d\vec{X}}} \mathbf{F}^T \mathbf{F} \frac{d\vec{X}^T}{\sqrt{d\vec{X}^T d\vec{X}}} = \vec{N}^T \mathbf{F}^T \mathbf{F} \vec{N} \quad (25)$$

Dabei ist  $\vec{N}$  der Einheitsvektor in Richtung der Teststrecke  $d\vec{X}$ . Wird nun  $\vec{N}$  dermaßen variiert, dass  $\lambda$  extremal wird, erhält man dessen Eigenwerte und die dazugehörigen Eigenvektoren, welche den Dehnungshauptachsen entsprechen. Die Dehnungshauptachsen lassen sich mit Hilfe des Deformationsgradienten von der Ausgangskonfiguration in die laufende Konfiguration umrechnen.

$$\vec{n}_I = \frac{\mathbf{F}}{\lambda_I} \vec{N}_I \quad (26)$$

Da die Dehnungshauptachsen beider Konfigurationen orthogonal sind, muss dieser Umrechnungsfaktor für alle drei Achsen derselbe sein, nämlich der Rotationstensor. Eine Teststrecke, die in einer Hauptdehnungsachse in der Ausgangskonfiguration liegt, kann also wie folgend beschrieben werden:

$$d\vec{x}_1 = \lambda_1 \mathbf{R} d\vec{X}_1 \quad (27)$$

Für alle drei Hauptachsen zusammengefasst ergibt das

$$d\vec{x} = \mathbf{V} \mathbf{R} d\vec{X} \quad (28)$$

Der Vergleich mit der Definition des Deformationsgradienten ergibt

$$\mathbf{F} = \mathbf{V} \mathbf{R} \quad (29)$$

Dieser Ausdruck ist als polare Zerlegung des Deformationsgradienten bekannt. Dabei wird der Deformationsgradient in den Verzerrungstensor  $\mathbf{V}$  und den Rotationstensor  $\mathbf{R}$  zerlegt. Für die Berechnung von Spannungen wird nur der Verzerrungstensor benötigt. Der Rotationstensor ist für die Berechnung der Verformung erst dann von Interesse, wenn anisotrope Werkstoffmodelle betrachtet werden. In dieser Arbeit werden aus Zeitgründen rein isotrope Materialmodelle verwendet. Tatsächlich befindet sich das Modell auf einer Längenskala, wo das gesamte Volumen aus einem einzelnen Korn mit einer bestimmten Orientierung besteht, wodurch die Materialrotationen tatsächlich von Bedeutung sind. Auf diese Tatsache wird in der Modellerstellung eingegangen.

## 2.2. Verzerrungen

Es wurde bereits das Dehnungsverhältnis als Maß für die Verzerrung eingeführt. Für die Verformung von Metallen ist dieses Dehnungsmaß wenig geeignet, da durch die geringe elastische Verformung erst ab der 4. Dezimalstelle eine Abweichung von null stattfindet. Aus diesem Grund werden abhängig von der Anwendung verschiedene Verzerrungsmaße eingeführt, wobei diese eine Funktion des Dehnungsverhältnisses sind.

$$\varepsilon = f(\lambda) \quad (30)$$

$$\varepsilon = f(1) + (\lambda - 1) \frac{df}{d\lambda} + \frac{1}{2!} (\lambda - 1)^2 \frac{d^2 f}{d\lambda^2} \dots \quad (31)$$

Dabei werden folgende Einschränkungen getroffen:

$$\begin{aligned} f(1) &= 0 \\ \frac{df}{d\lambda(1)} &= 1 \\ \frac{df}{d\lambda} &> 0 \text{ für } \lambda > 1 \end{aligned}$$

D.h. wenn keine Dehnung stattfindet (Dehnungsverhältnis 1) dann soll die Verzerrung 0 sein. Kleine Dehnungen sollen bei jedem Verzerrungsmaß gleiche Ergebnisse liefern und Zugverformung führt zu positiven Verzerrungen.

Beispiele:

Nominale Dehnung:  $f(\lambda) = \lambda - 1$

Wird für steife Proben verwendet.

Logarithmische Dehnung:  $f(\lambda) = \ln(\lambda)$

Findet Verwendung in der Metallplastizität.

Green Dehnung:  $f(\lambda) = \frac{1}{2}(\lambda^2 - 1)$

Ist ein praktisches Verzerrungsmaß für die Berechnung von Problemen mit großen Verschiebungen aber kleinen Dehnungen.

Es gibt weitere Beispiele für Verzerrungsmaße. Maßgeblich für deren Zweckmäßigkeit ist zum einen wie einfach sie sich aus den Verschiebungen berechnen lassen und zum anderen wie genau das Verzerrungsmaß in Bezug auf die Materialgesetze ist. So wird z.B. die logarithmische Dehnung für plastische Verformung von Metallen verwendet, da sich damit sowohl das Verhalten bei Zug- als auch bei Druckbelastung gut beschreiben lässt. Oben genannte Beispiele beziehen sich auf den eindimensionalen Fall. Bei der Verallgemeinerung auf drei Dimensionen gilt z.B. für die Green Dehnung:

$$\boldsymbol{\varepsilon}^G = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}) \quad (32)$$

### 2.3. Verformungsraten

Unter Punkt 2.1 wurde die Verformung in Abhängigkeit von Anfangs- und Endzustand angegeben. Da das Verhalten vieler Werkstoffe vom Verformungspfad abhängt, werden nun die Zusammenhänge in Ratenform beschrieben.

Die Geschwindigkeit eines Materialteilchens ergibt sich zu

$$\vec{v} = \frac{\partial \vec{x}}{\partial t} \quad (33)$$

Werden nun wieder zwei benachbarte Materialteilchen betrachtet, so gilt für die Geschwindigkeitsdifferenz in der laufenden Konfiguration

$$d\vec{v} = \frac{\partial \vec{v}}{\partial \vec{x}} d\vec{x} = \mathbf{L} d\vec{x} \quad (34)$$

wobei mit  $\mathbf{L}$  der Geschwindigkeitsgradient der laufenden Konfiguration eingeführt wurde. Unter Verwendung des Deformationsgradienten kann  $d\vec{x}$  ersetzt werden.

$$d\vec{v} = \mathbf{L} d\vec{x} = \mathbf{L} \mathbf{F} d\vec{X} \quad (35)$$

Der Geschwindigkeitsgradient wird in den symmetrischen und schiefssymmetrischen Anteil zerlegt. Der symmetrische Anteil entspricht der Dehnrate.

$$\begin{aligned} \dot{\epsilon} = \mathbf{D} &= \frac{1}{2}(\mathbf{L} + \mathbf{L}^T) = \\ &= \frac{1}{2} \left( \left[ \frac{\partial \vec{v}}{\partial \vec{x}} \right] + \left[ \frac{\partial \vec{v}}{\partial \vec{x}} \right]^T \right) \end{aligned} \quad (36)$$

Diese Gleichung erinnert in ihrer strukturellen Form an die Definition der Dehnung für kleine Verformungen. Man müsste nur  $\vec{v}$  durch  $\vec{u}$  ersetzen und man erhielte die Dehnung anstatt der Dehnrate.

Der schiefssymmetrische Anteil ist der Rotationsgeschwindigkeitstensor.

$$\begin{aligned} \mathbf{W} &= \frac{1}{2}(\mathbf{L} - \mathbf{L}^T) = \\ &= \frac{1}{2} \left( \left[ \frac{\partial \vec{v}}{\partial \vec{x}} \right] - \left[ \frac{\partial \vec{v}}{\partial \vec{x}} \right]^T \right) \end{aligned} \quad (37)$$

## 2.4. Multiplikative Zerlegung des Deformationsgradienten

Bei dem hier betrachteten Werkstoff Metall kann die elastische Dehnung als sehr klein gegenüber der plastischen Dehnung angesehen werden. Wenn nun in einem Materialpunkt sowohl elastische als auch plastische Dehnung vorliegt, so kann der rein plastische Zustand durch elastische Entlastung erreicht werden.

$$\begin{aligned} \mathbf{F}^{pl} &= (\mathbf{F}^{el})^{-1} \mathbf{F} \\ \mathbf{F} &= \mathbf{F}^{el} \mathbf{F}^{pl} \end{aligned} \quad (38)$$

Der Deformationsgradient setzt sich multiplikativ aus einem elastischen und einem plastischen Anteil zusammen.

## 2.5. Gleichgewicht

Lösung jeder mechanischen Problemstellung ist das Auffinden des mechanischen Gleichgewichts. Im Folgenden wird das exakte Gleichgewicht beschrieben. Das Kräftegleichgewicht besagt, dass die Summe der Kräfte an der Oberfläche und die Summe der Volumenkräfte zusammen null ergeben müssen.

$$\int_S \vec{t} dS + \int_V \vec{f} dV = \vec{0} \quad (39)$$

Dabei ist  $\vec{t}$  die Oberflächenkraft bezogen auf die Einheitsfläche und  $\vec{f}$  eine Volumenkraft bezogen auf das Einheitsvolumen. Die Oberflächenkraft an jedem Punkt der Oberfläche kann aus der Cauchy Spannung und dem auswärts gewandten Normalenvektor berechnet werden. Achtung: Im Folgenden sind Vektoren (z.B.:  $\vec{n}$ ) als Zeilenvektoren definiert.

$$\vec{t} = \vec{n}\boldsymbol{\sigma} \quad (40)$$

Nach Einsetzen in Gleichung 39 erhält man:

$$\int_S \vec{n}\boldsymbol{\sigma}dS + \int_V \vec{f}dV = \vec{0} \quad (41)$$

Mit Hilfe des Gauß Theorems kann das Oberflächenintegral in ein Volumenintegral umgewandelt werden.

$$\int_S \vec{n}\boldsymbol{\sigma}dS = \int_V \left( \frac{\partial}{\partial \vec{x}} \right) \cdot \boldsymbol{\sigma}dV \quad (42)$$

Da das Gleichgewicht für beliebige Teilvolumina gelten muss, können die Integrale weggelassen werden und man erhält das punktweise definierte Gleichgewicht für Translation.

$$\left( \frac{\partial}{\partial \vec{x}} \right) \cdot \boldsymbol{\sigma} + \vec{f} = \vec{0} \quad (43)$$

Stellt man dieselben Überlegungen für das Drehmomentengleichgewicht an, so erhält man, dass der Cauchy Spannungstensor symmetrisch sein muss (Dualität der Schubspannungen).

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \quad (44)$$

Bis zu diesem Punkt wurden keine Approximationen gemacht. Für die verschiebungsbasierte FE-Formulierung wird eine schwächere Formulierung des Gleichgewichts eingeführt. Dabei wird die Differentialgleichung mit einer Testfunktion multipliziert und das Ergebnis integriert. Diese Testfunktion entspricht einem virtuellen Geschwindigkeitsfeld. Dadurch ergibt sich anstatt der drei Gleichungen eine einzelne skalare Gleichung, welche besagt, dass die Summe der virtuellen Leistungen null sein muss.

$$\int_V \left[ \left( \frac{\partial}{\partial \vec{x}} \right) \cdot \boldsymbol{\sigma} + \vec{f} \right] \cdot \delta\vec{v}dV = 0 \quad (45)$$

Nach Anwendung der Kettenregel und des Gauß Theorems erhält man den Ausdruck für die virtuelle Leistung. Die Leistungen aller Oberflächenkräfte und Volumenkräfte sind gleich der Änderungsrate der Verzerrungsenergie.

$$\int_S \vec{t} \cdot \delta\vec{v}dS + \int_V \vec{f} \cdot \delta\vec{v}dV = \int_V \boldsymbol{\sigma} : \left( \frac{\partial\delta\vec{v}}{\partial\vec{x}} \right) dV \quad (46)$$

Im letzten Ausdruck finden wir den virtuellen Geschwindigkeitsgradienten.

$$\frac{\partial\delta\vec{v}}{\partial\vec{x}} = \delta\mathbf{L} = \delta\mathbf{D} + \delta\mathbf{W} \quad (47)$$

Da  $\mathbf{D}$  symmetrisch ist und  $\mathbf{W}$  schiefsymmetrisch ist, erhält man schließlich die Gleichung für die virtuelle Leistung:

$$\int_V \boldsymbol{\sigma} : \delta\mathbf{D}dV = \int_S \vec{t} \cdot \delta\vec{v}dS + \int_V \vec{f} \cdot \delta\vec{v}dV \quad (48)$$



## 2.6. Spannung

In der Gleichung für die virtuelle Leistung wird eine Spannung mit einer Dehnungsrate multipliziert. Da es, wie in Abschnitt 2.2 dargestellt, verschiedene Verzerrungsmaße gibt, muss es auch dazu konjugierte Spannungsgrößen geben. Unter konjugiert wird hier arbeitskonjugiert verstanden. Als Ausgangspunkt wird die interne Arbeit bezogen auf das entlastete Ausgangsvolumen ausgedrückt.

$$\int_V \boldsymbol{\sigma} : \mathbf{D} dV = \int_{V^0} J \boldsymbol{\sigma} : \mathbf{D} dV^0 = \int_{V^0} \boldsymbol{\tau} : \mathbf{D} dV^0 \quad (49)$$

Wobei  $J = dV/dV^0$  die Determinante der Jacobimatrix der elastischen Verformung zwischen Ausgangszustand und elastisch verformtem Zustand ist.

Verwendet man Green Verzerrungen

$$\begin{aligned} \varepsilon^G &= \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}) \\ \dot{\varepsilon}^G &= \frac{1}{2}(\dot{\mathbf{F}}^T \mathbf{F} + \mathbf{F}^T \dot{\mathbf{F}}) \end{aligned} \quad (50)$$

und setzt man für  $\dot{\mathbf{F}} = \mathbf{L}\mathbf{F}$  ein, so erhält man

$$\dot{\varepsilon}^G = \frac{1}{2} \mathbf{F}^T (\mathbf{L} + \mathbf{L}^T) \mathbf{F} = \mathbf{F}^T \mathbf{D} \mathbf{F} \quad (51)$$

damit wird

$$\mathbf{D} = \mathbf{F}^{-T} \dot{\varepsilon}^G \mathbf{F}^{-1} \quad (52)$$

Dies kann für die Berechnung der Arbeitsrate eingesetzt werden

$$\begin{aligned} dW^0 &= J \boldsymbol{\sigma} : \mathbf{D} = J \boldsymbol{\sigma} : (\mathbf{F}^{-T} \dot{\varepsilon}^G \mathbf{F}^{-1}) = \\ &= J (\mathbf{F}^{-1} \boldsymbol{\sigma} \mathbf{F}^{-T}) : \dot{\varepsilon}^G \end{aligned} \quad (53)$$

Somit ergibt sich die zu den Green Verzerrungen konjugierte Spannung (= der zweite Piola-Kirchhoff Spannungstensor):

$$\mathbf{S} = J \mathbf{F}^{-1} \boldsymbol{\sigma} \mathbf{F}^{-T} \quad (54)$$

## 2.7. FE-Ansatz

In der Gleichung für die virtuellen Arbeitsraten in der Form der konjugierten Spannungs- und Verzerrungsgrößen befinden sich Größen, die durch Interpolation von Knotengrößen bestimmt werden müssen. Die Spannung  $\boldsymbol{\tau}^c$  und die virtuelle Verzerrung  $\delta\varepsilon$  stellen dabei beliebige arbeitskonjugierte Größen dar.

$$\int_{V^0} \boldsymbol{\tau}^c : \delta\varepsilon dV = \int_S \vec{t} \cdot \delta\vec{v} dS + \int_V \vec{f} \cdot \delta\vec{v} dV \quad (55)$$

$$\delta\vec{v} = \mathbf{N}_N \delta\vec{v}^N \quad (56)$$

$$\delta\varepsilon = \boldsymbol{\beta}_N \delta\vec{v}^N \quad (57)$$

$\mathbf{N}_N$  ist die Interpolationsmatrix. Damit werden die  $N$  Knotenpunktverschiebungen sowie das virtuelle Geschwindigkeitsfeld  $\delta\vec{v}$  innerhalb der Elemente interpoliert. Die Verzerrungs-Verschiebungsmatrix  $\boldsymbol{\beta}_N$  stellt den Zusammenhang zwischen dem virtuellen Geschwindigkeitsfeld und den Verzerrungen her. Durch diese Interpolationen können die virtuellen Geschwindigkeiten vor die Integrale gezogen werden.

$$\delta\vec{v}^N \left[ \int_{V^0} \boldsymbol{\beta}_N : \boldsymbol{\tau}^c dV^0 - \int_S \mathbf{N}_N^T \cdot \vec{t} dS - \int_V \mathbf{N}_N^T \cdot \vec{f} dV \right] = 0 \quad (58)$$

Da die virtuellen Größen unabhängig und beliebig wählbar sind, erhalten wir die Basisgleichung für die verschiebungsbasierte FE-Methode:

$$\int_{V^0} \boldsymbol{\beta}_N : \boldsymbol{\tau}^c dV^0 = \int_S \mathbf{N}_N^T \cdot \vec{t} dS + \int_V \mathbf{N}_N^T \cdot \vec{f} dV \quad (59)$$

## 2.8. Plastizitätsmodelle

Plastizitätsmodelle können ratenabhängig oder ratenunabhängig sein. Die hier betrachtete Kaltumformung von Metallen kann als ratenunabhängig angesehen werden, da der Verlauf der Fließkurve kaum von der Dehnrage abhängt. Eine Grundannahme bei elastoplastischen Modellen ist, dass die gesamte Verformung in einen elastischen und einen plastischen Anteil aufgespalten werden kann (siehe Abschnitt 2.4). Dabei wird davon ausgegangen, dass die elastische Verformung klein ist. Die plastische Verformung kann groß sein. Diese Einschränkung trifft auf Metalle zu. Die elastische Antwort kann durch Ableiten des elastischen Verzerrungsenergiepotentials bestimmt werden.

$$\boldsymbol{\sigma} = \frac{\partial U}{\partial \boldsymbol{\varepsilon}^{el}} \quad (60)$$

Bei linear-elastischem Materialverhalten ist das Verzerrungsenergiepotential besonders einfach und die elastische Verzerrung ergibt sich zu

$$d\boldsymbol{\varepsilon}^{el} = \mathbf{C}_{el}^{-1} \cdot d\boldsymbol{\sigma} \quad (61)$$

Solange die Fließbedingung nicht erfüllt ist, reagiert das Material rein elastisch.

$$f(\boldsymbol{\sigma}, \theta, H_\alpha) < 0 \quad (62)$$

Wobei  $\theta$  die Temperatur und  $H_\alpha$  Verfestigungsparameter sind. Bei Ratenunabhängigkeit kann die Fließfunktion maximal 0 werden. Genau dann tritt Fließen ein. Die hier verwendete Fließbedingung ist die von Mises-Fließbedingung. Diese besagt, dass der Fließeintritt durch Oktaederschubspannungen bestimmt wird. Im Hauptnormalspannungsraum hat die von Mises-Fließfunktion folgende Form:

$$f = \frac{1}{2\sigma_F^2} [(\sigma_{22} - \sigma_{33})^2 + (\sigma_{33} - \sigma_{11})^2 + (\sigma_{11} - \sigma_{22})^2] - 1 \quad (63)$$

Diese Funktion beschreibt eine Zylindermantelfläche im Hauptnormalspannungsraum mit der Zylinderachse in  $[111]^T$ . Solange der Spannungszustand im Inneren dieses Zylinders liegt ist die Antwort des Materials rein elastisch. Wenn der Spannungszustand auf der Zylindermantelfläche liegt, tritt plastisches Fließen ein. Setzt man für die Fließbedingung  $f = 0$ ,  $\sigma_{22} = \sigma_{33} = 0$  ein, so erhält man

$$\sigma_{11} = \sigma_F \quad (64)$$

D.h.  $\sigma_F$  entspricht der Fließspannung beim einachsigen Zugversuch. Somit wurde eine Verbindung zwischen der Fließkurve beim Zugversuch und dem allgemeinen dreidimensionalen Problem hergestellt.

Der plastische Anteil der Verformung lässt sich mit der Fließregel beschreiben.

$$d\boldsymbol{\varepsilon}^{pl} = d\lambda \frac{\partial f}{\partial \boldsymbol{\sigma}} \quad (65)$$

$\lambda$  ist der plastische Konsistenzparameter. Dieser bedeutet bei ratenabhängigen Modellen den Zeitfortschritt bzw. ist er ein Maß für den Fortschritt der plastischen Verformung bei ratenunabhängigen Modellen. Die Richtung der plastischen Dehnung ist normal zur Fläche der Fließfunktion (Normalenregel). Die gesamte Verformung ergibt sich zu

$$d\boldsymbol{\varepsilon} = \mathbf{C}_{el}^{-1} d\boldsymbol{\sigma} + d\lambda \frac{\partial f}{\partial \boldsymbol{\sigma}} \quad (66)$$

Als Verfestigungsmodell wurde die isotrope Verfestigung eingesetzt. Das bedeutet, dass bei plastischem Fließen der Durchmesser der Zylindermantelfläche anwächst, währenddessen bleibt die Lage der Zylinderachse unverändert.

## 2.9. Numerische Lösung

Die Gleichung 59 kann bei den hier betrachteten elasto-plastischen Modellen nur noch iterativ gelöst werden. Der Grund hierfür ist, dass die Materialantwort von der Lösung abhängt (nichtlineares Verhalten). Bei der Lösung wird dermaßen vorgegangen, dass zuerst mit Hilfe einer sogenannten Prädiktorspannung die Gesamtverzerrung für den nächsten Zeitschritt berechnet wird. Diese Prädiktorspannung kann z.B. die elastische Antwort des Materials sein. D.h. im ersten Schritt verhält sich das Material rein elastisch. Nun wird überprüft, ob dieser neue Spannungszustand innerhalb der Fließfläche liegt. Ist dies der Fall, so war die Annahme des elastischen Verhaltens korrekt und der nächste Zeitschritt kann berechnet werden. Liegt der Punkt außerhalb der Fließfläche ( $f > 0$ ) so ist dieser Zustand in der ratenunabhängigen Plastizität nicht möglich. Nun muss der plastische Konsistenzparameter  $\lambda$  so variiert werden, dass der Spannungszustand auf der Fließfläche ( $f=0$ ) zu liegen kommt. Es ergibt sich eine sogenannte Korrektorspannung. Dabei muss noch berücksichtigt werden, dass sich die Fließfläche selbst durch die isotrope Verfestigung verschiebt. Dieses Verfahren wird als Radial-Return bezeichnet, da sich bei diesem Vorgang der Spannungspunkt im Hauptnormalspannungsraum radial von außen an die Fließfläche annähert.

### 3. Modellierung

Es wird ein Modell vorgestellt, welches voraussagen soll, ob an einem harten Teilchen mit einer bestimmten Größe Keimbildung für Rekristallisation stattfinden kann. Im Zentrum steht dabei die Bildung von Subkörnern mit einer hohen Missorientierung in der Nähe dieser Teilchen. Da Aluminium ein wichtiger Werkstoff mit harten Teilchen ist, wird es als Matrixwerkstoff herangezogen. Die Teilchen haben eine wesentlich höhere Festigkeit als der Matrixwerkstoff und verhalten sich aus diesem Grund bei Verformung rein elastisch. Somit kommt es in der Nähe des Teilchens zu einer inhomogenen plastischen Verformung der Matrix. Es wird vorausgesetzt, dass es dabei nicht zum Bruch zwischen Matrix und Teilchen kommt. Durch die plastische Verformung der Matrix kommt es zur Erzeugung von Versetzungen und somit zur Verfestigung. Durch die große plastische Dehnung kommt es zur dynamischen Erholung und somit zur Bildung von Zellen und Subkörnern. Die vorhandenen Teilchen führen nicht nur zu einer inhomogenen plastischen Dehnung der Matrix, sondern auch zu einer inhomogenen Rotation des Materials und somit zu Rotationsgradienten. Rotationsgradienten führen zur Bildung von zusätzlichen geometrisch notwendigen Versetzungen. Diese wiederum sind verantwortlich für Subkörner mit hoher Missorientierung.

Für die Berechnung der plastischen Verformung wird eine FE-Berechnung durchgeführt. Dabei kommt es zu folgenden Vereinfachungen:

- Aus Ermangelung an einer kommerziell verfügbaren FE-Software für Kristallplastizität wird isotropes elastisches und plastisches Werkstoffverhalten angenommen, obwohl das betrachtete Volumen innerhalb eines einzelnen Kornes liegt. Da dieses Korn jedoch in einem polykristallinen Werkstoff eingeschlossen ist, müssten für die plastische Verformung mehr Gleitsysteme aktiviert werden, als dies für einen Einkristall der Fall wäre. Trotzdem ist die Annahme eines isotropen Werkstoffverhaltens nicht erfüllt und stellt somit eine große Einschränkung dar.
- Obwohl auf der betrachteten Längenskala ( $\mu\text{m}$ ) Größeneffekte beobachtet werden, wird in der eigentlichen FE-Rechnung nicht auf dieses Phänomen eingegangen.
- Es werden makroskopische Fließkurven verwendet, wobei versucht wird, die Parameter für die Fließkurve an das Verhalten im Mikrobereich anzupassen.
- Trotz der getroffenen Vereinfachungen wird davon ausgegangen, dass sich mit dieser FE-Rechnung die Rotationen der Matrix annähernd realistisch berechnen lassen. Dabei stellt die FE-Lösung einen hypothetischen Zustand dar, bei dem geometrisch notwendige Versetzungen während bzw. nach der Kaltverformung an jenem Ort verweilen, an dem sie aufgrund der Rotationsgradienten erzeugt werden müssen.

Die FE-Berechnung kann gewisse physikalische Vorgänge nicht abbilden. So stellt sich bei der Kaltverformung von Aluminium ein Gefüge mit einer Substruktur ein. Versetzungen sind nicht homogen verteilt, sondern ordnen sich bevorzugt in Zellwänden an. Es ergibt sich somit eine hohe Versetzungsdichte in den Zellwänden und eine geringere Versetzungsdichte im Inneren der Zellen. Weiters wird in Versuchen beobachtet (Abbildung

20), dass die Materialrotationen in der Nähe der Teilchen von der Größe der Teilchen selbst abhängen.

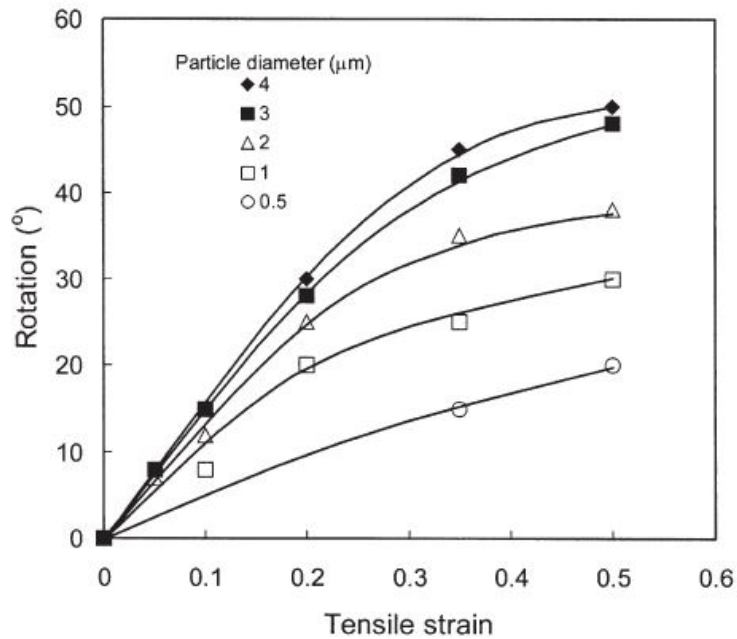


Abbildung 20: Mittlere maximale Missorientierung an Teilchen [3]

Da es für das verwendete FE-Model keine Rolle spielt auf welcher Längenskala es sich befindet, kann dieses Verhalten nicht abgebildet werden.

Um dieses Verhalten in der Berechnung berücksichtigen zu können, werden im Postprocessing folgende Schritte durchgeführt:

- Es werden Flächen im Nahbereich des Teilchens definiert, welche die Größe eines Subkorns haben.
- Die Größe dieser Subkörner soll unabhängig von der Größe des Teilchens sein.
- Die Orientierung des Subkornes ist homogen und wird durch Mittelung der Orientierung der entsprechenden Fläche im hypothetischen Zustand berechnet.

Dieses nachträgliche Berechnen der Orientierung entspricht physikalisch dem Vorgang der Polygonisation. Durch Anlagern der Versetzungen in Zellwänden kommt es zum Abbau von Orientierungsgradienten innerhalb der Zellen und zu starken Gradienten in den Zellwänden (siehe Abbildung 13).

### 3.1. Subkornrotation

Es stellt sich nun die Frage, wie eine solche Mittelung der Orientierung von Subkörnern aussehen kann. Dazu werden einfache Fälle von Orientierungsgradienten betrachtet. Für

das leichtere Verständnis sei der untersuchte Bereich in diskrete Teilbereiche aufgeteilt. Tatsächlich sollte der Orientierungsverlauf kontinuierlich sein.

Im ersten Beispiel wird ein räumlich eindimensionales Problem betrachtet, in dem ein quasi kontinuierlicher Orientierungsgradient vorhanden ist (Abbildung 21). Dabei stellt  $\Theta_0$  die Orientierung des unrotierten Referenzzustandes dar.  $\Theta_i$  stellt die Rotation der jeweiligen Teilbereichs relativ zu  $\Theta_0$  dar.

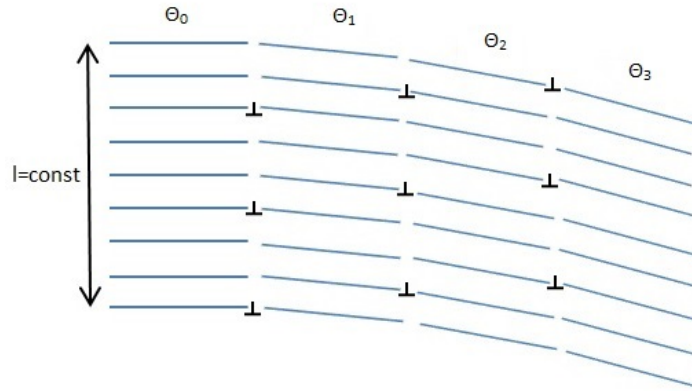


Abbildung 21: Orientierungsgradient 1D

Mit der Gleichung 3 aus Kapitel 1.1.3 kann die Missorientierung an einer Subkorngrenze berechnet werden. Da in unserem Beispiel diskrete Teilbereiche eingeführt wurden kann diese Formel auch in einem Orientierungsgradient angewendet werden.

$$\Delta\Theta_{i-j} = 5^\circ = \frac{\pi}{36} = \frac{b}{D} = \frac{b n_{i-j}}{l} \quad (67)$$

$\Delta\Theta_{i-j}$  sei dabei die Missorientierung zwischen zwei Teilbereichen, die in unserem Fall mit  $5^\circ$  gewählt wird.  $l$  ist die Länge der Grenze und  $n_{i-j}$  die Anzahl der Versetzungen in der Grenze. Die Anzahl der Versetzungen bleibt dabei unbekannt. Auf diese Weise ergibt sich folgende Rotation der Teilbereiche:

- $\Theta_1 = 5^\circ$
- $\Theta_2 = 10^\circ$
- $\Theta_3 = 15^\circ$

In einem Erholungsprozess sollen sich nun die Versetzungen tatsächlich in Subkorngrenzen anordnen. Dabei können zwei Fälle unterschieden werden. Im ersten Fall sollen sich alle Versetzungen an der Grenzfläche zum unrotierten Referenzzustand anordnen (Abbildung 22).

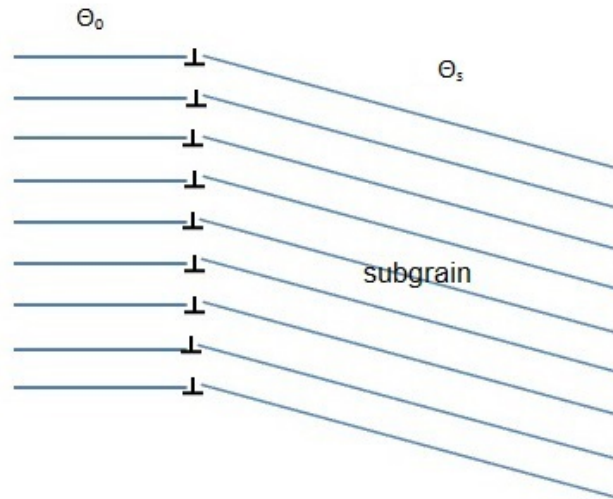


Abbildung 22: Maximale Missorientierung Subkorn 1D

Durch diesen Vorgang bildet sich eine Subkorngrenze mit der maximal vorstellbaren Missorientierung. Da der unrotierte Bereich mit der Orientierung  $\Theta_0$  als Bezugspunkt für die Rotation  $\Theta_s$  innerhalb des Subkornes herangezogen wird und die Missorientierung  $\Delta\Theta_{0-s}$  der gebildeten Subkorngrenze ebenfalls zwischen diesen beiden Bereichen berechnet wird, sind Missorientierung der Subkorngrenze und Rotation des Subkornes identisch. Dies wäre nicht der Fall, wenn die Missorientierung zwischen zwei Bereiche berechnet würden, die selbst bezüglich eines dritten Referenzbereiches rotiert wären.

$$\Theta_s = \Delta\Theta_{0-s} = \frac{b \ 3 \ n_{i-j}}{l} = \frac{\pi}{12} = 15^\circ \quad (68)$$

Diese Rotation entspricht der maximalen Rotation vor der Erholung. Da es eher unwahrscheinlich ist, dass sich alle Versetzungen auf einer Seite anlagern, wird ein zweiter Fall betrachtet. Nun soll sich die Hälfte der Versetzungen links und die zweite Hälfte rechts anordnen.

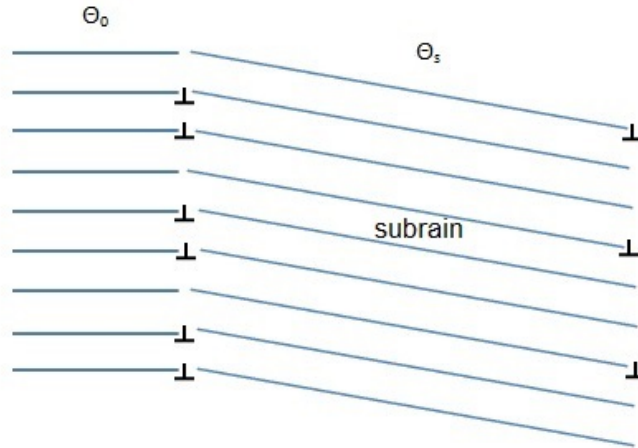


Abbildung 23: Wahrscheinliche Missorientierung Subkorn 1D

Auf diese Weise wird folgende Subkornrotation erreicht

$$\Theta_s = \Delta\Theta_{0-s} = \frac{b \sum n_{i-j}}{l} = \frac{\pi}{18} = 10^\circ \quad (69)$$

Eine andere Möglichkeit, die mittlere Orientierung zu berechnen, ist das gewichtete arithmetische Mittel. Bei gleicher Größe der Teilbereiche ergibt sich die mittlere Orientierung zu

$$\Theta_s = \Delta\Theta_{0-s} = \frac{5 + 10 + 15}{3} = 10^\circ \quad (70)$$

Bei dieser gewählten Anordnung der Versetzungen ergibt dies zufällig genau die Rotation die auch für den wahrscheinlichen Fall der Anordnung der Versetzungen erhalten wurde.

Im zweiten Beispiel soll ein schematisches Beispiel eines Orientierungsgradienten im Nahbereich eines Teilchens betrachtet werden. In diesem Fall sei ein Orientierungsgradient in radialer Richtung vorhanden. Wieder werden Teilbereiche eingeführt. Da die Längen der Grenzflächen im Gegensatz zum ersten Beispiel nicht konstant sind, soll dies als räumlich zweidimensionales Beispiel bezeichnet werden.



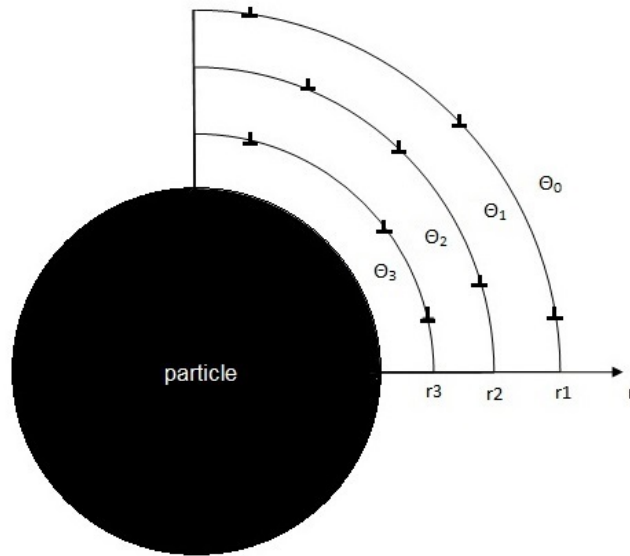


Abbildung 24: Orientierungsgradient 2D

Die Missorientierung an den Grenzflächen sei wiederum  $5^\circ$ .  
 $\Delta\Theta_{0-1} = \Delta\Theta_{1-2} = \Delta\Theta_{2-3} = 5^\circ$  und somit ergibt sich für die einzelnen Teilbereiche:

- $\Theta_1 = 5^\circ$
- $\Theta_2 = 10^\circ$
- $\Theta_3 = 15^\circ$

Die unterschiedlichen Längen  $l_j$  der Grenzflächen zwischen den Teilbereichen, lassen sich aus dem jeweiligen Radius  $r_j$  berechnen

$$l_j = r_j \frac{\pi}{2} \quad (71)$$

und führen unter Verwendung von Gleichung 67 dazu, dass bei einer Missorientierung von  $5^\circ$  ( $=\pi/36$ ) die Anzahl der Versetzungen in den Grenzflächen unterschiedlich ist.

$$n_{i-j} = \frac{r_j \pi/2 \pi/36}{b} \quad (72)$$

Es wird nun wieder ein Erholungsvorgang betrachtet, bei dem sich wie im ersten Fall alle Versetzungen wieder an der Grenze zum unrotierten Material anlagern (Abbildung 25).

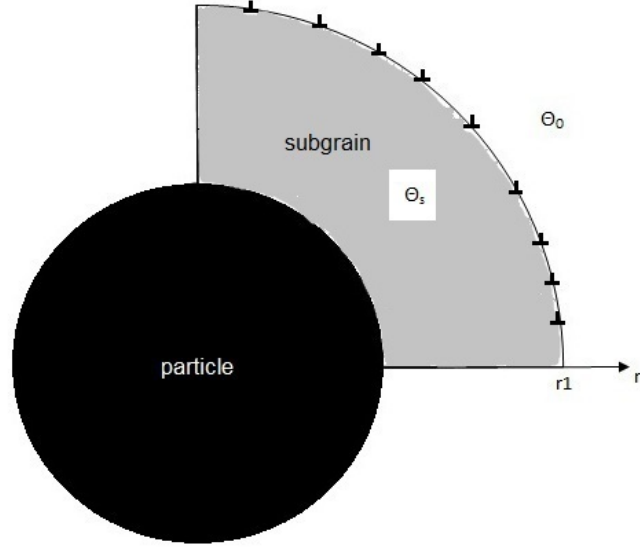


Abbildung 25: Maximale Missorientierung 2D

Bei der Berechnung für die Subkornrotation müssen Verhältnisse für die Radien der Teilbereiche eingeführt werden.

$$\frac{r_1}{r_1} = 1 \quad \frac{r_2}{r_1} = \frac{4}{5}$$

$$\frac{r_3}{r_1} = \frac{3}{5} \quad \frac{r_p}{r_1} = \frac{2}{5}$$

$$\Theta_s = \Delta\Theta_{0-s} = \frac{b}{l} n_{0-s} = \frac{b}{r_1 \pi/2}$$

$$\left[ \frac{r_1 \pi/2 \pi/36}{b} + \frac{r_2 \pi/2 \pi/36}{b} + \frac{r_3 \pi/2 \pi/36}{b} \right] = \quad (73)$$

$$\frac{1}{r_1} [r_1 + r_2 + r_3] \frac{\pi}{36} = \frac{\pi}{15} = 12^\circ$$

Es wird hier ersichtlich, dass im zweidimensionalen Fall die maximal denkbare Rotation des Subkornes geringer als die maximale Rotation vor der Erholung ist. Dies ist auch im Einklang mit experimentellen Beobachtungen (Abbildung 17b) wobei die maximale Rotation in der Verformungszone in Bezug auf das unrotierte Material mit zunehmender Erholung abnimmt. Die Missorientierung an der Subkorngrenze selbst nimmt jedoch zu. Auch diese Bedingung ist hier erfüllt da  $\Delta\Theta_{0-1} = 5^\circ$  vor der Erholung kleiner als  $\Delta\Theta_{0-s} = 12^\circ$  nach der Erholung ist.

Wie auch für den eindimensionalen Fall, soll nun der Fall betrachtet werden, in dem ein Teil der Versetzungen an die Grenzfläche zum unrotierten Material und ein Teil zur Grenzfläche mit dem Teilchen wandert (Abbildung 26).

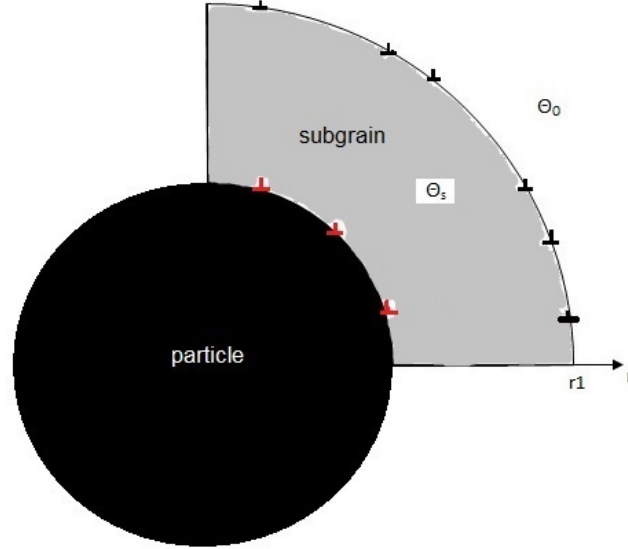


Abbildung 26: Wahrscheinliche Missorientierung 2D

Für diesen Fall ergibt sich die Subkornrotation zu

$$\begin{aligned} \Theta_s = \Delta\Theta_{0-s} &= \frac{b}{l} n_{0-s} = \frac{b}{r_1 \pi/2} \\ &= \left[ \frac{r_1 \pi/2 \pi/36}{b} + \frac{r_2 \pi/2 \pi/36}{b} \right] = \\ &= \frac{1}{r_1} [r_1 + r_2] \frac{\pi}{36} = \frac{\pi}{20} = 9^\circ \end{aligned} \quad (74)$$

Wie zu erwarten war ergibt sich ein Wert, der zwischen den maximalen und minimalen Werten vor der Erholung liegt. Zum Vergleich wird das gewichtete arithmetische Mittel der Orientierung vor der Erholung berechnet.

$$\begin{aligned} \bar{\Theta}_s &= \left[ \frac{(r_1^2 - r_2^2)\pi}{4} \Delta\Theta_1 + \frac{(r_2^2 - r_3^2)\pi}{4} \Delta\Theta_2 + \frac{(r_3^2 - r_p^2)\pi}{4} \Delta\Theta_3 \right] \frac{4}{(r_1^2 - r_p^2)\pi} = \\ &= \left[ \frac{9\pi}{4} \frac{\pi}{36} + \frac{6\pi}{4} \frac{\pi}{18} + \frac{5\pi}{4} \frac{\pi}{12} \right] \frac{4}{21\pi} = \frac{\pi}{21} = 8.57^\circ \end{aligned} \quad (75)$$

Auch im zweidimensionalen Fall kommt das arithmetische Mittel der Orientierungen sehr nahe an das Ergebnis für die Versetzungsbewegung heran.

Die Berechnung einer mittleren Orientierung durch Betrachtung einzelner Versetzungen ist bei tatsächlich kontinuierlichen Orientierungsgradienten nicht mit Formel 67 möglich. Aus diesem Grund wird im Folgenden nur auf das arithmetische Mittel zurückgegriffen. Eine Verfeinerung des Modells könnte durch Verwendung des von Nye eingeführten Versetzungsdichtentensors möglich sein, welcher den Zusammenhang zwischen geometrisch notwendigen Versetzungen und der Gitterkrümmung beschreibt [11]. Aus zeitlichen Gründen wird hier nicht weiter darauf eingegangen und die mittlere Orientierung durch das arithmetische Mittel (Formel 76) berechnet

$$\Theta_m = \frac{\sum \Theta_i^0 A_i}{\sum A_i} \quad (76)$$

wobei  $\Theta_i^0$  und  $A_i$  der lokale Rotationswinkel und die Fläche des betrachteten Elementes aus der FE-Lösung nach der plastischen Verformung sind.

Da der Vorgang der Subkornbildung stetig ist, soll auch der Übergang vom Zustand ohne Subkörner hin zu den voll entwickelten Subkörnern betrachtet werden. Wie ausgeprägt die Subkörner sind, hängt vom Erholungsgrad ab. Hier soll der Erholungsgrad als der Parameter verstanden werden, der beschreibt inwieweit sich Orientierungsgradienten innerhalb des Subkornes abgebaut haben. Er sei also wie folgend definiert:

$$R = \frac{\Theta_i^t - \Theta_i^0}{\Theta_m - \Theta_i^0} \quad (77)$$

Daraus ergibt sich für jedes Element die momentane Orientierung

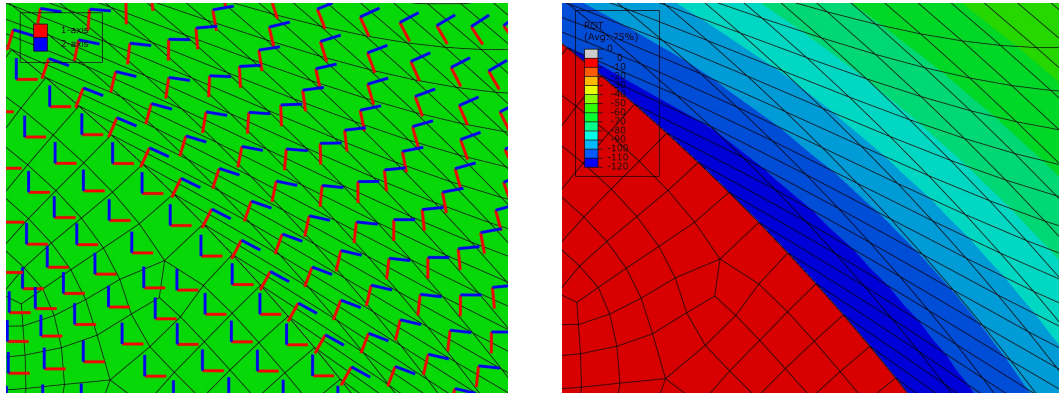
$$\Theta_i^t = \Theta_i^0 + R(\Theta_m - \Theta_i^0) \quad (78)$$

Ist der Erholungsgrad eins, so hat das gesamte Subkorn die gleiche Orientierung.

### 3.2. FE-Modell

Die rechnerische Umsetzung des Modells der Subkornmissorientierung wird mit der FE-Software ABAQUS in Zusammenarbeit mit Python Skript durchgeführt. Die Modellierung des starren Einschlusses in einer weichen Matrix erfolgt dabei auf der graphischen Benutzeroberfläche von ABAQUS, wobei jeder einzelne Schritt als Makro aufgezeichnet wird und somit als Pythoncode zur Verfügung steht. Aus diesen Makroaufzeichnungen wird ein Skript zusammengebaut, in dem wichtige Parameter wie Dehnung und Teilchengröße als Variablen definiert werden, um dann beim Durchlaufen von Schleifen variiert zu werden. Die tatsächliche Simulation wird dann nur mehr durch Aufrufen des Skripts gestartet, wodurch es bei geeignetem Skript möglich ist, automatisierte Parameterstudien durchzuführen. Das Verwenden von Skripten erlaubt nicht nur das Parametrisieren von Standardsimulationen, sondern es wird dadurch auch möglich Postprocessing durchzuführen. In dieser Arbeit wurde mit Hilfe von [12] ein Skript angefertigt, welches die Ausgabedatei der ABAQUS-Simulation liest, mit den Daten Berechnungen durchführt und Daten in Form von neuen ABAQUS Ausgabedateien oder Textdateien ausgibt. Für

den interessierten Leser ist das Skript im Anhang angefügt. Die Hauptaufgabe liegt dabei darin, die Starrkörperrotationen, die von ABAQUS in jedem Integrationspunkt berechnet und als gedrehte Koordinatenachsen ausgegeben werden, herauszulesen und durch trigonometrische Berechnungen in Winkelmaße überzuführen. Diese Ergebnisse können für die Berechnung der Subkornrotation benutzt werden bzw. lässt sich damit ein farbiges Bild mit den lokalen Rotationen erzeugen (Abbildung 27).



(a) Darstellung ABAQUS Simulation in Koordinatenachsen (b) Darstellung nach Postprocessing in Grad Winkelmaß

Abbildung 27: Rotation des Matrixmaterials im Nahbereich eines Einschlusses

### 3.2.1. Geometrischer Aufbau des FE-Modells

Da nur rotationssymmetrische Probleme betrachtet werden, reduziert sich die Rotation auf die Rotation um eine Koordinatenachse. Da der Querschnitt des Teilchens in der  $x,y$ -Ebene modelliert wird, bedeutet das, dass sich das Material um die  $z$ -Achse dreht (vergleiche Abbildung 27).

Als erster Schritt wird der einfachste Fall eines Teilchens, nämlich einer Kugel, untersucht. Das kugelförmige Teilchen befindet sich in einer Matrix, welche durch einen Stauchvorgang in  $y$ -Richtung verformt wird. Die Oberfläche des Teilchens ist fest mit der Matrix verbunden. Aufgrund der rotationssymmetrischen Geometrie wird ein rotationssymmetrisches FE-Modell verwendet (Abbildung 28). Um den numerischen Aufwand weiter zu reduzieren wird die Spiegelsymmetrie des Problems ausgenutzt. Alternativ zum kugelförmigen Einschluss wird ein dreh-elliptisches Teilchen untersucht. Dabei erstreckt sich die Hauptachse in  $x$ -Richtung. Dadurch ist gewährleistet, dass das Teilchen beim Stauchvorgang seine Position behält und nicht selbst rotiert. Die ähnliche Situation trifft im Fall von gewalzten Blechen ein, wo flache Teilchen bevorzugt in Walzrichtung ausgerichtet sind.

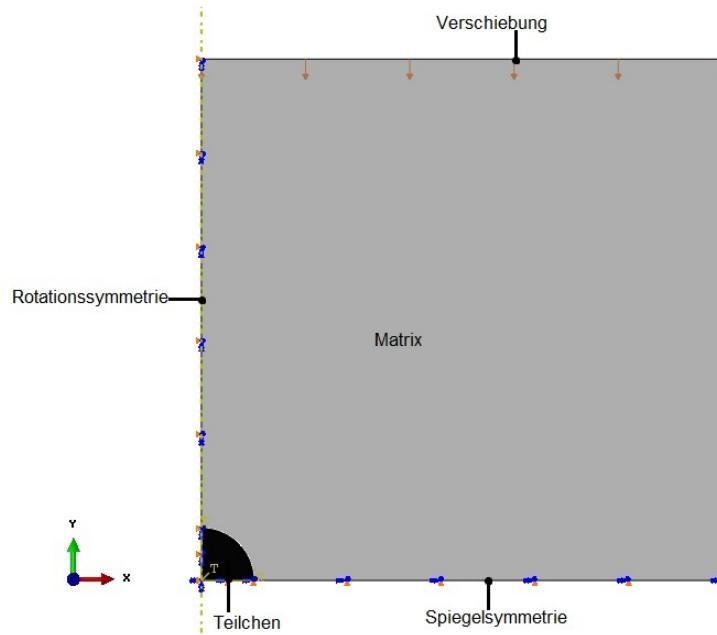
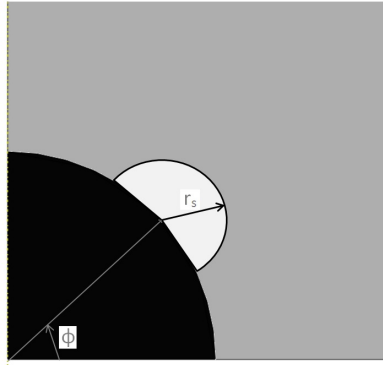


Abbildung 28: Randbedingungen

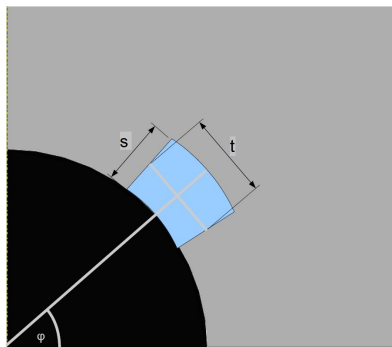
Für die Berechnung der Subkornrotation muss die Form, Lage und Größe der Subkörner vorgegeben werden. Im Postprocessing werden alle Elemente, deren Mittelpunkt innerhalb des Bereiches des Subkorns liegen, für die Berechnung herangezogen bzw. wird deren Wert der Rotation in der neuen Ausgabedatei durch den Wert des Subkornes überschrieben. Dadurch erhalten die Subkörner eine den Elementen entsprechende gezackte Form. Es wurden zwei verschiedene Subkornformen implementiert. Zum einen wurde ein halbkreisförmiges Subkorn modelliert, bei dem der Mittelpunkt auf der Außenfläche des Teilchens zu liegen kommt (Abbildung 29a). Der Vorteil dieser Form ist, dass sie aufgrund der einheitlichen konvexen Krümmung am ehesten der typischen Form eines Keimes entspricht. Der Nachteil ist, dass sich durch eine Anordnung dieser Form nicht die gesamte Fläche rund um den Einschluss abdecken lässt. Aus diesem Grund wurde eine weitere Form eingeführt, die dem äußeren Teil eines Tortenstückes entspricht (Abbildung 29b). Diese Form lässt sich dermaßen anordnen, dass ein zentrales Subkorn von drei anderen Subkörnern umschlossen wird (Abbildung 29c). Die Lage der Subkörner wird jeweils so gewählt, dass die Rotation des zentralen Subkornes maximal und somit die Keimbildung am wahrscheinlichsten wird. Diese Wahl der Position wird als zweckmäßig empfunden, da aufgrund der Rotationssymmetrie der Kugel und aufgrund des isotropen Werkstoffverhaltens und der gegebenen Randbedingung diese Position nicht nur einmal im Modell vorkommt, sondern in Wirklichkeit zwei Kreisringe auf der Ober- und Unterseite der Kugel gleichwertig sind. Geht man davon aus, dass die Lage der Subkörner zufällig ist, so ist die Wahrscheinlichkeit trotzdem hoch, dass irgendein Subkorn sich in etwa auf einem dieser Kreisringe befindet. Für die Abmessungen der Subkörner gelten folgende Zusammenhänge, wobei  $d_{SG}$  als Subkorngröße bezeichnet wird und in der Simulation als

Parameter herangezogen wird:

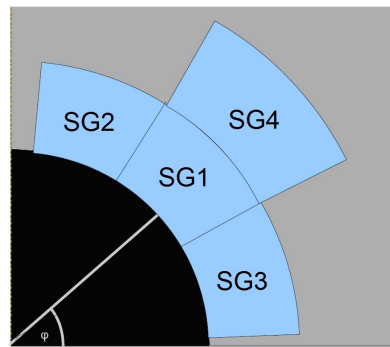
- $r_s = d_{SG}/2$
- $t = d_{SG}$
- $s = d_{SG}/2$



(a) Halbkreisförmiges Subkorn



(b) Tortenstückförmiges Subkorn



(c) Anordnung Subkörner

Abbildung 29: Modellierung Subkörner

### 3.2.2. Vernetzung

Da große plastische Verformungen berechnet werden, kommt der Vernetzung große Bedeutung zu. Es wird das Element CAX4RH verwendet. Dabei handelt es sich um ein 4-Knotenelement mit reduzierter Integration und Hybridformulierung [10, Analysis User's Manual, Kapitel 27.1.6]. Durch die reduzierte Integration werden Lockingeffekte vermieden. Die Hybridformulierung ist besonders dann wichtig, wenn inkompressible Werkstoffe untersucht werden. Da Metalle im plastischen Zustand nahezu inkompressibel sind, würde es bei Verwendung von rein verschiebungsbasierten Elementen zu einem Volumenlocking kommen. Für die Analyse der Rotation ist dieses Element dahingehend praktisch, da es nur über einen Integrationspunkt verfügt an dem die Rotation berechnet wird. Die

Elemente werden bei der Simulation stark verzerrt. Um dem entgegenzuwirken, wird bei der Vernetzung darauf geachtet, dass jene Elemente, die gestaucht werden, vor der Verformung gestreckt sind und umgekehrt. Außerdem werden Elemente, die einer Drehung unterworfen werden, der Drehung entgegen gerichtet. Damit wird erreicht, dass selbst bei großen Dehnungen ein passables Netz erhalten bleibt.

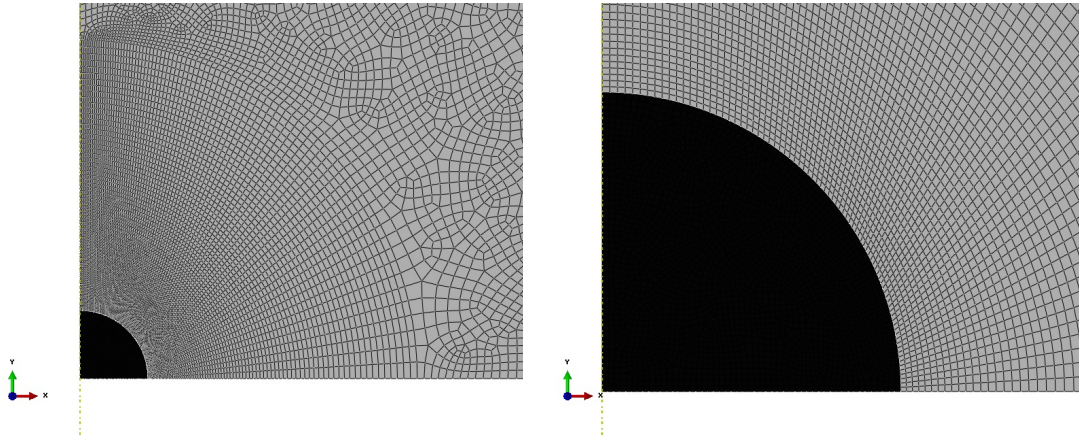


Abbildung 30: Vernetzung

### 3.2.3. Materialkennwerte

Für den harten Einschluss wird ein rein elastisches Materialmodell mit folgenden Eigenschaften verwendet:

- $E_P = 400\text{GPa}$
- $\nu_P = 0,2$

Für das Matrixmaterial wurden die elastischen Eigenschaften von Aluminium verwendet. Für das plastische Verhalten wurde das Johnson-Cook Materialmodell [10, Analysis User's Manual, Kapitel 22.2.7] verwendet, wobei die Temperatur- und Dehnratenabhängigkeit nicht betrachtet wurde.

$$\sigma = a + b\varepsilon^c \quad (79)$$

Es wurden folgende Kennwerte für die Matrix verwendet:

- $E_M = 70\text{GPa}$
- $\nu_M = 0,34$
- $a = 25\text{MPa}$
- $b = 165\text{MPa}$
- $c = 0,75(0,25; 0,5)$



Die verschiedenen Werte für den Exponenten  $c$  wurden verwendet, um das lokale Fließverhalten des Materials in der Nähe des Einschlusses zu beeinflussen. Der Exponent  $c = 0,25$ , der im Abgleich eines makroskopischen Fließversuchs mit der numerischen Simulation bestimmt wurde [6], führt im Mikromodell zu extrem hohen lokalen Dehnungen. Aus diesem Grund wurde der Exponent variiert, wobei mit zunehmendem Exponenten die Spitzen der lokalen Dehnung abnehmen. Der Einfluss des Exponenten wird in der Simulation untersucht. Wenn nichts anderes angemerkt wird, wurde der Exponent  $c = 0,75$  verwendet.

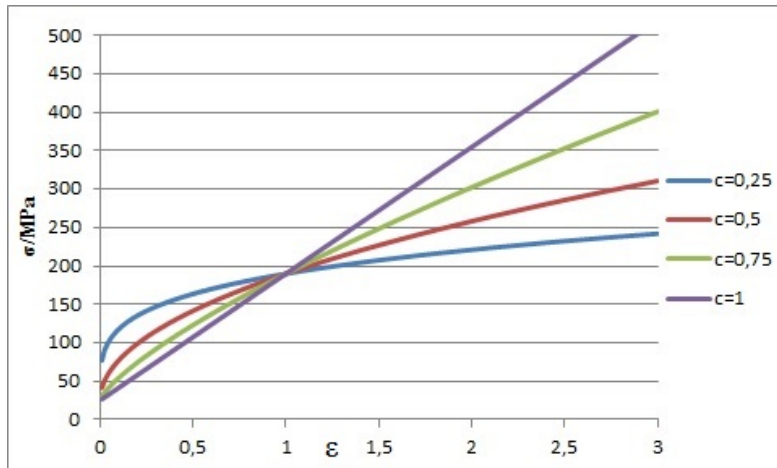


Abbildung 31: Fließkurven Johnson-Cook

# Teil II.

## Ergebnis und Diskussion

In diesem Teil werden zuerst die Ergebnisse der FE-Simulation gezeigt. Darauf aufbauend ist die im Postprocessing bestimmte Rotation von Subkörnern. Aus den Subkornrotationen lässt sich schließlich ein Kriterium für Keimbildung der Rekristallisation ableiten. In der Diskussion werden Vergleiche mit experimentellen Beobachtungen angestellt.

### 4. Ergebnisse

#### 4.1. Dehnungen und Rotationen

Im Folgenden werden die Ergebnisse der FE-Simulation für die plastische Dehnung und die Rotation rund um einen harten Einschluss gezeigt. Es handelt sich hierbei um die unveränderten Ergebnisse der FE-Software, d.h. es wurden hier keine nachträglichen Änderungen im Sinne des in Kapitel 3 eingeführten Modells vorgenommen.

Im ersten Schritt wird an einem kugelförmigen Einschluss der Einfluss des Exponenten  $c$  der Fließkurve gezeigt. Im zweiten Schritt wird das Ergebnis für den kugelförmigen Einschluss mit dem von rotationselliptischen Einschlüssen verglichen. In den betrachteten Beispielen hat die globale Dehnung (die Dehnung der Matrix weit weg vom Einschluss) einen Wert von 1. Dadurch zeigt die Legende sowohl den absoluten Wert der Dehnung als auch den relativen Wert in Bezug auf die globale Dehnung. In Abbildung 32 wird die maximale Rotation in der Verformungszone in der Nähe einer Kugel als Funktion der Dehnung und des Exponenten der Fließkurve gezeigt. Als Vergleich dient die experimentelle Kurve für  $4\mu\text{m}$  Teilchen, wie sie in Abbildung 20 zu finden ist.

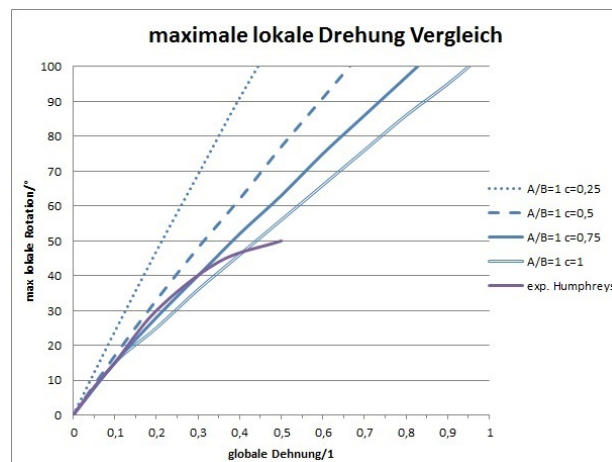


Abbildung 32: Maximale Rotation an einem kugelförmigen Einschluss

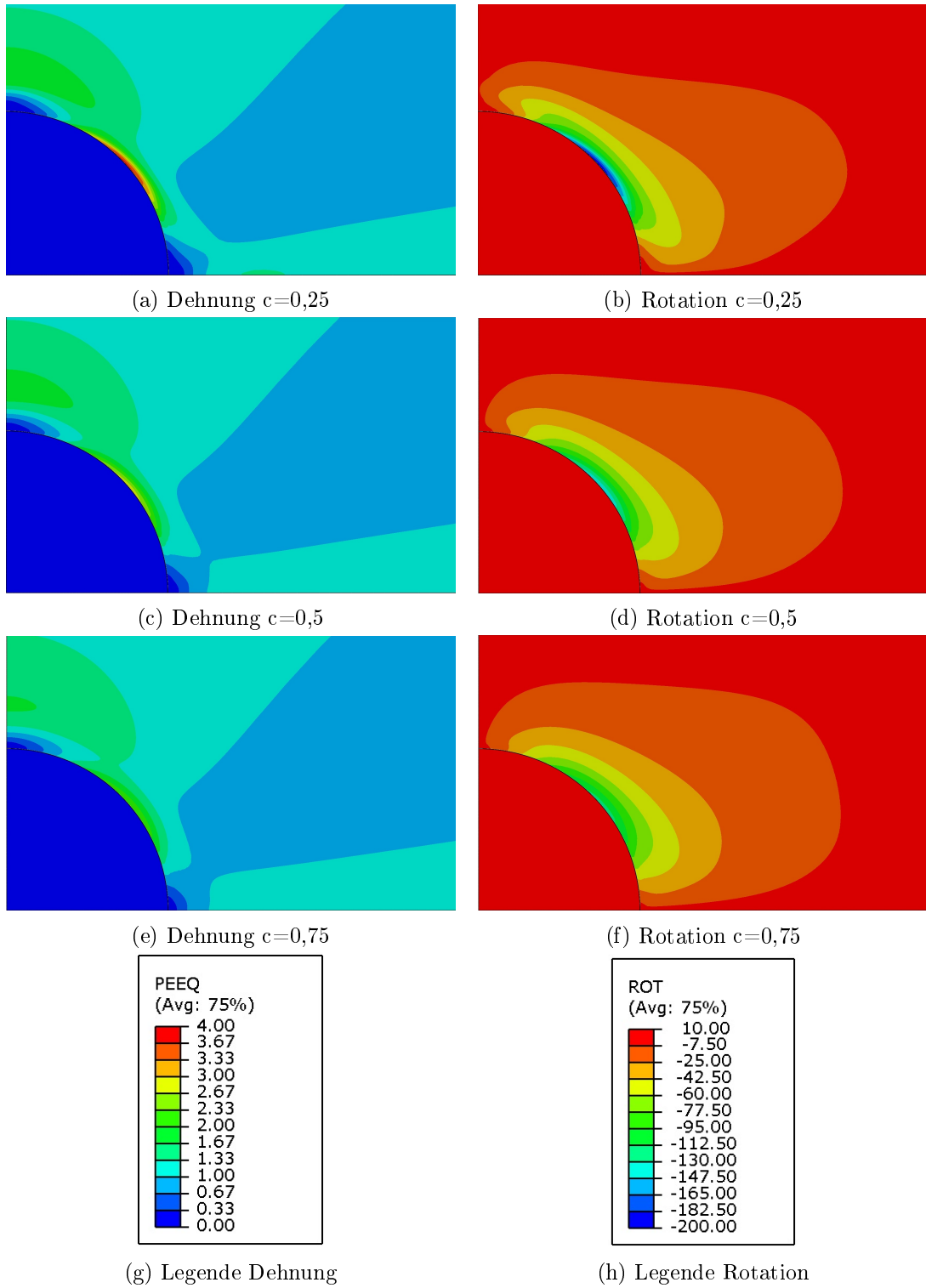


Abbildung 33: Lokale Dehnung und Rotation als Funktion des Verfestigungsexponenten

In Abbildung 33 werden die Dehnungen und Rotationen aus der FE-Lösung für verschiedene Fließkurven verglichen.

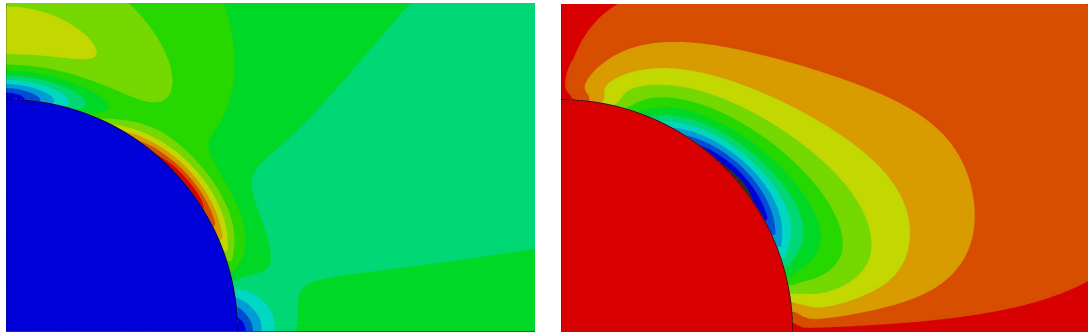
Folgende Erkenntnisse lassen sich aus den Simulationsergebnissen ableiten:

- Sowohl die maximale Dehnung als auch die maximale Rotation finden in der Nähe der Kugel unter  $45^\circ$  zur Waagrechten statt.
- Hohe Dehnung bedeutet jedoch nicht automatisch hohe Rotation, da z.B. oberhalb der Kugel die Dehnung hoch, jedoch die Rotation gering ist.
- Im betrachteten Quadranten ist die positive Drehrichtung im Gegenuhrzeigersinn (vgl. Koordinatensystem in Abbildung 30). Die Materialrotationen im betrachteten Quadranten sind im Uhrzeigersinn und ergeben sich deshalb negativ.
- Der Verfestigungsparameter hat einen Einfluss auf die maximalen lokalen Dehnungen und Rotationen.

Für den Verfestigungsparameter  $c=0,25$  ergibt sich eine lokale Dehnung, die den vierfachen Wert der globalen Dehnung entspricht. Die maximale lokale Rotation beträgt  $200^\circ$ . Durch Erhöhung des Verfestigungsexponenten werden diese lokal hohen Werte verringert. Da diese hohen lokalen Dehnungen kaum mit der Grundannahme, dass es zu keinem Bruch zwischen Teilchen und Matrix kommt, vereinbar sind und zudem das FE-Modell an seine gültigen Grenzen stößt, wird im Folgenden der Verfestigungsparameter  $0,75$  verwendet.

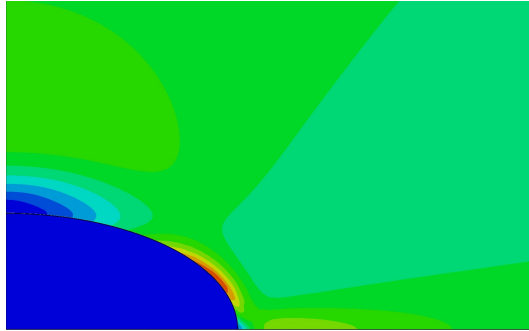
Nun werden Einschlüsse unterschiedlicher Form analysiert. Alle Formen haben gemeinsam, dass sie durch Drehung einer halben Ellipse um  $360^\circ$  erzeugt werden können, wobei  $A/B$  das Verhältnis zwischen Hauptscheitel und Nebenscheitel ist. Im Fall der Kugel sind Hauptscheitel und Nebenscheitel gleich groß.

Es zeigt sich, dass sich der Punkt mit der höchsten Dehnung entlang der Ellipse in den Punkt verschiebt, an dem die Tangente um  $45^\circ$  geneigt ist (Abbildung 34). D.h. die höchste Dehnung findet in der Nähe des Hauptscheitels statt. Gleiches gilt für die Rotation.

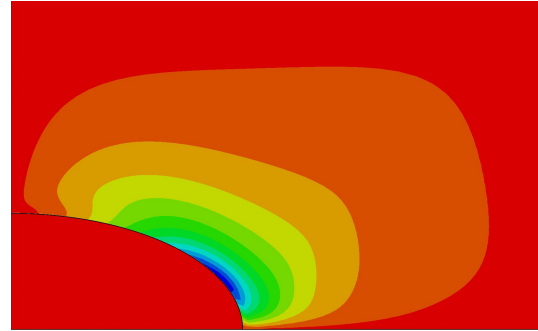


(a) Dehnung  $A/B=1$

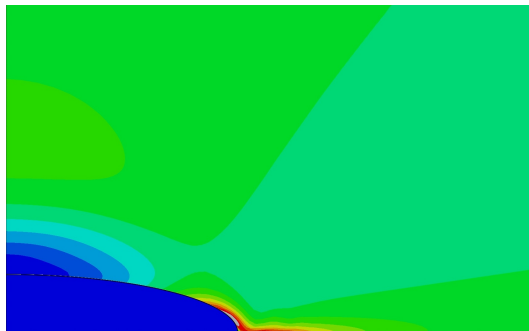
(b) Rotation  $A/B=1$



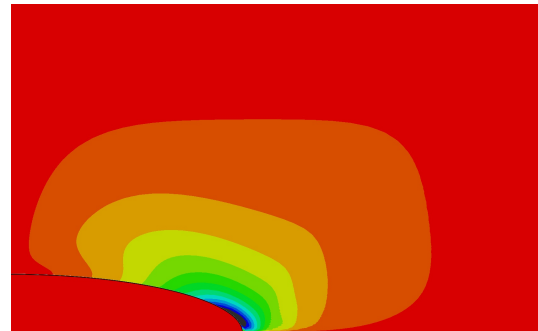
(c) Dehnung  $A/B=2$



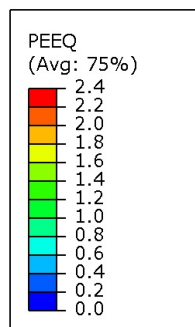
(d) Rotation  $A/B=2$



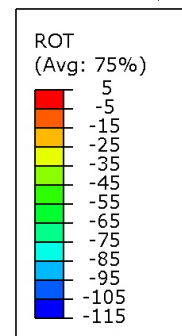
(e) Dehnung  $A/B=4$



(f) Rotation  $A/B=4$

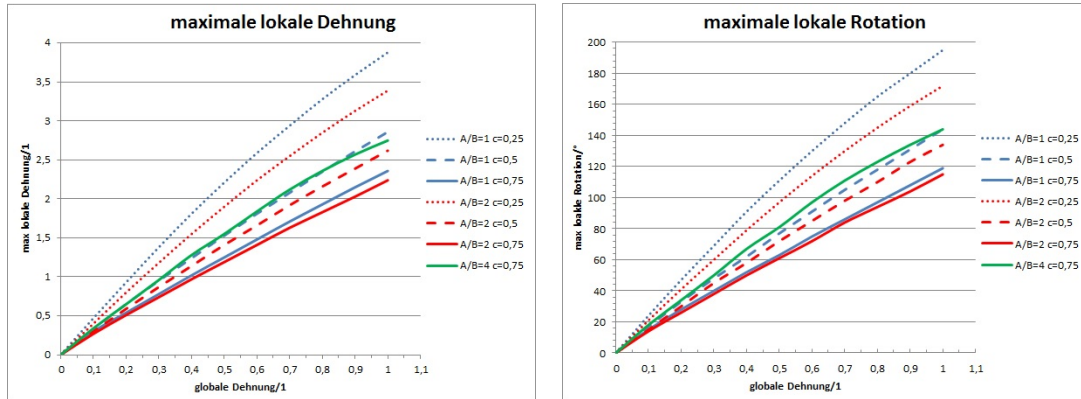


(g) Legende Dehnung



(h) Legende Rotation

Abbildung 34: Lokale Dehnung und Rotation als Funktion der Teilchenform



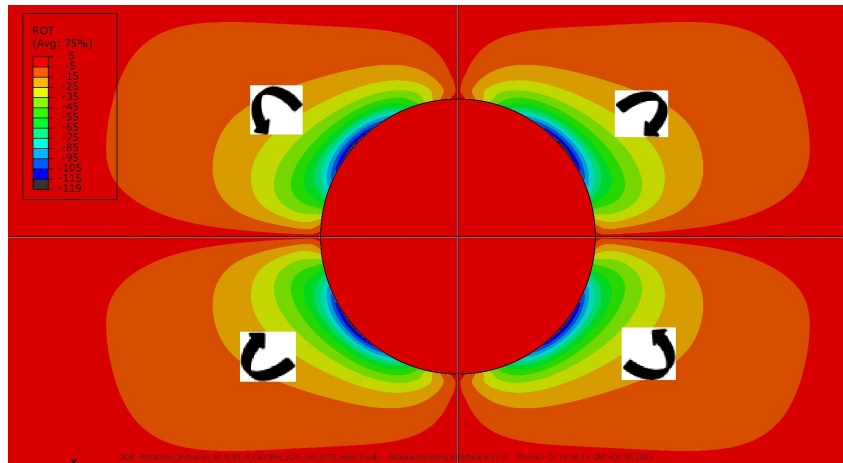
(a) Maximale lokale Dehnung

(b) Maximale lokale Rotation

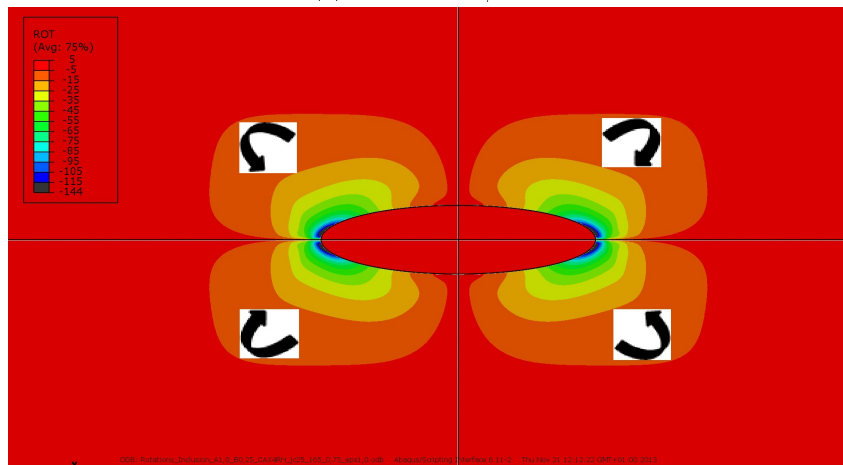
Abbildung 35: Maximale lokale Dehnung und Rotation als Funktion der Dehnung

In Abbildung 35 werden die Ergebnisse für die maximale lokale Dehnung und Rotation für verschiedene Teilchenformen und Fließkurven zusammengefasst. Für das Scheitelverhältnis von 2 ergeben sich geringere maximale Werte. Im Gegensatz nehmen die Spitzenwerte bei einem Scheitelverhältnis von 4 wieder zu. Für diese flachen Teilchen ergeben sich an der Spitze starke lokale Verformungen, wodurch es nur mit dem Parameter  $c=0,75$  möglich war die Simulation durchzuführen.

Abschließend wird in Abbildung 36 durch Spiegelung das gesamte Teilchen dargestellt. Bei dieser Darstellung ist darauf zu achten, dass die in den Quadranten 2 und 4 gezeigten Rotationen mit falschen Vorzeichen dargestellt werden, da sich bei der Spiegelung das Vorzeichen der Rotation fälschlicherweise nicht umdreht. Beträgsmäßig stimmt die Farbschattierung jedoch mit der Legende überein. Die Drehrichtung wird mittels Rotationspfeilen angedeutet.



(a) Rotation  $A/B=1$



(b) Rotation  $A/B=4$

Abbildung 36: Lokale Rotation als Funktion der Teilchenform

## 4.2. Subkornbildung durch Erholung

Es wird der Übergang vom in Kapitel 4.1 gezeigten Zustand zu jenem Zustand gezeigt, in dem klar ausgebildete Subkörner vorhanden sind. Der Erholungsgrad  $R$  aus Gleichung 77 dient dabei als Parameter. Dieser Parameter beschreibt inwieweit sich Rotationsgradienten innerhalb der Subkörner abgebaut haben. Für den Fall, dass sich alle Versetzungen in den Subkorn Grenzen befinden (also keine Gradienten innerhalb des Subkorns) gilt  $R=1$ . Welche Wärmebehandlung (Temperatur, Zeit) notwendig ist um einen bestimmten Wert von  $R$  zu erreichen, hängt vom betrachteten Material ab und wird hier nicht untersucht. Um die Auswirkungen der Subkornbildung zu verdeutlichen, wird dieser Prozess anhand des Modells mit 4 Subkörnern veranschaulicht.

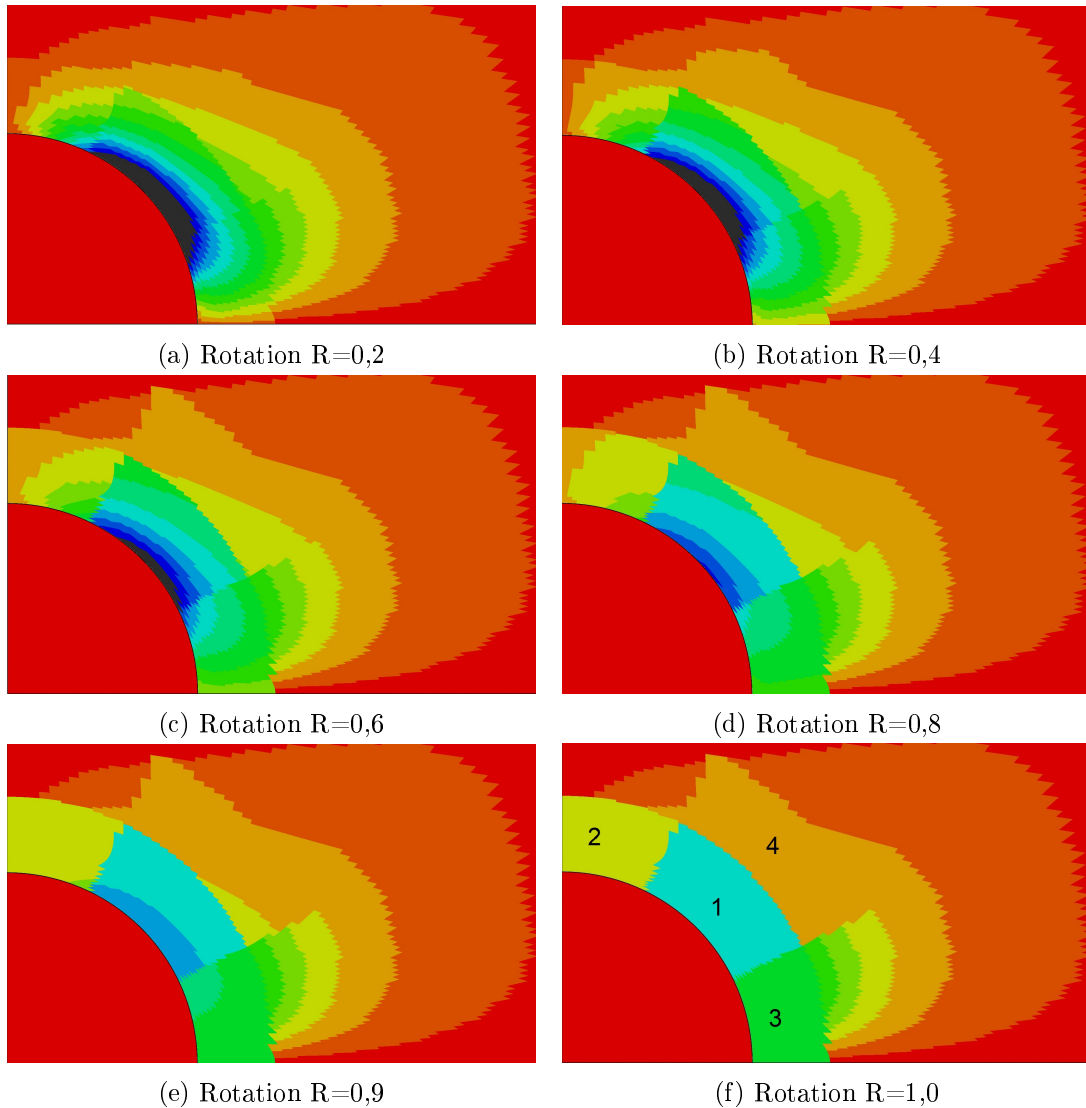


Abbildung 37: Subkornorientierung als Funktion der Erholung

Zur Bestimmung der Missorientierung als Funktion der Erholung wurden an den Subkornengrenzen gegenüberliegende Elementpaare betrachtet. An jeder Grenzfläche des zentralen Subkornes (1) mit seinen Nachbarn wurde an jeweils 5 benachbarten Elementpaaren die Differenz der Rotation berechnet. Aus den 5 Werten wurde eine mittlere Missorientierung an der jeweiligen Grenze berechnet. Dies wurde für drei Erholungszustände durchgeführt (Abbildung 38). Mit zunehmender Erholung steigt die Missorientierung zwischen dem zentralen Subkorn und den umliegenden Subkörnern. Die maximale Missorientierung innerhalb des zentralen Subkorns in Bezug zur ungestörten Matrix ( $\max rot1$ ) nimmt jedoch ab.



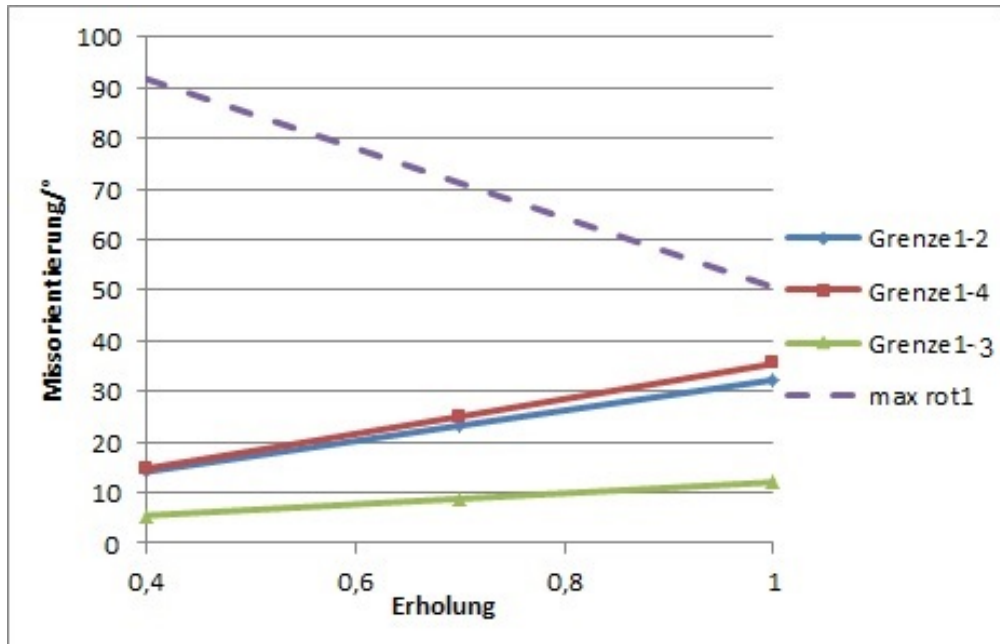


Abbildung 38: Missorientierung als Funktion der Erholung

### 4.3. Rotation nach Subkornbildung

Die Ergebnisse aus Kapitel 4.1 sind unabhängig von der Teilchengröße. Da im betrachteten Längenbereich jedoch eine Längenabhängigkeit vorhanden ist, wird nun das in Kapitel 3 vorgeschlagene Verfahren zur Bestimmung der Missorientierung von Subkörnern systematisch angewandt. Dabei wird der Zustand untersucht, bei dem sich keine Orientierungsgradienten im Subkorn mehr befinden (entspricht  $R=1$  aus Kapitel 4.2). Es wird dabei ein halbkreisförmiges Subkorn verwendet.

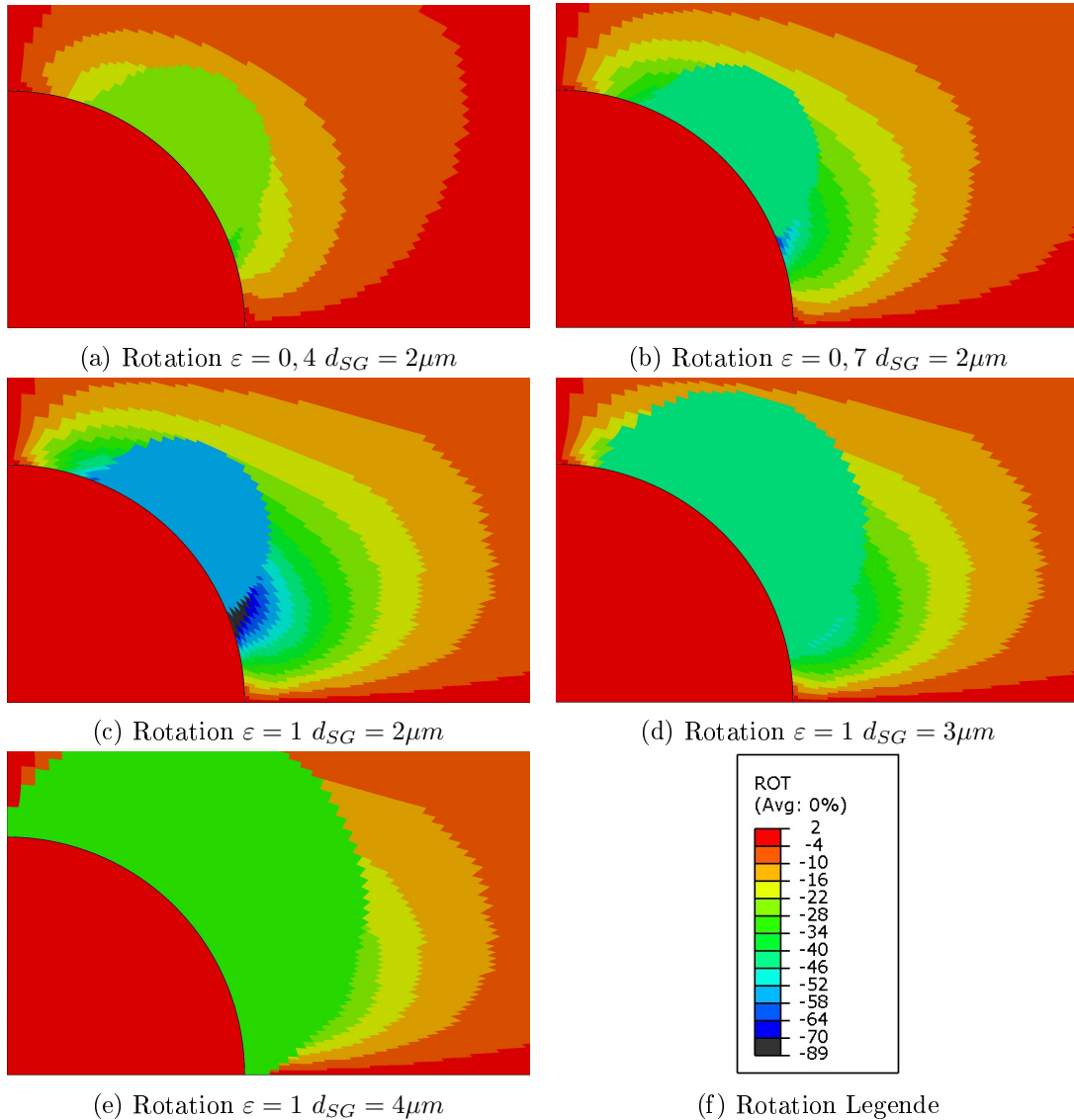


Abbildung 39: Rotation eines Subkornes an einem  $d_p = 5\mu m$  Teilchen als Funktion der globalen Dehnung und der Subkorngröße

Wie auch in Abbildung 37 erkennt man in Abbildung 39 sofort, dass die Rotation der Subkörner geringer als die ursprüngliche maximale Rotation ist. Mit zunehmender globaler Dehnung nimmt die Rotation der Subkörner zu (Abbildung 39a-39c). Mit zunehmendem Subkorndurchmesser sinkt die Rotation der Subkörner (Abbildung 39c-39e) wohingegen sie mit zunehmendem Teilchendurchmesser steigt (Abbildung 40a-40d). Tatsächlich ist die Rotation nicht von den Absolutwerten von Subkorngröße und Teilchendurchmesser, sondern von deren Verhältnis abhängig. Dies wird ersichtlich wenn man die Abbildungen 39c und 40d vergleicht.

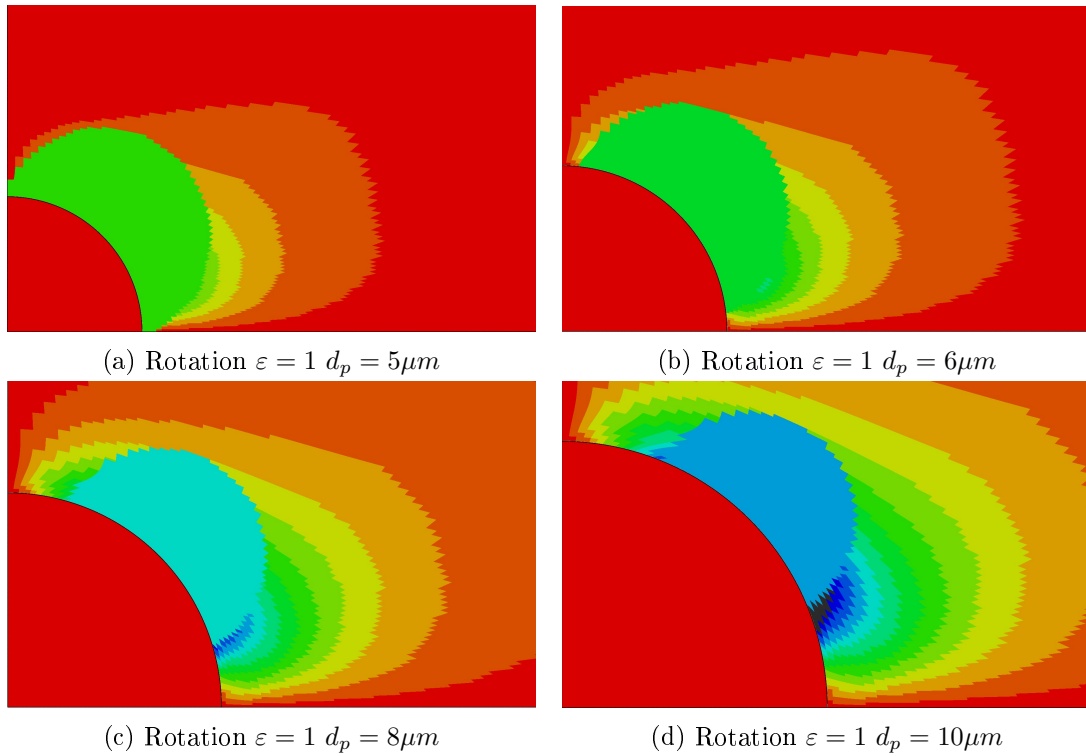


Abbildung 40: Rotation eines Subkornes mit Durchmesser  $4\mu m$  an Teilchen verschiedener Größe

Zur quantitativen Darstellung werden Konturdiagramme verwendet. Damit lassen sich die Subkornrotationen in Abhängigkeit von zwei Parametern darstellen. Abbildung 41a bis 41e zeigen die Rotation eines Subkornes mit bestimmter Größe als Funktion von globaler Dehnung und dem Teilchendurchmesser. Da die Subkornrotation vom Verhältnis von Teilchendurchmesser zu Subkorndurchmesser abhängt kann die gesamte Information dieser fünf Diagramme in ein Diagramm zusammengefasst (Abbildung 41f) werden. Dabei wird  $d_p/d_{SG}$  als relativer Teilchendurchmesser bezeichnet.

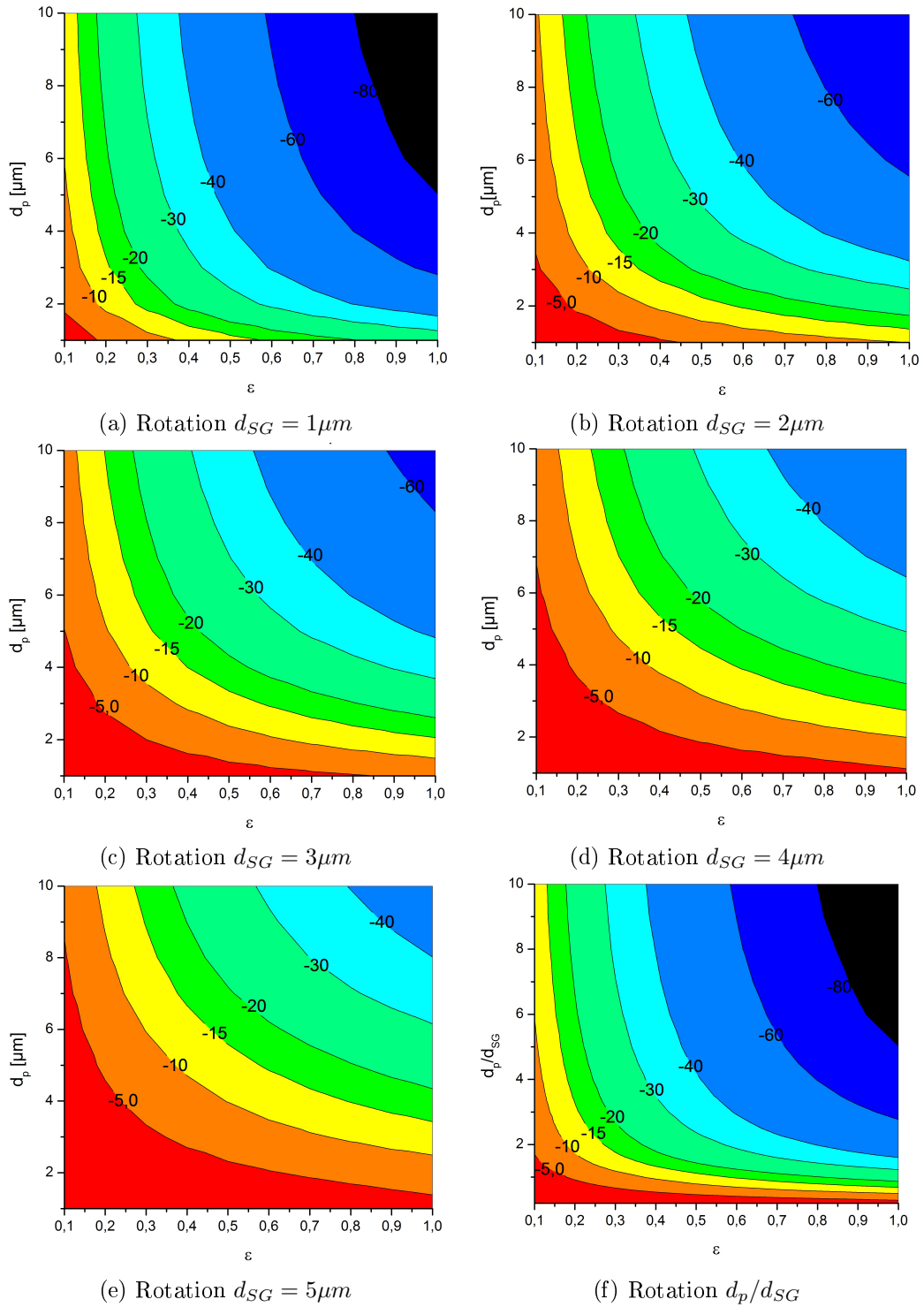


Abbildung 41: Rotation eines Subkornes bestimmter Größe als Funktion von Dehnung und Teilchendurchmesser

In Kapitel 4.1 wurde der Einfluss des Werkstoffmodells bzw. des Verfestigungsexponenten  $c$  auf die maximale Rotation im Nahbereich des Teilchens untersucht. Nun soll überprüft werden inwieweit die Wahl des Werkstoffverhaltens das Ergebnis für die Subkornrotation beeinflusst.

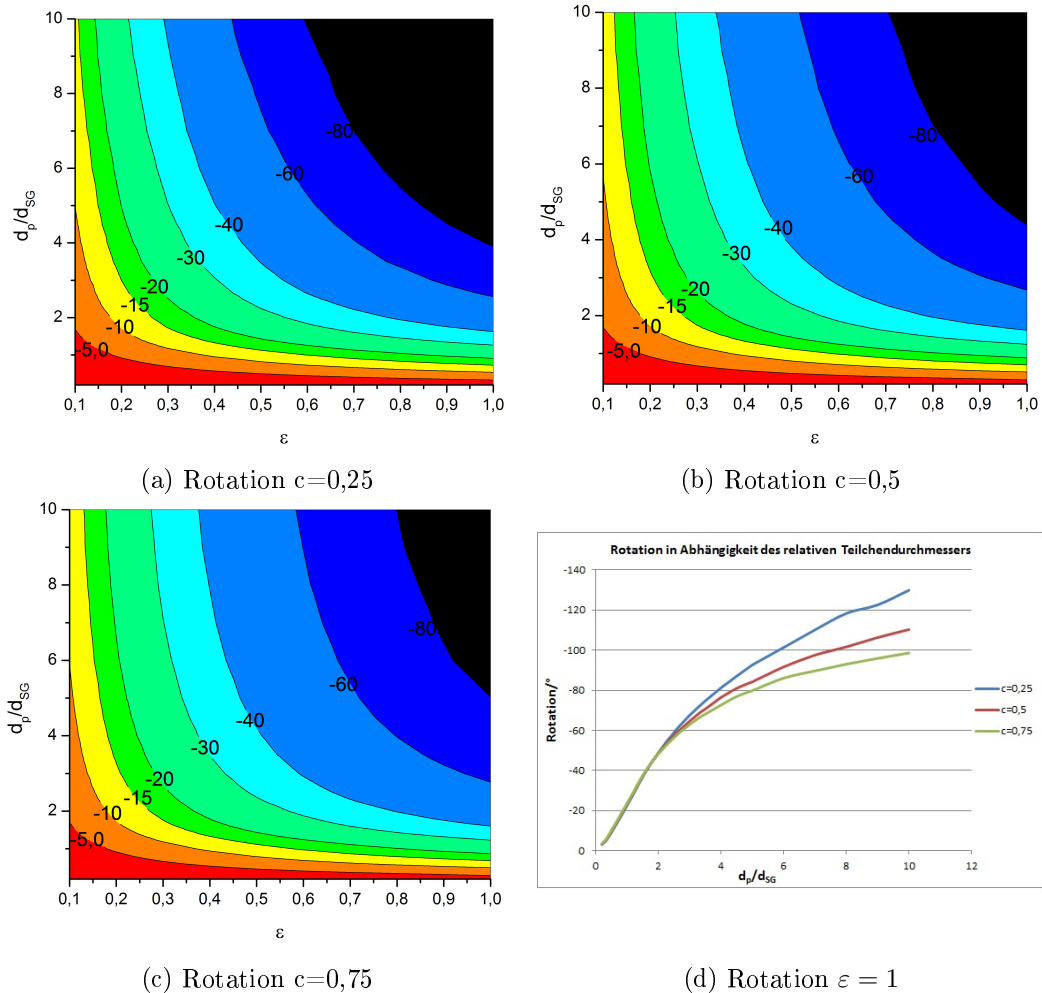


Abbildung 42: Rotation eines Subkornes als Funktion des Verfestigungsexponenten

Abbildung 42d zeigt, dass mit zunehmender relativer Teilchengröße bzw. bei relativ kleinen Subkörnern das Materialmodell sehr wohl einen Einfluss hat. Dies lässt sich dadurch erklären, dass sich relativ kleine Subkörner auf einen sehr kleinen Bereich in der Nähe der Teilchen erstrecken und sich somit die Subkornrotation der maximalen Rotation direkt am Teilchen annähert. Die maximale Rotation wird wie schon in Kapitel 4.1 gezeigt stark vom Materialmodell beeinflusst. Für relativ große Subkörner spielt das Materialmodell keine große Rolle, da in diesem Fall über einen großen Bereich gemittelt wird, wodurch die lokal hohen Werte wenig ins Gewicht fallen.

Ein anderer wesentlicher Parameter ist die Form des Teilchens. Der geometrische Aufbau des Modells erlaubt es rotationselliptische Teilchen zu untersuchen.

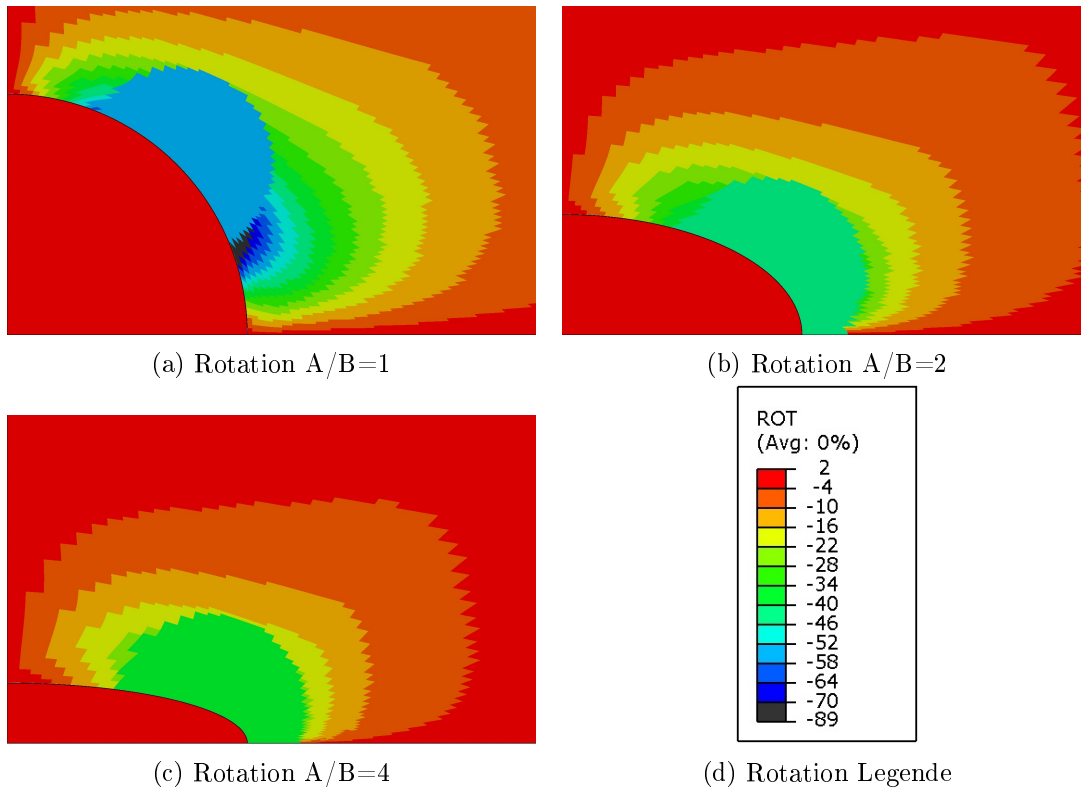


Abbildung 43: Rotation eines  $2\mu\text{m}$  Subkornes an einem rotationselliptischen Teilchen mit Hauptachsenlänge  $A=5\mu\text{m}$

Obwohl wie in Abbildung 35b dargestellt in der FE-Lösung die maximale Rotation für das flache rotationselliptische Teilchen am größten ist, zeigt sich in Abbildung 43 und 44, dass nach Bildung von Subkörnern der gegenteilige Fall eintritt. Grund hierfür ist, dass die Gesamtmenge an rotiertem Material (Rotation mal Fläche) für die Kugel am größten ist. Das kann leicht nachvollzogen werden, indem man die Materialrotationen in Abbildung 34 betrachtet.

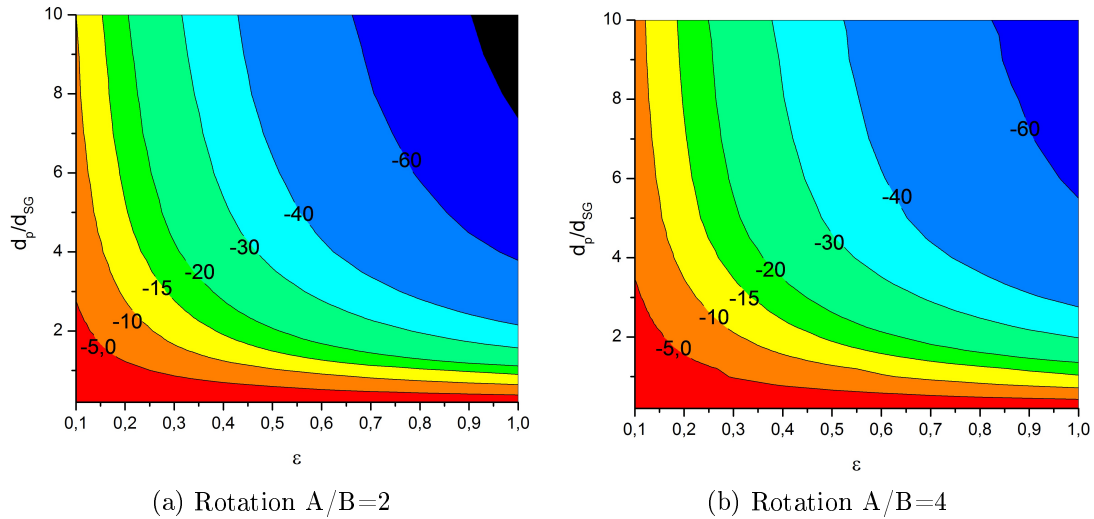


Abbildung 44: Rotation eines Subkornes an rotationselliptischen Teilchen

#### 4.4. Missorientierung benachbarter Subkörner

Um tatsächlich die Keimbildung für Rekristallisation beschreiben zu können, reicht es nicht aus die Rotation eines einzelnen zentralen Subkornes zu berechnen. Diese Rotation entspricht der Missorientierung in Bezug auf das unrotierte Material weit weg vom Teilchen und somit auch weit weg vom Subkorn. Für die Wachstumsfähigkeit eines Subkornes ist vielmehr die Missorientierung in Bezug auf das umliegende Material ausschlaggebend. Aus diesem Grund wird nun wie schon im Kapitel 4.2 auf das Modell mit 4 Subkörnern zurückgegriffen (siehe Abbildung 45).

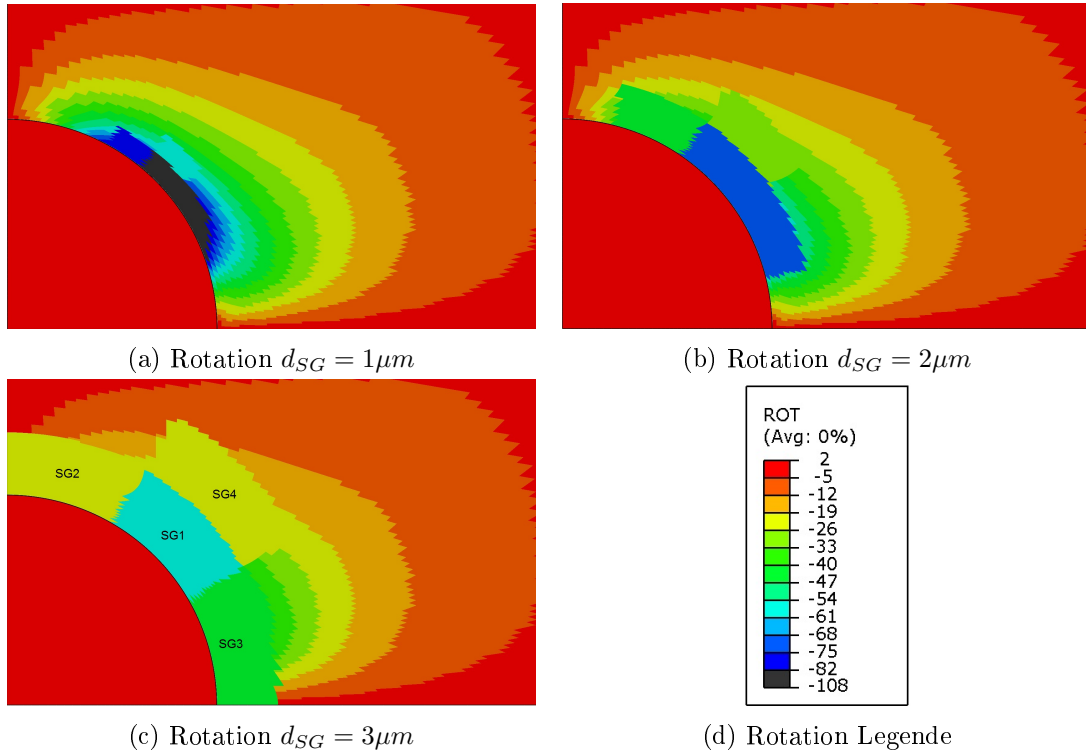


Abbildung 45: 4 Subkörner an  $10\mu m$  Teilchen

Die mittlere Missorientierung des zentralen Subkornes zu den 3 umliegenden Subkörnern wird in Abbildung 46a dargestellt. Dabei fällt auf, dass mit zunehmender Subkorngröße die mittlere Missorientierung zunimmt. Da im verwendeten Modell nur ein Viertel des Querschnittes betrachtet wird, ist bei einem Teilchendurchmesser von  $10\mu m$  die Subkorngröße von 4 Subkörnern auf jeweils etwa  $3\mu m$  beschränkt. Sieht man sich die Missorientierung des zentralen Subkornes zu dem radial nach außen angrenzenden Subkorn 4 an (Abbildung 46b), so erkennt man, dass bei relativ kleinen Subkörnern die Missorientierung mit der Größe zunimmt. Ab einer Größe von etwa  $2\mu m$  nimmt die Missorientierung wieder ab. Dies erklärt sich dadurch, dass bei großen Subkörnern bereits die Missorientierung in Bezug auf das unrotierte Matrixmaterial abnimmt.



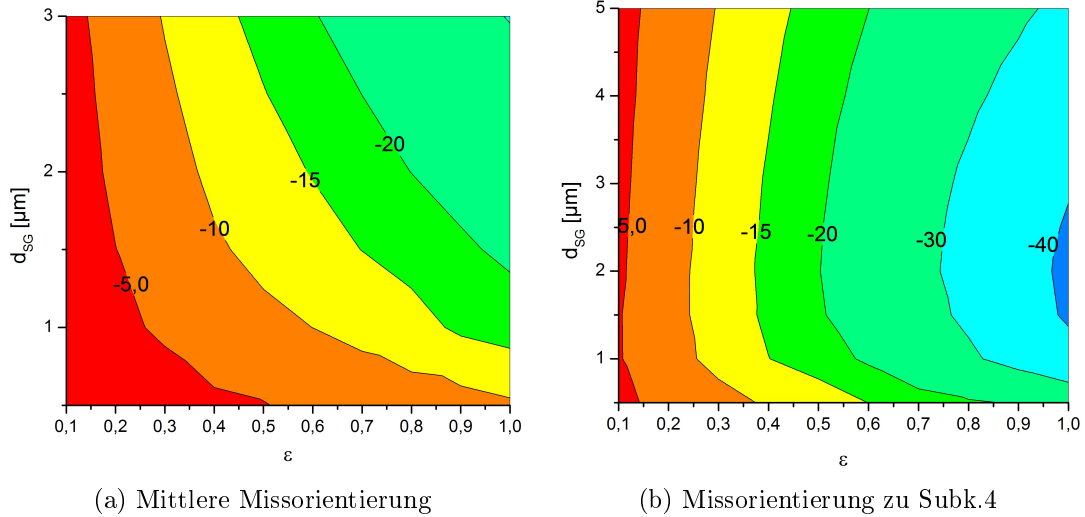
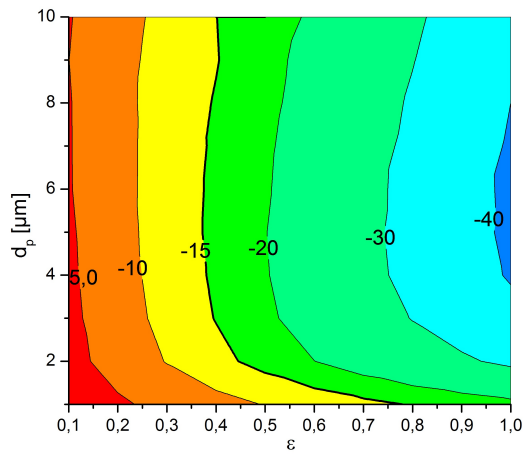
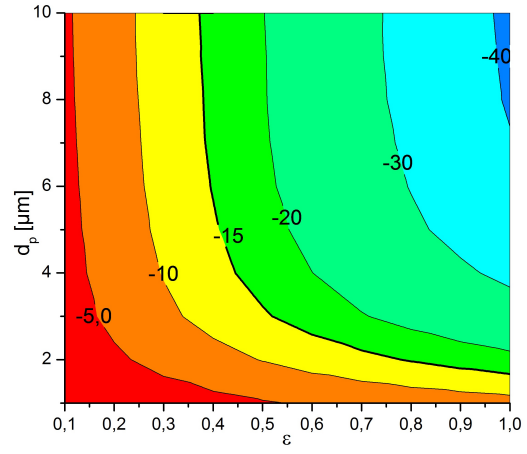


Abbildung 46: Missorientierung des zentralen Subkornes an einem  $10\mu\text{m}$  Teilchen zu seinen Nachbarn

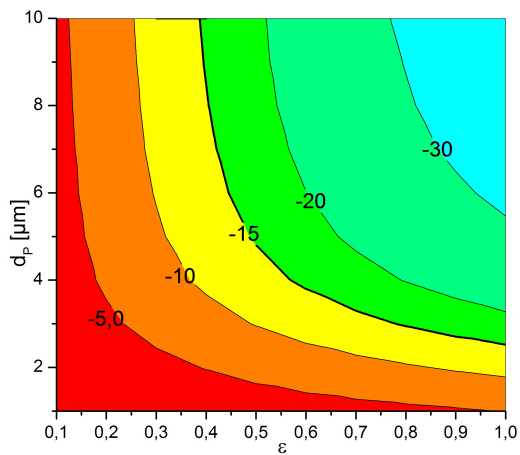
Die Missorientierung zwischen zentralem Subkorn und dem radial nach außen angrenzenden Nachbarn (Subkorn 4) soll als Keimbildungskriterium herangezogen werden. Gründe hierfür werden in der Diskussion erörtert. Da in der Regel die Größe der im Werkstoff vorhandenen Teilchen und die Dehnung bekannt sind, soll die Missorientierung als Funktion dieser beiden Variablen aufgezeichnet werden. Für die Festlegung der Größe der Subkörner gibt es verschiedene denkbare Ansätze. Zum einen kann die mittlere Subkorngröße im betrachteten Gefüge als Wert für das eine spezielle Subkorn, welches als Keimbildungsstelle fungieren soll, angenommen werden. Dabei wird außer Acht gelassen, dass die Subkorngröße einer gewissen Verteilung unterliegt. Eine andere Möglichkeit wäre die notwendige Subkorngröße für die Keimbildung aufgrund der thermodynamischen Stabilität (siehe Gleichung 17 in Kapitel 1.4) abzuschätzen. In der Literatur werden Keime mit einem Durchmesser von  $0,1\mu\text{m}$  als Untergrenze für die Keimbildung zur Rekristallisation genannt [3, Kapitel 7.6]. In dieser Arbeit wurde in Gleichung 18 ein kritischer Keimradius von  $1\mu\text{m}$  abgeschätzt. Tatsächlich ist der kritische Keimradius keine konstante Größe, sondern hängt vom Verformungszustand des Materials ab. Es werden zusätzliche Modelle benötigt, welche eine Aussage über die kritische Keimgröße machen. So wird in [13] die kritische Keimgröße als Funktion der Temperatur und der Dehnrates berechnet. Damit der Einfluss der kritischen Keimgröße auf das in dieser Arbeit betrachtete Modell gezeigt werden kann, werden die Berechnungen für verschiedene Subkorngrößen durchgeführt. In Abbildung 47 wird die Missorientierung des zentralen Subkorns radial nach außen dargestellt, wobei für jede Subkorngröße ein eigenes Diagramm gezeigt wird. Die Isolinie für  $15^\circ$  Missorientierung wird fett dargestellt, da sich hier theoretisch der Übergang zwischen Kleinwinkelkorngrenze und Großwinkelkorngrenze befindet.



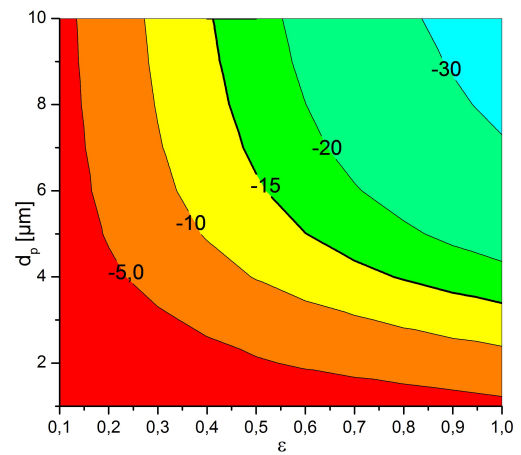
(a) Keimdurchmesser  $1\mu\text{m}$



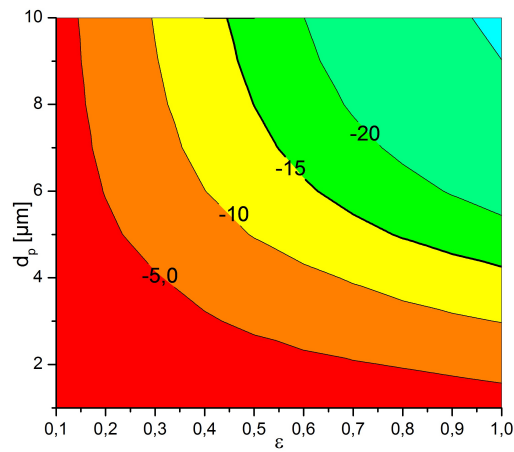
(b) Keimdurchmesser  $2\mu\text{m}$



(c) Keimdurchmesser  $3\mu\text{m}$



(d) Keimdurchmesser  $4\mu\text{m}$



(e) Keimdurchmesser  $5\mu\text{m}$

Abbildung 47: Missorientierung des zentralen Subkornes radial nach außen zu Subkorn 4

## 5. Diskussion

Die FE-Lösung mit isotropem Werkstoffverhalten ergibt Rotationen im Nahbereich harter Teilchen, die sich aufgrund des inkompatiblen Fließverhaltens zwischen harten Teilchen und weicher Matrix ergeben. Das Werkstoffverhalten mit verschiedenen Fließkurven verändert nur die Spitzenwerte der lokalen Rotationen (siehe Abbildung 35). Die Form der Teilchen hat insofern einen Einfluss, als mit der Form der Teilchen die Randbedingung für das Matrixmaterial verändert wird (Abbildung 34). Wie schon in Kapitel 3 angemerkt ist eine Schwäche des verwendeten FE-Modells, dass das anisotrope Verhalten des Werkstoffes nicht abgebildet wird. Eine etwaige durch die Orientierung des Kristalls, der das harte Teilchen umschließt, verursachte Rotation fehlt in der Lösung gänzlich. Zusammenfassend bedeutet dies, dass in der Realität die Rotation sowohl von den Randbedingungen als auch vom Werkstoffverhalten abhängt. In den hier betrachteten Ergebnissen wird hauptsächlich der Einfluss der Randbedingungen beschrieben. Vergleicht man die Ergebnisse mit experimentellen Befunden von [5] so erkennt man, dass zumindest die Lage der Rotationszone richtig vorausgesagt werden kann (Abbildung 48 und 49).

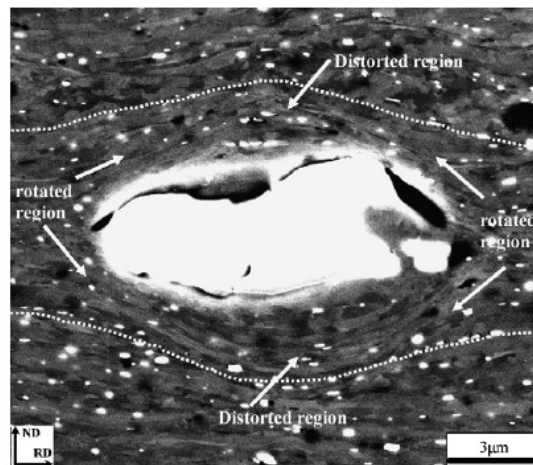


Abbildung 48: Rotationszonen in der Nähe eines Teilchens [5]

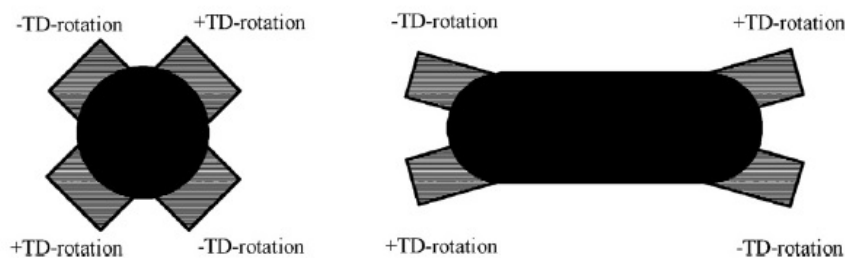


Abbildung 49: Rotationszonen schematisch in der Nähe von Teilchen [5]

In einer parallel zu dieser Arbeit durchgeführten Masterarbeit wurde experimentell bestätigt, dass sich das Material wie im Simulationsergebnis Abbildung 36 dreht. Dazu wurde mit EBSD eine GROD-Karte (Grain Rotation Orientation Deviation) angefertigt, welche den Orientierungsunterschied zwischen der mittleren Orientierung des Kornes und jedem einzelnen Punkt im Korn zeigt. Um zusätzlich die Drehrichtung zu visualisieren wurden an einzelnen Punkten Einheitszellen gezeichnet (Abbildung 50).

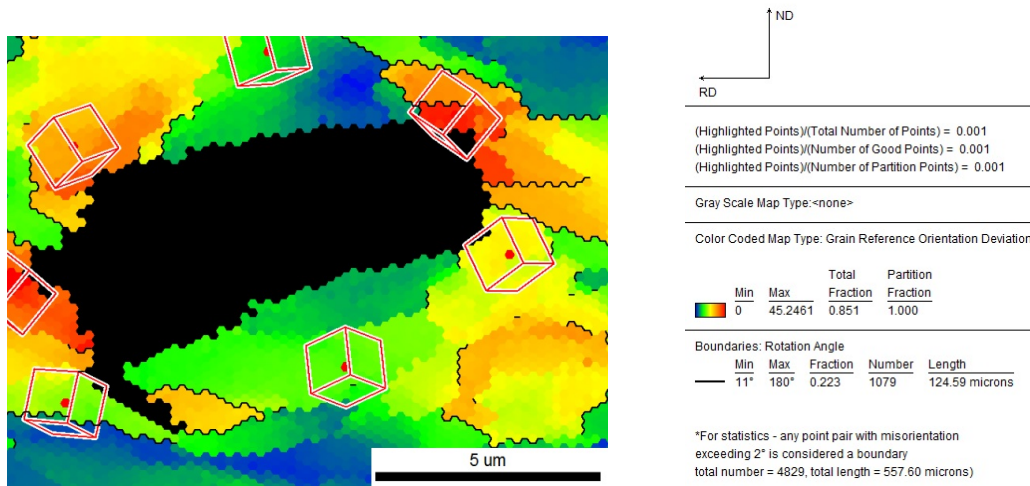


Abbildung 50: GROD-Karte in der Nähe von Teilchen nach Kaltverformung  $\varepsilon \approx 0,5$  [6]

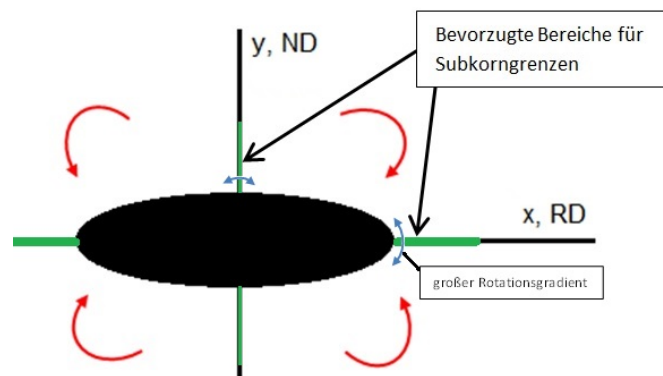


Abbildung 51: Bereiche mit großen Rotationsgradienten in der Simulation

In Abbildung 50 erkennt man, dass sich bei dem in Walzrichtung erstreckten Teilchen (schwarze Fläche) bevorzugt in Walzrichtung Grenzflächen mit hohen Missorientierungen bilden. Vergleicht man dies mit dem Simulationsergebnis (Abbildung 36), so erkennt man, dass für flache Teilchen normal zur x-Richtung (bzw. RD in Abbildung 50), also in Umfangsrichtung am Hauptscheitel des Teilchens, der Rotationsgradient am größten ist, da dort zwei Rotationszonen mit entgegengesetzter Drehrichtung aufeinander tref-

fen. Für das kugelförmige Teilchen ist der Rotationsgradient in Umfangsrichtung sowohl entlang der x als auch der y-Achse hoch, wobei er entlang der x-Achse etwas höher ist. Unter der Voraussetzung, dass sich Subkorn Grenzen großer Missorientierungen bevorzugt in Bereichen großer Rotationsgradienten bilden, sollten sich somit Subkorn Grenzen in vertikaler und horizontaler Richtung ausbilden (Abbildung 51). Ansatzweise kann dies auch beobachtet werden, jedoch wird das idealtypische Verhalten in der Simulation durch die Aktivierung von Gleitsystemen in der Realität nicht beobachtet (Abbildung 52). In der Darstellung bedeuten die schwarzen Linien Grenzflächen mit einer Mindestmissorientierung von  $11^\circ$ . Die Missorientierung von  $11^\circ$  wird hier verwendet, da dieser Wert bei der Analyse der Kornstruktur von Aluminiumlegierungen gebräuchlich ist. Es bilden sich bevorzugt 4 radial von den Teilchen ausgehende Grenzflächen aus, wobei diese nicht senkrecht aufeinander stehen, wie es vom isotropen Modell vorausgesagt wird.

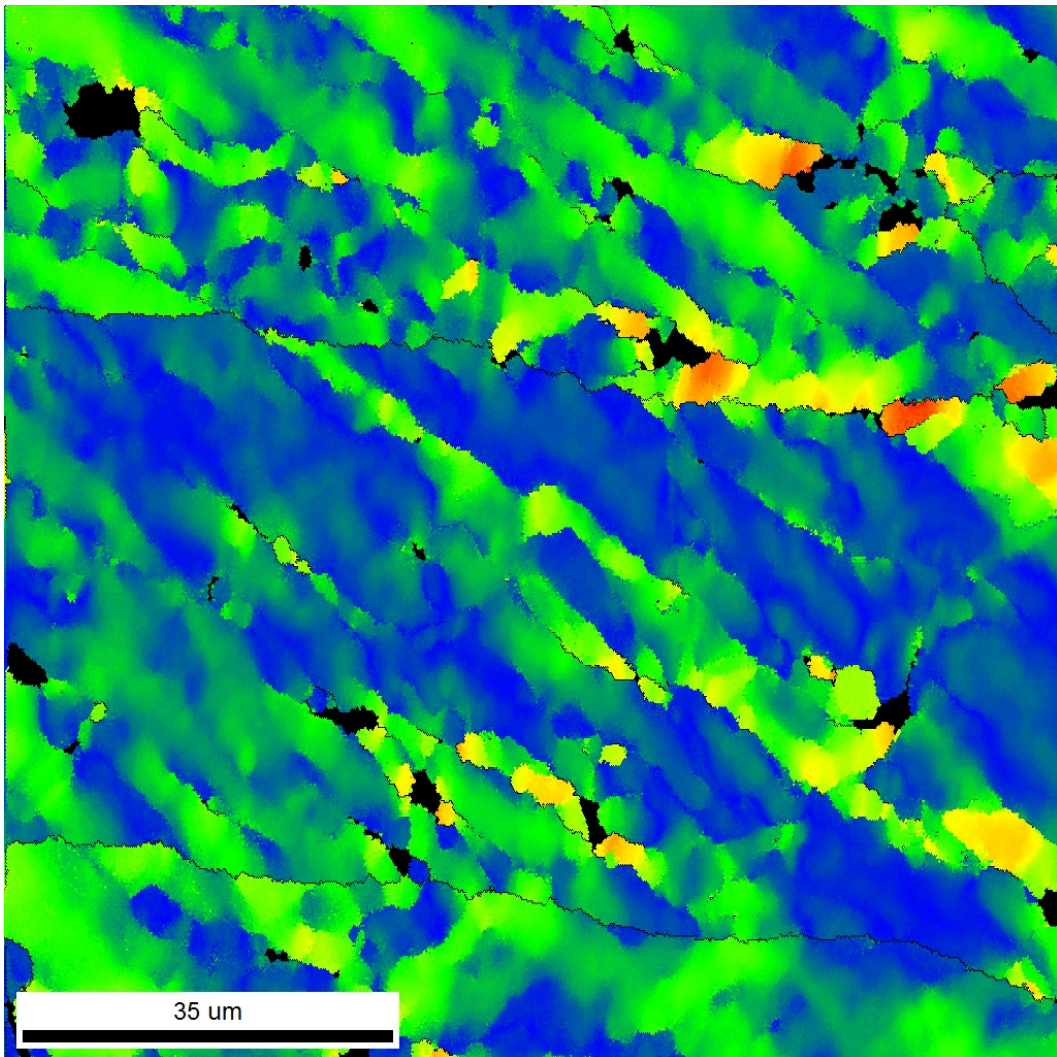


Abbildung 52: GROD-map Übersicht [6]

Die Bildung von einzelnen Grenzen innerhalb der Verformungszone reicht vermutlich nicht für die Keimbildung aus, da ein Keim für die Rekristallisation eine einheitliche Orientierung haben muss. Dies kann durch die Subkornbildung, wie die Simulationsergebnisse in Kapitel 4.2 zeigen, erreicht werden. Der Erholungsvorgang wird dabei eher schematisch dargestellt. Die Kinetik mit der dieser Vorgang vonstatten geht, wird dabei nicht betrachtet. Für die Bildung von Keimstellen für Rekristallisation spielt jedoch die Geschwindigkeit mit der die Keime gebildet werden durchaus eine Rolle, da es verschiedene mögliche Keimstellen gibt. Somit führt nicht jede mögliche Keimstelle zu einem Korn, da womöglich andere schnellere Keimstellen schon vorher zur Rekristallisation geführt haben, wodurch die treibende Kraft abgebaut wird.

Die in Kapitel 4.3 gezeigten Parameterstudien zeigen den Einfluss einzelner Parameter auf die Subkornbildung bzw. welche Missorientierung in Bezug auf die unrotierte Matrix erreicht werden kann. Wird die Keimbildung als abnormales Subkornwachstum verstanden und verwendet man  $15^\circ$  als minimalen Orientierungsunterschied für wachstumsfähige Keimstellen, so müssen die gebildeten Subkörner mindestens um diesen Betrag von der Matrix missorientiert sein. Je höher die Missorientierung ist, desto höher ist auch die Wahrscheinlichkeit für Keimbildung. Unter dieser Prämisse kommt man mit Abbildung 41 zu dem Ergebnis, dass mit zunehmendem Teilchendurchmesser und Dehnung die Keimbildungswahrscheinlichkeit steigt bzw. bei größeren Teilchen weniger Dehnung benötigt wird, um die besagte Mindestgrenze von  $15^\circ$  Missorientierung zu erreichen. Mit zunehmender Subkorngröße bzw. Keimgröße nimmt deren Rotation ab. Die Subkorngröße wurde in dieser Arbeit als gegeben angenommen. In Wirklichkeit hängt sie vom Verformungs- und Glühzustand ab wie es in Abbildung 17 dargestellt ist. Untersuchungen der Subkornstruktur werden jedoch meist in einphasigen Werkstoffen ohne große Rotationsgradienten durchgeführt. Im Nahbereich von Einschlüssen wird jedoch eine schnellere Erholung beobachtet, was mit den vorhandenen Rotationsgradienten zusammenhängen kann [3].

Da das in dieser Arbeit verwendete FE-Modell keine Größenabhängigkeit hat, wurde für die FE-Lösung nur ein Modell mit der Teilchengröße  $1\mu\text{m}$  simuliert. Die Größenabhängigkeit ergibt sich ausschließlich durch die Definition von Subkorngrößen im Postprocessing. Aus diesem Grund kann die Teilchen- und Subkorngrößenabhängigkeit zu einem einzigen Parameter, der relativen Teilchengröße (dem Verhältnis Teilchengröße zu Subkorngröße), zusammengefasst werden. Das führt zu der Aussage, dass die Missorientierung mit zunehmender relativer Teilchengröße zunimmt.

Aus dem hier Gesagten lässt sich theoretisch ableiten, dass in Werkstoffen mit kleineren Subkörnern die maximale Missorientierung größer ist als in Werkstoffen mit größeren Subkörnern. In [14] wird für die Dehnung  $\varepsilon=1,5$  ein mittlerer Subkorndurchmesser von  $0,29\mu\text{m}$  in Kupfer und  $0,6\mu\text{m}$  in Aluminium angegeben. Daraus ergibt sich, dass bei gleicher Dehnung in Kupfer eine höhere Missorientierung auftreten sollte als bei Aluminium. Die Größe der Subkörner spielt vor allem dann eine große Rolle, wenn sie die Größe der Teilchen erreichen. In Abbildung 53 wird die Subkornstruktur einer kaltverformten Aluminiumprobe mit einigen Einschlüssen gezeigt.

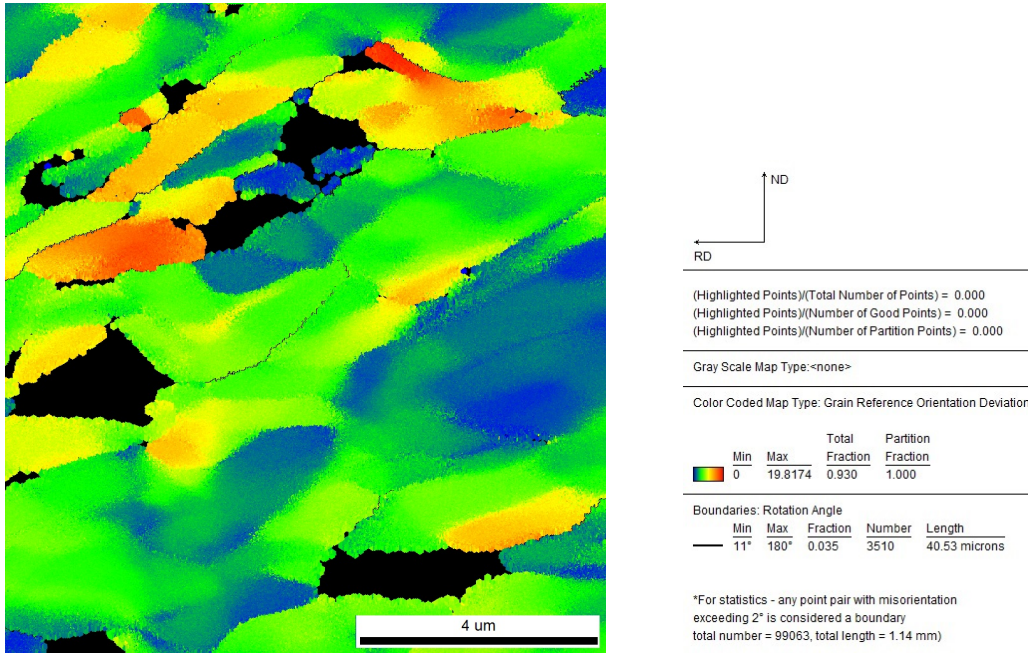


Abbildung 53: GROD-map in der Nähe von kleinen Teilchen nach Kaltverformung  $\epsilon \approx 0,5$  [6]

Als endgültiges Keimbildungskriterium wird schlussendlich die Missorientierung benachbarter Subkörner herangezogen (Abbildung 46). Dabei gibt es die Möglichkeit alle Nachbarsubkörner oder nur jenes in radialer Richtung miteinzubeziehen. Die beiden in Umfangsrichtung des Teilchens angrenzenden Subkörner bereiten insofern Probleme, als dass die Missorientierung sehr stark von der Definition der Subkörner beeinflusst wird. Bei kleinen Subkörnern befinden sich alle Nachbarn im gleichen Quadranten und deren Missorientierung ist gering. Sind die Subkörner relativ groß, so befinden sich zwei Subkörner in anderen Quadranten in denen sich die Rotationsrichtung umkehrt (Abbildung 54). In diesem Fall könnte das zentrale Subkorn entlang der grün eingezeichneten Grenzflächen ein Stück weit wachsen, bis diese Quadranten aufgebraucht sind. Ein ähnlicher Vorgang ist auch in der Realität bekannt, wo es oft zur Bildung von sogenannten Inselkörnern kommt. Ein Beispiel davon wird in Abbildung 16 gezeigt, wo ein neues Korn in der Nähe eines Teilchens zu wachsen beginnt, jedoch dann von einem schneller wachsenden Korn umwachsen wird. Als Erklärungsgrund hierfür wird in der Literatur jedoch weniger der Orientierungsgradient, sondern die erhöhte gespeicherte Versetzungsenergiedichte in der Nähe der Teilchen herangezogen, wodurch die treibende Kraft für die Wanderung von Grenzflächen erhöht ist (siehe Gleichung 15 und 16). Nach Gleichung 16 ist für die Bewegung von Grenzflächen jedoch auch deren Mobilität wichtig, welche wiederum nach Abbildung 15 stark von der Missorientierung abhängt. Da die Prozesse Erholung und Rekristallisation simultan ablaufen, ist es vorstellbar, dass es in einem frühen Stadium der Erholung zur Bildung von mobilen Grenzflächen in Bereichen großer Rotationsgradienten (z.B. zwischen den Quadranten mit entgegengesetzter Rotation) kommt, wodurch

es relativ rasch zum Anwachsen einzelner Körner kommt. Damit der Keim jedoch in die Matrix wachsen kann, müssen sich in alle Richtungen entsprechende Grenzflächen ausbilden. Da dieser Vorgang jedoch länger dauert (bzw. eventuell niemals die geforderte Missorientierung erreicht wird) können in der Zwischenzeit andere Körner wachsen und zur vollständigen Rekristallisation führen.

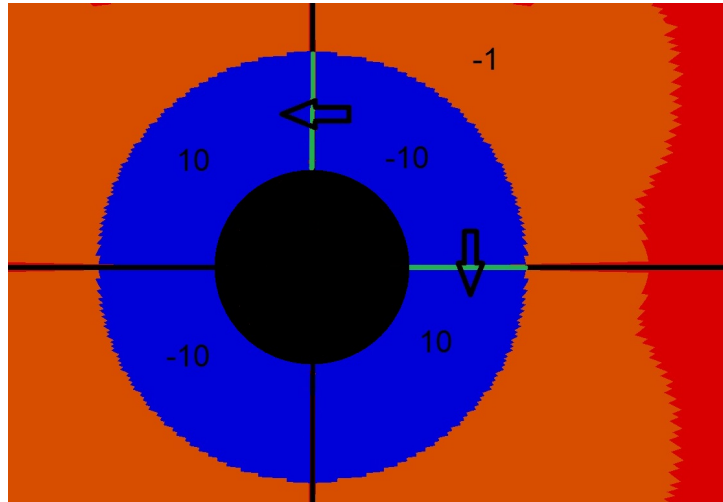


Abbildung 54: Mögliches Anwachsen der Subkörner über den Quadranten hinaus

Da die Inselkörner für die Mikrostruktur keine so große Bedeutung haben, wird nur die Orientierung des zentralen Subkornes mit dem radial nach außen angrenzenden Subkorn verglichen und als Keimbildungskriterium herangezogen. In Abbildung 47 wird in Abhängigkeit von der Subkorngröße dargestellt, ob Keimbildung möglich ist. In Abbildung 18 sind die experimentellen Beobachtungen dazu zu sehen. Zur besseren Vergleichbarkeit werden die Simulationsergebnisse für  $2\mu\text{m}$  Subkorn Durchmesser mit der gleichen Ausrichtung der Achsen und als Funktion der Dickenreduktion anstatt der wahren Dehnung dargestellt (Abbildung 55).



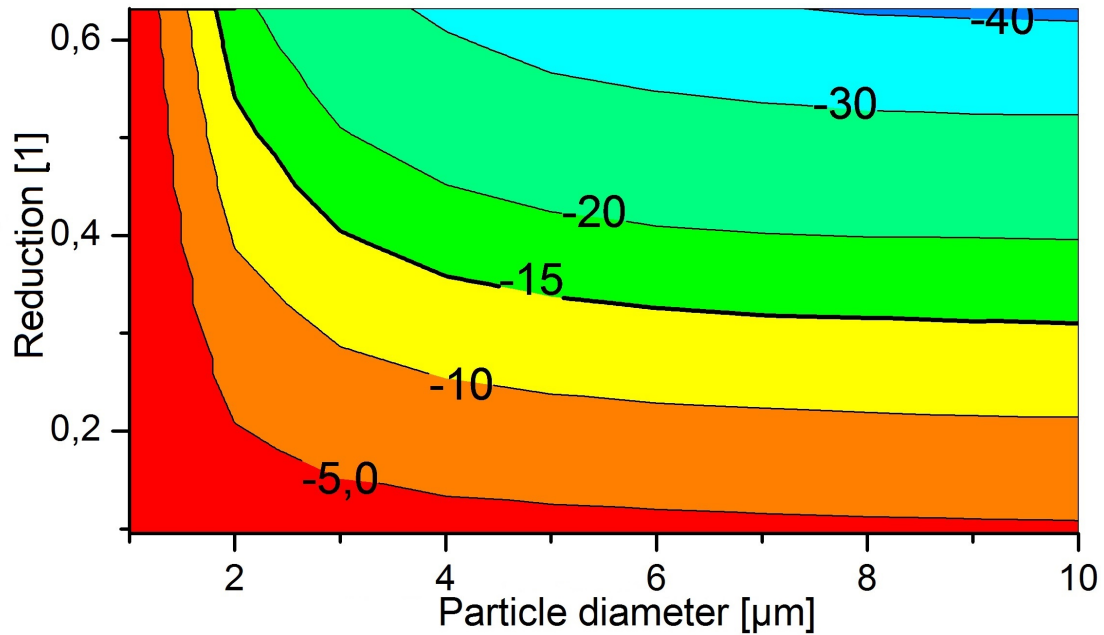


Abbildung 55: Keimbildung als Funktion der Teilchengröße und der Dickenreduktion

Qualitativ erhält man ein ähnliches Ergebnis. Im Bereich von großen Teilchen ist die Kurve, die man durch die Simulation erhält flacher, als im Experiment. D.h. in der Simulation bleibt die Wahrscheinlichkeit für Keimbildung ab einem Teilchendurchmesser von etwa  $6\mu\text{m}$  konstant. Quantitativ kann die kritische Linie, also die Linie mit  $15^\circ$  Missorientierung, durch die Wahl des Subkorndurchmessers verschoben werden. Die gewählte Subkorngröße von  $2\mu\text{m}$  scheint ein realistischer Wert zu sein, da er sich innerhalb der Größenverteilung der Subkörner in Abbildung 53 befindet.

## 6. Ausblick

Das hier eingeführte Modell zur Beschreibung der Keimbildung an harten Teilchen scheint plausible Ergebnisse zu liefern. Nichtsdestotrotz gibt es einige Ansatzpunkte für Verbesserungen.

Das verwendete FE-Modell hat die Schwächen, dass es sowohl die Anisotropie des Matrixwerkstoffes als auch die fehlende Sensitivität bezüglich jeglicher auf Gradienten basierende Effekte wie z.B. der bereits erwähnten Größenabhängigkeit der maximalen Missorientierung in der Nähe von Teilchen unterschiedlicher Größe nicht berücksichtigt. Eine Möglichkeit diesen Problemen entgegenzuwirken ist die Verwendung entsprechender kristallplastischer FE-Modelle (CPFE). Dabei kann das Abgleiten an den verschiedenen Gleitsystemen betrachtet werden. Der gesamte plastische Geschwindigkeitsgradient

ergibt sich dabei zu

$$L_p = \sum_{\alpha=1}^{N_{slip}} \dot{\gamma}^\alpha \vec{m}^\alpha \otimes \vec{n}^\alpha \quad (80)$$

wobei  $\vec{m}^\alpha$  die Gleitrichtung und  $\vec{n}^\alpha$  der Normalenvektor des Gleitsystems  $\alpha$  ist.  $N_{slip}$  bezeichnet die Anzahl der Gleitsysteme, welche im Fall des kfz-Gitters 12 ist. In konstitutiven Modellen wird die Scherrate  $\dot{\gamma}^\alpha$  als Funktion der resultierenden Scherspannung  $\tau^\alpha$  formuliert [15].

$$\dot{\gamma}^\alpha = \dot{\gamma}_0 \left| \frac{\tau^\alpha}{\tau_c^\alpha} \right|^n \text{sgn}(\tau^\alpha) \quad (81)$$

$\tau_c^\alpha$  ist dabei eine kritische Schubspannung,  $\dot{\gamma}_0$  und  $n$  sind Materialparameter. In physikalischen Modellen kann die Scherrate als Funktion der Versetzungsbewegung beschrieben werden.

$$\dot{\gamma}^\alpha = \rho_m^\alpha b v^\alpha \quad (82)$$

$\rho_m^\alpha$  ist die Dichte der mobilen Versetzungen,  $b$  der Burgersvektor und  $v^\alpha$  die Geschwindigkeit der Versetzungen. Der Vorteil dieser physikalischen Modelle ist, dass geometrisch notwendige Versetzungen (GND) berücksichtigt werden können. Diese hängen mit den Dehnungsgradienten zusammen und ermöglichen dadurch, dass Größeneffekte simuliert werden können. Die Änderung der geometrisch notwendigen Versetzungsdichte wurde von Ma et al. [16] abgeleitet.

Neben der zusätzlichen Verfestigung spielen GND die entscheidende Rolle bei der Bildung von Subkörnern mit großen Missorientierungen wie es in dieser Arbeit modelliert wurde.

Neben einem physikalisch entsprechenden FE-Modell für die plastische Verformung ist ein physikalisches Modell zur Bildung der Subkörner notwendig. In dieser Arbeit wurde die Subkorngröße vorgegeben. Es stellt sich die Frage, ob es möglich ist das Anordnen der GND in Zellwänden und Subkörnern zu modellieren. Nach [17] bleibt die Modellierung von Substrukturen wie Zellen eine Herausforderung für die physikalisch basierte CPFE. Für einfache Substrukturen können dafür diskrete versetzungsdynamische Simulationen durchgeführt werden wie in [7] (siehe Abbildung 56). Für komplexe Situationen mit vielen Versetzungen scheint diese Methode aufgrund des numerischen Aufwands jedoch nicht anwendbar zu sein.

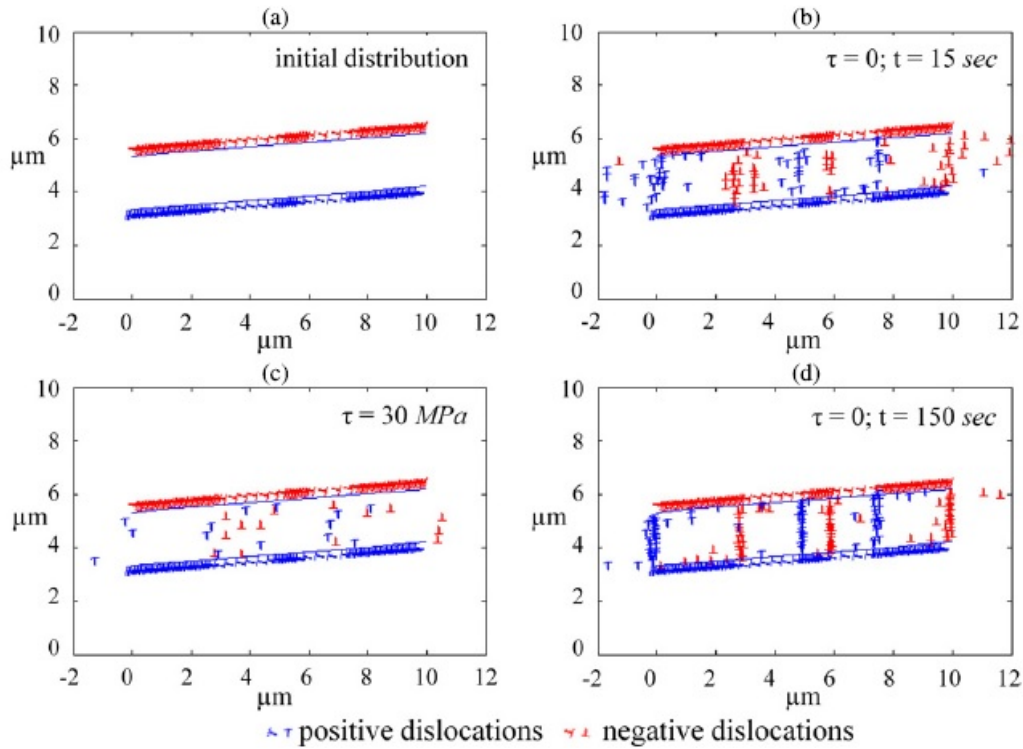


Abbildung 56: Substrukturen mit diskreten Versetzungen [7]

Unabhängig von der Einführung komplexerer Werkstoffmodelle könnten mit isotropen Modellen, wie sie in dieser Arbeit verwendet wurden, weitere Untersuchungen hinsichtlich komplexerer Geometrien gemacht werden. Da ein Teil der lokalen Phänomene wie Rotationen und Rotationsgradienten rein auf die Randbedingungen zurückzuführen sind, können durch Variation der Teilchenform bzw. der aufgetragenen Belastung Erkenntnisse gewonnen werden. Der Vorteil von den hier verwendeten Modellen ist, dass neben dem geringen Modellierungs- und Programmieraufwand die Rechenzeit wesentlich kürzer ist und somit geometrisch komplexere Modelle betrachtet werden können, als es für CPFE-Modelle der Fall ist.

Neben zusätzlicher geometrischer Komplexität kann das vorhandene Modell durch Betrachtung der Kinetik der Subkornbildung erweitert werden. Dazu wird es notwendig sein Erholungsmodelle für die Bildung von Subkörnern in Bereichen inhomogener Verformung aufzustellen. Diese müssen die Rotationsgradienten berücksichtigen, um die beschleunigte Erholung in Nahbereich von harten Teilchen abbilden zu können.

# Teil III.

## Anhang

### 7. Beschreibung der Erholung im Nahbereich eines harten Einschlusses

Im Hauptteil dieser Arbeit wurde die Keimbildung für die Rekristallisation als ein rein geometrisches Problem betrachtet, in dem versucht wurde, die Bildung von mobilen Grenzflächen, also Korngrenzen, zu beschreiben. In diesem Zusammenhang wurden Orientierungsgradienten im Nahbereich von harten Einschlüssen untersucht. Ein anderer Zugang ist die Beschreibung der treibenden Kräfte auf diese Grenzflächen, wie sie in Gleichung 15 beschrieben wird. In diesem Zusammenhang sind Gradienten der Versetzungsdichte im Nahbereich von harten Einschlüssen von Interesse. In der Arbeit von Mitter und Sommitsch [8] wurde ein Modell vorgestellt, in dem die inhomogene Versetzungsdichteverteilung im Nahbereich von harten Teilchen bzw. Poren beschrieben wird. Es wird dabei der Verlauf der Versetzungsdichte und der Punktdefektkonzentration gekoppelt betrachtet. Der Grund hierfür ist, dass bei der statischen Erholung, also beim Abbau von Versetzungen nach der Verformung, Leerstellen notwendig sind damit Stufenversetzungen ihre Gleitebene verlassen können [18].

#### 7.1. Modell zur Beschreibung der lokalen Erholung

In der Originalfassung beinhaltet das Modell sechs gekoppelte Differentialgleichungen, wobei der zeitliche Verlauf der Versetzungsdichte, der Leerstellenkonzentration, der Zwischengitteratomkonzentration, der Porosität, der Porengröße und der Temperatur berechnet wird. Im Zentrum steht dabei der Verlauf der lokalen Versetzungsdichte, welche als Funktion der Verfestigung aufgrund plastischer Dehnung und der Erholung aufgrund Versetzungsannihilation beschrieben wird. Versetzungsannihilation entsteht durch Klettern und Gleiten von Versetzungen, wodurch es den Stufenversetzungen möglich wird, auf Stufenversetzungen entgegengesetzten Vorzeichens zu treffen. Klettern wird ermöglicht, indem sich Leerstellen oder Zwischengitteratome an die Versetzungslinie anlagern oder abspalten. Leerstellen und Zwischengitteratome können durch verschiedene Mechanismen gebildet werden. Im betrachteten Modell werden mechanisch und thermisch erzeugte Punktdefekte berücksichtigt. Mechanisch erzeugte Punktdefekte entstehen durch das Schneiden von Versetzungslinien während der plastischen Verformung. Thermisch erzeugte Punktdefekte können durch zwei verschiedene Mechanismen entstehen. Beim Schottkymechanismus wandert ein Atom an eine freie Oberfläche, während eine Leerstelle in den Pulk wandert. Im Gegensatz dazu steht der Frenkelmechanismus, wo ein Atom auf einen Zwischengitterplatz wandert und somit zugleich eine Leerstelle und ein Zwischengitteratom geschaffen wird. Da die Aktivierungsenergie für den Frenkelmechanismus höher als für den Schottkymechanismus ist und nach Gleichung 1 die Konzentration an Punktdefekten exponentiell von der Aktivierungsenergie abhängt, sind Zwischengitteratome

von untergeordneter Bedeutung. Für den Schottkymechanismus werden Oberflächen benötigt. Hier kommen hauptsächlich innere Oberflächen wie Poren und Korngrenzen in Frage. Aus diesem Grund wird neben der Versetzungsdichte und den Punktdefekten auch die Porosität und Porengröße betrachtet. Die Temperatur ist insofern von Bedeutung, als dass sowohl die Bildung als auch die Wanderung von Punktdefekten von der Temperatur abhängen. Zur Vereinfachung kann sie jedoch als räumlich konstant angenommen werden.

## 7.2. Vereinfachtes Modell

Das vorgeschlagene Modell hat eine große Anzahl an Parametern, die experimentell nur schwer bestimmbar sind. Außerdem stellt die Kopplung der Differentialgleichungen einen nicht unwesentlichen numerischen Aufwand dar. Aus diesen Gründen wurde das Modell auf den Kern reduziert. Dabei werden anstatt aller Punktdefekte nur noch die Leerstellen betrachtet. Darüber hinaus werden nur Leerstellen betrachtet, die mit Versetzungen interagieren. Dies sind zum einen die mechanisch erzeugten Leerstellen, sowie Leerstellen, die thermisch an Versetzungslinien erzeugt oder vernichtet werden. Die Begründung hierfür ist, dass mechanisch erzeugte Leerstellen zu einer Übersättigung führen können. So ist nach der Kaltverformung die Leerstellenkonzentration höher als die des thermischen Gleichgewichts (Gleichung 1) sein. Die thermischen Leerstellen, die an Versetzungen entstehen bzw. verschwinden, müssen betrachtet werden, da erst dadurch die Erholung der Versetzungen beschrieben werden kann. Alle anderen Quellen und Senken für Leerstellen werden weggelassen, da sie überwiegend dazu beitragen, dass das thermische Gleichgewicht erreicht wird. Bei erhöhter Temperatur ist Diffusion möglich wodurch sich die Leerstellenkonzentration der Gleichgewichtskonzentration annähert. Dies bedeutet, dass bei Übersättigung an den inneren Oberflächen mehr Leerstellen verschwinden müssen als erzeugt werden. Da keine anderen Senken als die Versetzungen betrachtet werden, bedeutet das, dass sich alle überschüssigen Leerstellen an Versetzungen anlagern müssen. Damit wird gewissermaßen die Obergrenze für Erholung während des Hochheizens nach der Kaltverformung betrachtet. Die zeitliche Änderung der Leerstellen ergibt sich zu:

$$\frac{\partial c}{\partial t} = \Phi + P_{VD}^M + P_{VD}^T - R_{VD}^T \quad (83)$$

Dabei ist  $\Phi$  die Leerstellendiffusion,  $P_{VD}^M$  ist die Produktionsrate von mechanisch erzeugten Leerstellen,  $P_{VD}^T$  ist die Produktionsrate von thermischen Leerstellen erzeugt an Versetzungen und  $R_{VD}^T$  ist die Annihilationsrate von Leerstellen an Versetzungen. Für die Versetzungsdichte gilt

$$\frac{\partial \rho}{\partial t} = WH - A \quad (84)$$

wobei WH (work hardening) ein Term ist, der die Kaltverfestigung beschreibt und A der Annihilationsterm, der die thermische Erholung der Versetzungen beschreibt. Geht man weiter ins Detail, so erhält man für die einzelnen Terme:

(Alle Faktoren, die nicht im Text beschrieben sind, sind in Tabelle 1 ersichtlich.)

$$\Phi = D \left( \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) \quad (85)$$

$$D = 4b^2\nu_l f_c \exp\left(-\frac{A_{Vm}}{kT}\right) \quad (86)$$

Für die Leerstellenkonzentration nach der Kaltverformung wurde in der Literatur [19] (bzw. ursprünglich veröffentlicht in [20]) folgende Gleichung gefunden

$$c = \frac{1}{2} \frac{\beta}{G} \int_0^\varepsilon \sigma d\varepsilon \quad (87)$$

wobei bei dem betrachteten Werkstoff in [19] für die Konstante  $\beta$  der Wert 0,06 verwendet wurde. In dieser Arbeit wird der Wert  $\beta = 0,08$  verwendet, da dadurch bessere Übereinstimmung mit experimentellen Beobachtungen erreicht wird. Durch Überführen in die Ratenform und durch Ausnutzen des Zusammenhangs zwischen Fließspannung und Versetzungsdichte aus Gleichung 12 wird folgende Gleichung erhalten

$$P_{VD}^M = \frac{\beta}{2} \alpha^* b \sqrt{\rho} \dot{\varepsilon} \quad (88)$$

wobei  $\alpha^* = \alpha\sqrt{3} = 0,866$  den Zusammenhang zwischen Schubspannung und Normalspannung beschreibt [8].

Die thermisch produzierten Leerstellen ergeben sich mit [8] zu

$$P_{VD}^T = \frac{K_{VD}}{N} \frac{\rho}{b} \nu_L \exp\left(-\frac{A_{Vc}}{kT}\right) \quad (89)$$

Ebenfalls aus [8] entnommen wurde

$$R_{VD}^T = K_{RVD} 2Dc\rho \quad (90)$$

wobei die Konstante  $K_{RVD}$  angefügt wurde. Der Grund hierfür ist, dass die Terme  $P_{VD}^T$  und  $R_{VD}^T$  nicht unabhängig voneinander sind. Für den Fall von längeren Glühzeiten muss sich die Leerstellenkonzentration der Gleichgewichtskonzentration  $c = \exp(-A_{Vg}/kT)$  annähern. Das bedeutet, dass gleich viele Leerstellen erzeugt wie vernichtet werden müssen. Damit kann folgende Bedingung formuliert werden

$$\frac{K_{VD}}{N} \frac{\rho}{b} \nu_L \exp\left(-\frac{A_{Vc}}{kT}\right) = K_{RVD} 2 \cdot 4b^2\nu_l f_c \exp\left(-\frac{A_{Vm}}{k * T}\right) \exp\left(-\frac{A_{Vg}}{kT}\right) \rho \quad (91)$$

Da  $A_{Vc} = A_{Vg} + A_{Vm}$  gilt und  $N = \sqrt{2}b^{-3}$  [8] kann man daraus einen Zusammenhang zwischen  $K_{VD}$  und  $K_{RVD}$  herleiten

$$K_{RVD} = 0,088K_{VD} \quad (92)$$

d.h. nur mehr eine Konstante muss gewählt werden. In dieser Arbeit wurde  $K_{VD} = 1$  gesetzt. Für die Kaltverfestigung WH wurde die Beziehung von Kocks und Mecking verwendet

$$\frac{\partial \rho}{\partial \varepsilon} = k_1 \sqrt{\rho} - k_2 \rho \quad (93)$$

bzw. in der hier verwendeten Ratenform

$$\frac{\partial \rho}{\partial t} = k_1 \sqrt{\rho} \dot{\varepsilon} - k_2 \rho \dot{\varepsilon} \quad (94)$$

für den Annihilationsterm  $A$  wurde wieder auf [8] zurückgegriffen

$$A = K_{RD} 2 D c \rho (K_{RD} 2 N b^2 r \rho) \quad (95)$$

wobei  $K_{RD}$  eine weitere zusätzlich angefügte Konstante ist. Durch die aufgespaltete Schreibweise wird der Zusammenhang zwischen Leerstellenannihilation und Erholung von Versetzungen klar.

Da im Modell die Diffusion der Leerstellen in x- und y-Richtung betrachtet wird, ist das Modell zweidimensional. Damit jedoch die Diffusion überhaupt zu tragen kommt, muss die Leerstellenkonzentration zumindest vorübergehend inhomogen sein. Dies wird dadurch erreicht, dass die Dehnrates  $\dot{\varepsilon}$  und die damit einhergehende mechanische Produktion der Leerstellen vom Ort abhängt. Die örtliche Dehnrates wird dabei in einer vorhergehenden FE-Simulation bestimmt. In der FE-Simulation wird die lokale Dehnung des Matrixwerkstoffes in der Nähe eines harten Teilchens berechnet. Da im betrachteten Modell die Werkstoffdaten für Kupfer verwendet werden, liegt der FE-Simulation eine Fließkurve für polykristallines Kupfer zu Grunde.

### 7.3. Simulation

In Tabelle 1 werden alle Größen, die für die Simulation gebraucht werden, zusammengefasst.

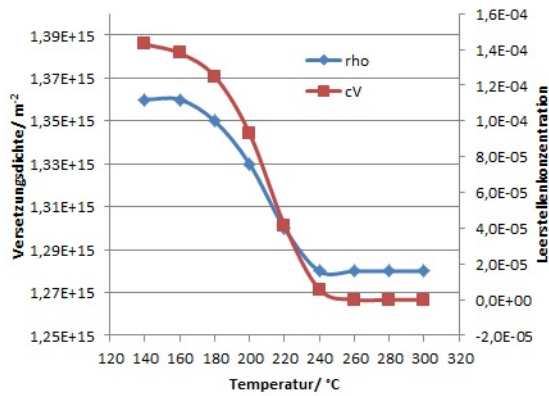
#### 7.3.1. Lösung für homogene Verformung

Als erster Schritt wird das Modell auf ein Feld mit homogener Verformung angewendet. Die Erholungskonstante  $K_{RD}$  wird dabei auf 0,1 gesetzt. Es wird in 10 Sekunden auf  $\varepsilon = 0,6$  verformt. Anschließend wird in 60 Sekunden auf die Zieltemperatur aufgeheizt und 600 Sekunden bei konstanter Temperatur gegläht. Die Versetzungsdichte und Leerstellenkonzentration am Ende dieses Vorgangs wird gegen die Temperatur aufgetragen und mit Versuchsergebnissen verglichen (Abbildung 57). Bei Abbildung 57b handelt es sich um einen Versuch an ECAP-Kupfer, welches nach der Verformung bei unterschiedlichen Temperaturen gegläht wurde. Anschließend wurde der Restwiderstand gemessen [21]. Der Restwiderstand gibt Aufschluss über die Gesamtdefektdichte, also die Leerstellen- und Versetzungsdichte. Bei den hochverformten Proben findet der Abfall bei etwa 200°C statt. Bei der Probe mit Scherung 0,85, was in etwa der Dehnung in der Simulation entspricht, ist dies erst bei ca. 250°C der Fall. Die direkte Vergleichbarkeit von Versuch und Simulation ist insofern schwierig, als bei der Simulation nur der Mechanismus der Erholung betrachtet wird. Beim Versuch dürfte jedoch tatsächlich auch Rekristallisation auftreten, da ansonsten der große Abfall des Widerstands nicht erklärbar ist. Die einzige Bestätigung aus dem Versuch ist, dass ab etwa 200°C Diffusionsvorgänge vonstatten gehen können, was auch bei der Simulation zu sehen ist.

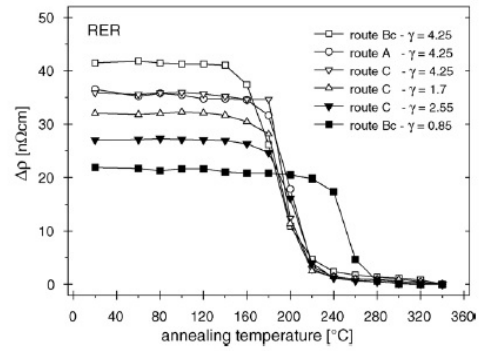
Tabelle 1: Parameterliste

Symbol	Beschreibung	Einheit	Wert
$c$	Leerstellenkonzentration	-	
$\rho$	Versetzungsdichte	$m^{-2}$	
$t$	Zeit	s	
$P_{VD}^M$	mechanisch erzeugte Leerstellen	$s^{-1}$	
$\beta$	Prop.Faktor für die mech. Leerstellenproduktion	-	0,08
$\alpha^*$	Prop. aktor zur Berechnung der Fließspannung	-	0,866
$G$	Schubmodul	MPa	40000
$b$	Burgersvektor	m	2,5E-10
$\dot{\epsilon}$	Dehnrage	$s^{-1}$	
$P_{VD}^T$	thermisch erzeugte Leerstellen an Versetzungen	$s^{-1}$	
$K_{VD}$	Konstante therm. erzeugte Leerstellen	-	1
$N$	Anzahl der Gitterplätze je Einheitsvolumen	$m^{-3}$	9,05E28
$\nu_L$	Frequenz Gitterschwingungen	$s^{-1}$	3,0E12
$A_{Vc}$	Aktivierungsenergie Versetzungsklettern	J/Atom	3,348E-19
$A_{Vg}$	Aktivierungsenergie Leerstellenproduktion	J/Atom	1,65E-19
$A_{Vm}$	Aktivierungsenergie Leerstellenmigration	J/Atom	1,698E-19
$k$	Boltzmannkonstante	J/KAtom	1,3806E-23
$T$	Absoluttemperatur	°K	
$R_{VD}^T$	Annihilation von Leerstellen an Versetzungen	$s^{-1}$	
$K_{RVD}$	Konstante Annihilation Leerstellen	-	0,088
WH	Kaltverfestigung	$m^{-2}s^{-1}$	
$k_1$	Koeffizient Verfestigung	$m^{-1}$	4,3E8
$k_2$	Koeffizient dynamische Erholung	-	11,28
$A$	Annihilation von Versetzungen	$m^{-2}s^{-1}$	
$K_{RD}$	Konstante für Annihilation von Versetzungen	-	1(0,1)





(a) Isochrone Simulation



(b) Isochrone Versuch [21]

Abbildung 57: Defektdichte nach 10min Glühzeit

Da der zeitliche Verlauf während des Aufheizens und Glühens von Interesse ist wurden in einem zweiten Schritt Kurzzeitversuche simuliert. Wie aus Tabelle 1 ersichtlich wurde die Simulation für zwei verschiedene Werte für  $K_{RD}$ , nämlich 1 und 0,1, durchgeführt. Die Ergebnisse werden vergleichend gegenübergestellt. Der zeitliche Verlauf der Temperatur ist in Abbildung 58 dargestellt.

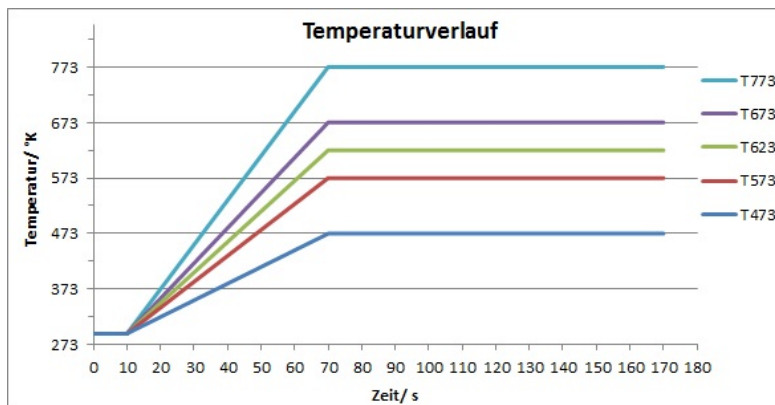
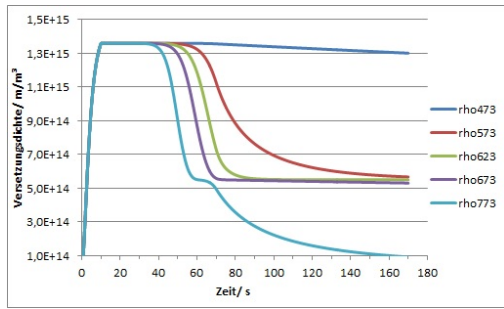
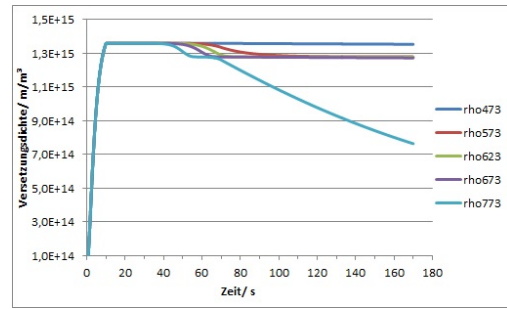


Abbildung 58: Temperaturverlauf

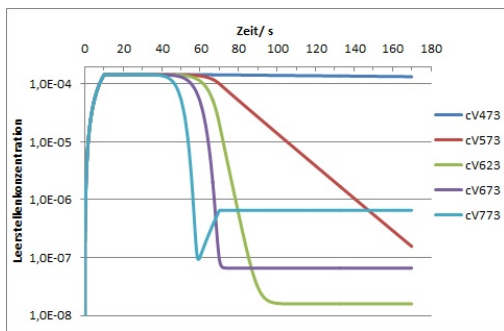


(a)  $K_{RD} = 1$

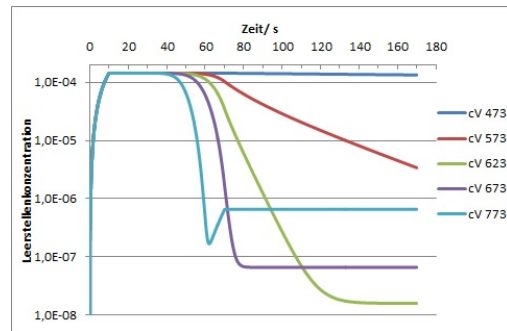


(b)  $K_{RD} = 0,1$

Abbildung 59: Versetzungsdichte nach Kaltverformung  $\varepsilon = 0,6$  und Glühen nach Abb.58

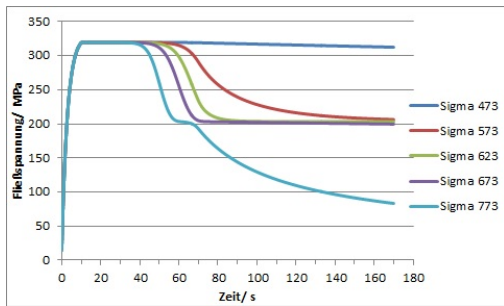


(a)  $K_{RD} = 1$

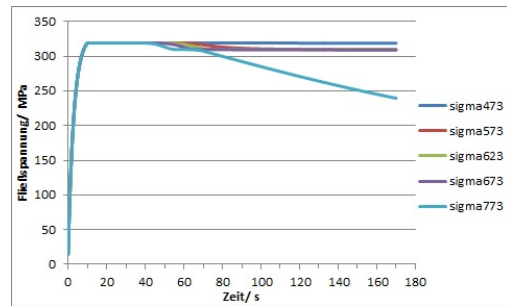


(b)  $K_{RD} = 0,1$

Abbildung 60: Leerstellenkonzentration nach Kaltverformung  $\varepsilon = 0,6$  und Glühen nach Abb.58



(a)  $K_{RD} = 1$

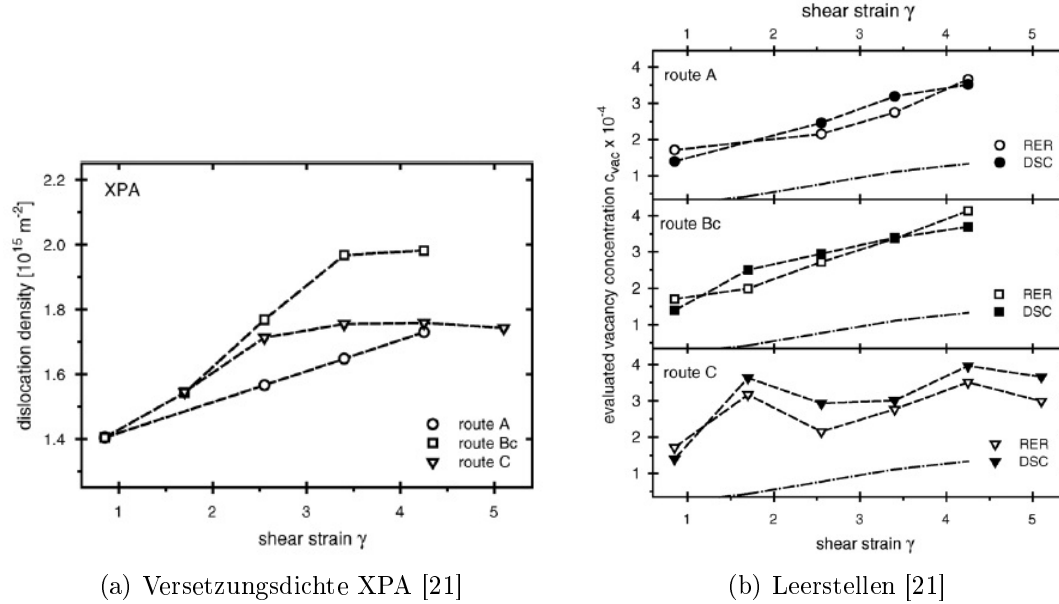


(b)  $K_{RD} = 0,1$

Abbildung 61: Fließspannung nach Kaltverformung  $\varepsilon = 0,6$  und Glühen nach Abb.58

Durch die Kaltverformung steigt die Leerstellenkonzentration sowie die Versetzungsdichte stark an. Nach der plastischen Dehnung von 0,6 wird eine Versetzungsdichte von  $1,35E15m^{-2}$  erhalten. Dies ist in guter Übereinstimmung mit experimentellen Beobach-

tungen in Abbildung 62a, wo in ECAP-Kupfer mit dieser Dehnung durch eine Röntgenpeakanalyse eine ähnliche Versetzungsdichte gemessen wurde. Für die maximale Leerstellenkonzentration ergibt sich  $1.4 \cdot 10^{-4}$  was ebenfalls mit dem Ergebnis aus Abbildung 62b im Einklang ist.



(a) Versetzungsdichte XPA [21]

(b) Leerstellen [21]

Abbildung 62: Defektdichten nach unterschiedlichen Dehnungen

Beim Hochheizvorgang beginnt zwischen  $200^\circ\text{C}$  und  $300^\circ\text{C}$  die Annihilation der Versetzungen und Leerstellen. Dabei strebt die Versetzungsdichte einen Plateauwert an. Vergleicht man das mit den Kurven der Leerstellenkonzentration, so erkennt man, dass die schnelle Erholung der Versetzungen am Beginn aufgrund der Übersättigung mit Leerstellen zustande kommt. Sobald die Leerstellen abgebaut sind verlangsamt sich die Versetzungserholung. Für den Fall mit  $500^\circ\text{C}$  fällt die Leerstellenkonzentration auf einen Minimalwert, bevor aufgrund der thermischen Aktivierung Leerstellen gebildet werden und die Leerstellenkonzentration wieder ansteigt. Für die Kurven von  $450^\circ\text{C}$  und  $400^\circ\text{C}$  wird im betrachteten Zeitraum ebenfalls das Gleichgewicht erreicht, wobei dieses mit abnehmender Temperatur bei sinkender Konzentration liegt. Aus diesem Grund ist bei  $500^\circ\text{C}$  auch noch nennenswerte Erholung erkennbar nachdem das thermische Gleichgewicht erreicht wurde, wohingegen dies bei niedrigeren Temperaturen nicht der Fall ist. Die Kurven der Fließspannung verlaufen aufgrund des direkten Zusammenhangs sehr ähnlich wie die der Versetzungsdichte. Sie erlauben eine Plausibilitätsüberprüfung der berechneten Erholung. Für den Fall mit der Konstante  $K_{RD} = 1$  wird durch die Erholung die Fließspannung von  $330\text{MPa}$  auf  $200\text{MPa}$  reduziert. Dies erscheint zu hoch, da im betrachteten Modell nur Erholung betrachtet wird und diese in der Realität wohl nicht ausreicht um diese Abnahme der Fließspannung zu bewirken. Die Ergebnisse für  $K_{RD} = 0, 1$  sind realistischer. Zur Bestimmung des idealen Wertes für  $K_{RD}$  müssten kalt-

verformte Kupferproben gegläht und die Abnahme der Fließspannung gemessen werden.

### 7.3.2. Lösung für heterogene Verformung

Zur Beschreibung von Keimbildung muss ein heterogenes Verformungsfeld betrachtet werden. Die lokalen Dehnungen wurden durch eine FE-Simulation bestimmt. Dabei wurde die Dehnung im Nahbereich eines harten Einschlusses bei einer globalen Dehnung von 0,5 bestimmt (Abbildung 63).

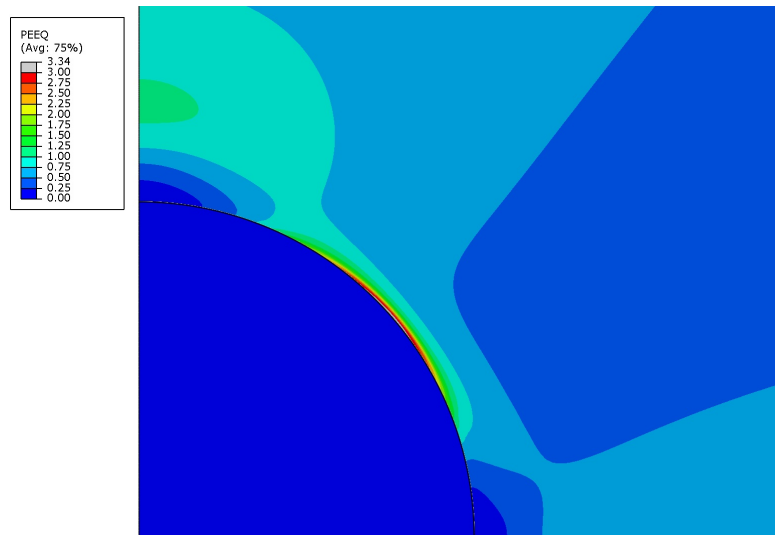


Abbildung 63: Lokale plastische Dehnung  $\varepsilon_{global} = 0,5$

Der zeitliche Ablauf von Verformung und Glühbehandlung entspricht dem Verlauf in Abbildung 58. Die Randbedingungen für die Leerstellendiffusion sind so gesetzt, dass es zu keinem Fluss über den betrachteten Bereich kommt. Das ist zulässig, da das Modell symmetrisch ist. Für die Erholungskonstante  $K_{RD}$  wurde der realistische Wert 0,1 genommen.

In Abbildung 64 wird der Verlauf der Leerstellenkonzentration als Funktion der Zeit dargestellt wobei 64a bis 64e den Verformungsvorgang beschreiben und 64f bis 64i die entscheidenden Zeitpunkte des Aufheizvorganges zeigen. Während der Verformung werden in den Bereichen mit hoher Dehnung bevorzugt Leerstellen produziert, wodurch die räumliche Verteilung der Leerstellenkonzentration sehr der Dehnungsverteilung ähnelt. Beim Aufheizen kommt es naturgemäß zum Abbau der Leerstellen, wobei sich die Temperatur bei der das passiert aufgrund der Heizrate zu höheren Temperaturen hin verschiebt, als dies bei den isochronen Simulationen der Fall war (Abbildung 57a).

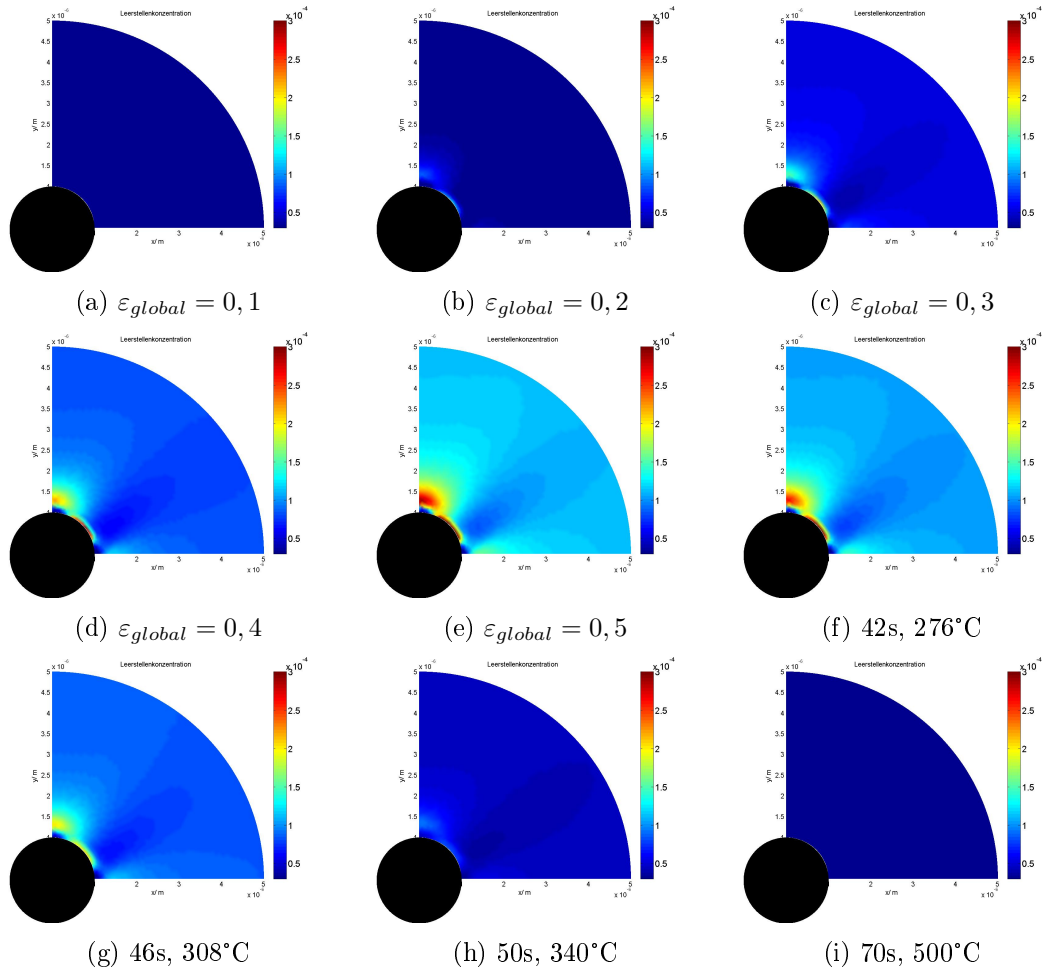


Abbildung 64: Lokale Leerstellenkonzentration zu unterschiedlichen Zeitpunkten

Entsprechend zu den gleichen Zeitpunkten wird auch die Verteilung der Versetzungsdichte gezeigt (Abbildung 65). Am Beginn des Verformungsvorganges ähnelt das Bild dem der Dehnung, jedoch wird mit zunehmender Zeit bzw. globaler Dehnung die räumliche Verteilung der Versetzungsdichte homogener. Dies ergibt sich aufgrund des verwendeten Verfestigungsgesetzes, bei dem es bei erhöhter Versetzungsdichte zu zunehmender dynamische Erholung kommt. Während des Hochheizens kommt es analog zum Abbau von Leerstellen auch zum Abbau von Versetzungen, wobei die Abnahme der Versetzungsdichte bei der verwendeten Konstante  $K_{RD}=0,1$  gering ist.

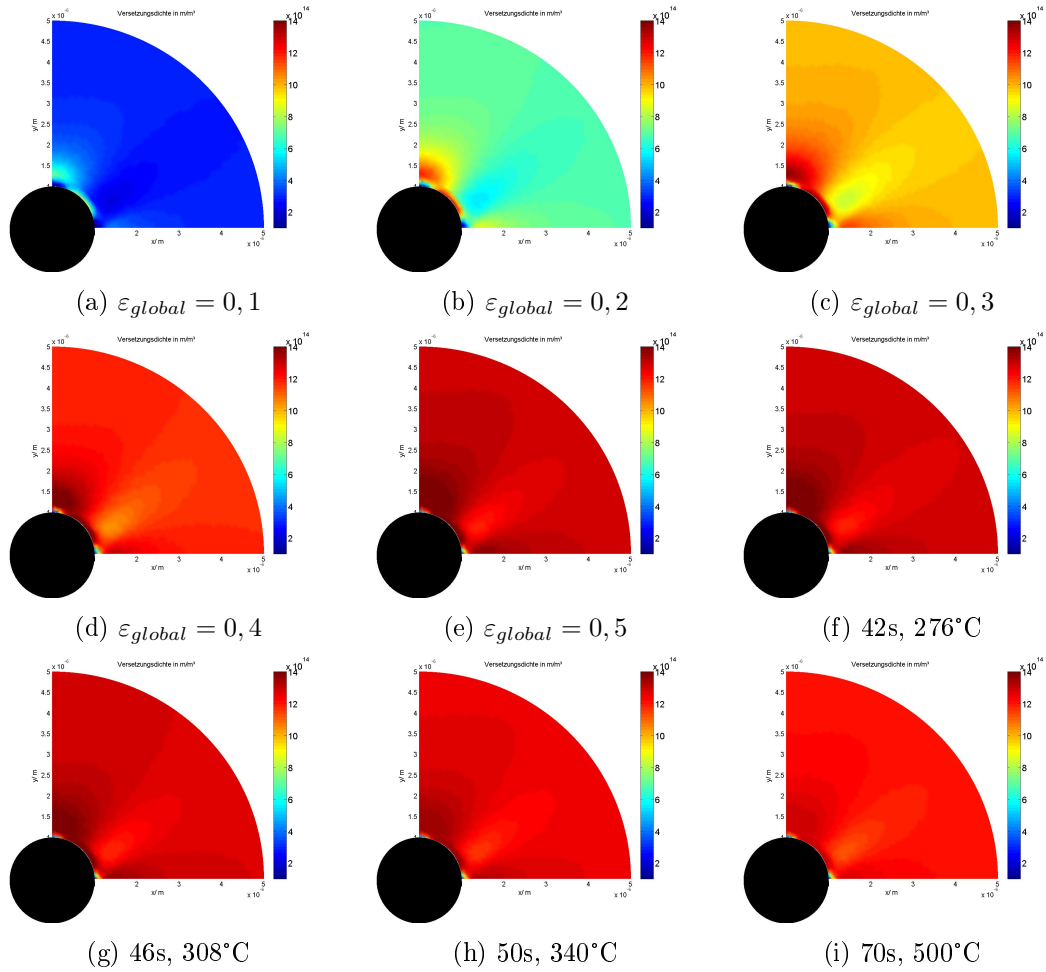
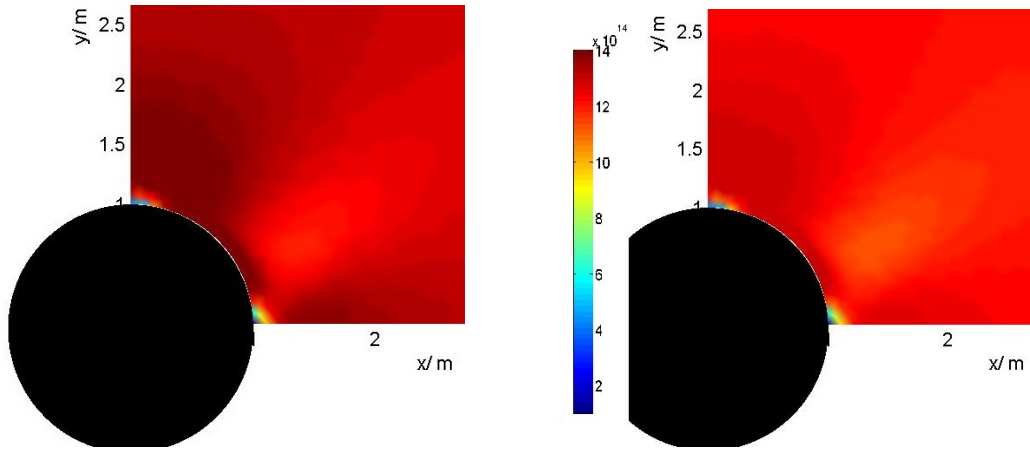


Abbildung 65: Lokale Versetzungsdichten zu unterschiedlichen Zeitpunkten

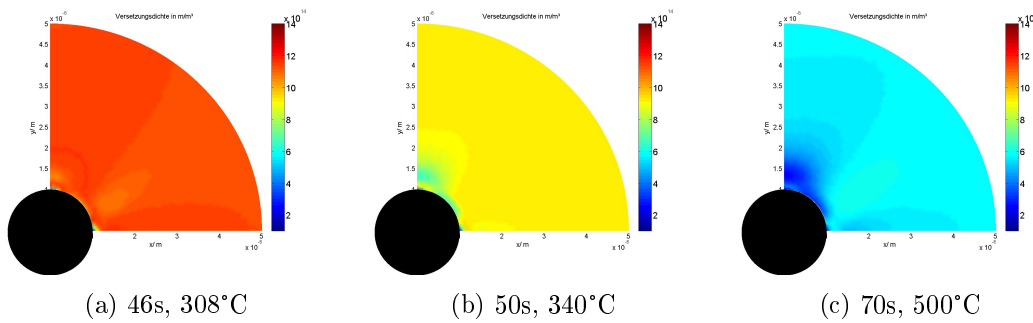
Betrachtet man die Verteilung der Versetzungsdichte nach dem Hochheizen im Detail (Abbildung 66b), so hat diese im Wesentlichen das gleiche Muster wie die Verteilung nach der Verformung (Abbildung 66a), wobei die Spitzen der Versetzungsdichte etwas geringer sind. Man erkennt deutlich, dass es in den Bereichen, wo die x- und y-Achse die Teilchenoberfläche schneiden, aufgrund des Staupunktverhaltens während der Verformung zu Minima der Versetzungsdichte kommt. Dies ist jedoch nicht auf Erholungsvorgänge zurückzuführen, sondern ergibt sich alleine aus der Dehnungsverteilung (Abbildung 63).



(a) Versetzungsdichte nach Kaltverformung      (b) Versetzungsdichte nach Hochheizen

Abbildung 66: Detail lokale Versetzungsdichten

Alternativ zur Berechnung mit der Konstanten  $K_{RD} = 0,1$  soll in Abbildung 67 gezeigt werden, was passiert, wenn man  $K_{RD} = 1$  verwendet. Es wird nun erreicht, dass in Bereichen, in denen nach der Verformung eine hohe Versetzungsdichte vorhanden ist, nach dem Aufheizen Minima der Versetzungsdichte entstehen. Verursacht wird dies durch die lokal hohe Konzentration an Leerstellen nach der Verformung, welche sich beim Aufheizen an den Versetzungen anlagern und so zu Erholung führen. Durch die 10-mal so hohe Erholungskonstante für Versetzungen können sich pro angelagerte Leerstelle 10-mal so viele Versetzungen erholen, wodurch es zu einem extremen Abbau von Versetzungen kommen kann. Wie jedoch schon in Kapitel 7.3.1 beschrieben, handelt es sich hier um einen eher unrealistischen Wert.



(a) 46s, 308°C

(b) 50s, 340°C

(c) 70s, 500°C

Abbildung 67: Lokale Versetzungsdichten zu unterschiedlichen Zeitpunkten für  $K_{VD} = 1$

#### 7.4. Zusammenfassung

Das hier verwendete vereinfachte Verfestigungs- und Erholungsmodell beschreibt die Erzeugung von Versetzungen und Leerstellen während der Kaltverformung sowie deren Abbau beim Glühen. Die mechanisch erzeugten Leerstellen können bei großer plastischer

Verformung zu Konzentrationen führen, die um Größenordnungen höher sind als es dem thermischen Gleichgewicht entspricht. Die Ergebnisse für die Versetzungsdichte sowie der Leerstellenkonzentration sind in Übereinstimmung mit Versuchsergebnissen. Ebenso scheint die Temperatur, ab der es in der Simulation zum Leerstellenabbau kommt, in einem richtigen Bereich zu liegen. In der Simulation kommt es durch die Übersättigung an Leerstellen zu einer Erholung von Versetzungen bei relativ niedrigen Temperaturen, wobei ein Plateauwert erreicht wird, sobald die Übersättigung abgebaut ist. Bei ausreichend hoher Temperatur verlieren die mechanisch erzeugten Leerstellen an Bedeutung, da ausreichend thermisch aktivierte Leerstellen erzeugt werden können.

Wird das Modell für das heterogene Verformungsfeld im Nahbereich eines harten Einschlusses angewendet, so muss berücksichtigt werden, dass die zugrundeliegenden FE-Simulation die Verteilung der plastischen Dehnung nur annähernd beschreiben kann. Da die mit Hilfe des Modells berechneten Versetzungsdichten jedoch proportional zur plastischen Dehnung sind, wird die Verteilung der Versetzungsdichte ebenfalls direkt vom Ergebnis der FE-Simulation beeinflusst. Aus diesem Grund ist es schwierig auf diese Weise eine Aussage über eine mögliche Keimbildung zu treffen. Es müssen deswegen fortgeschrittenere FE-Modelle entwickelt werden um die plastische Verformung in Mikrometerbereich angemessen beschreiben zu können.



## 8. Python Skript für die Berechnung der Subkornrotationen

(Dieses Programm wurde selbstständig unter Zuhilfenahme von [12] geschrieben.)

```
# -*- coding: mbcs -*-
# Do not delete the following import lines
from abaqus import *
from abaqusConstants import *
import __main__
import mesh
import regionToolset

Mdb()

mdb.models.changeKey(fromName='Model-1', toName='InclusionModel')
#create axisymmetric model
s = mdb.models['InclusionModel'].ConstrainedSketch(name='__profile__',
    sheetSize=20.0)
g, v, d, c = s.geometry, s.vertices, s.dimensions, s.constraints
s.sketchOptions.setValues(viewStyle=AXISYM)
s.setPrimaryObject(option=STANDALONE)
s.ConstructionLine(point1=(0.0, -10.0), point2=(0.0, 10.0))
s.FixedConstraint(entity=g[2])
#A=inclusion main axis, B= inclusion small axis, h,w height, width of matrix
A=0.5
B=A
h=max(A,B)*10
w=max(A,B)*10

#draw matrix
s.rectangle(point1=(0.0, 0.0), point2=(w, h))
s.EllipseByCenterPerimeter(center=(0.0, 0.0), axisPoint1=(0.0, B),
    axisPoint2=(A, 0.0))

s.autoTrimCurve(curvel=g[7], point1=(-A, 0))
s.autoTrimCurve(curvel=g[9], point1=(0, -B))
s.autoTrimCurve(curvel=g[6], point1=(A/2, 0))
s.autoTrimCurve(curvel=g[3], point1=(0, B/2))
#create matrix
p = mdb.models['InclusionModel'].Part(name='Matrix',
    dimensionality=AXISYMMETRIC, type=DEFORMABLE_BODY)
p = mdb.models['InclusionModel'].parts['Matrix']
p.BaseShell(sketch=s)
s.unsetPrimaryObject()
p = mdb.models['InclusionModel'].parts['Matrix']
del mdb.models['InclusionModel'].sketches['__profile__']

s1 = mdb.models['InclusionModel'].ConstrainedSketch(name='__profile__',
    sheetSize=20.0)
g, v, d, c = s1.geometry, s1.vertices, s1.dimensions, s1.constraints
s1.sketchOptions.setValues(viewStyle=AXISYM)
s1.setPrimaryObject(option=STANDALONE)
s1.ConstructionLine(point1=(0.0, -10.0), point2=(0.0, 10.0))
s1.FixedConstraint(entity=g[2])

#draw inclusion
s1.rectangle(point1=(0.0, 0.0), point2=(w, h))
s1.EllipseByCenterPerimeter(center=(0.0, 0.0), axisPoint1=(0.0, B),
    axisPoint2=(A, 0.0))

s1.autoTrimCurve(curvel=g[7], point1=(-A, 0))
s1.autoTrimCurve(curvel=g[9], point1=(0, -B))
s1.autoTrimCurve(curvel=g[6], point1=(2*A, 0))
s1.autoTrimCurve(curvel=g[3], point1=(0, 2*B))

#create inclusion
s1.delete(objectList=(g[4], ))
s1.delete(objectList=(g[5], ))
p = mdb.models['InclusionModel'].Part(name='Inclusion',
    dimensionality=AXISYMMETRIC, type=DEFORMABLE_BODY)
p = mdb.models['InclusionModel'].parts['Inclusion']
p.BaseShell(sketch=s1)
s1.unsetPrimaryObject()
p = mdb.models['InclusionModel'].parts['Inclusion']
del mdb.models['InclusionModel'].sketches['__profile__']
```

```

#loop material parameter c of johnson-cook-flowcurve
#for jcc in range(3, 4):

cc=0.75
cc_string=str(cc)
cc_string=cc_string.replace('.',',')
#text file for solutions
name='spherical inclusion_4SG_jc25_165_'+cc_string
filename = name + ".txt"
myElFile = open(filename, 'w')

#define materials
mdb.models['InclusionModel'].Material(name='alu6082')
mdb.models['InclusionModel'].materials['alu6082'].Elastic(table=((
    70000000000.0, 0.34), ))
mdb.models['InclusionModel'].materials['alu6082'].Plastic(
    hardening=JOHNSON_COOK, table=((25000000.0, 165000000.0, cc, 1.0,
    1.0, 0.0), ))

mdb.models['InclusionModel'].Material(name='Inclusion')
mdb.models['InclusionModel'].materials['Inclusion'].Elastic(table=((
    400000000000.0, 0.2), ))

#create sections
mdb.models['InclusionModel'].HomogeneousSolidSection(name='InclusionSection',
    material='Inclusion', thickness=None)
mdb.models['InclusionModel'].HomogeneousSolidSection(name='MatrixSection',
    material='alu6082', thickness=None)
p = mdb.models['InclusionModel'].parts['Inclusion']
f = p.faces
faces = f.getSequenceFromMask(mask=('[#1 ]', ), )
region = p.Set(faces=faces, name='Set-1')
p = mdb.models['InclusionModel'].parts['Inclusion']
p.SectionAssignment(region=region, sectionName='InclusionSection', offset=0.0,
    offsetType=MIDDLE_SURFACE, offsetField='',
    thicknessAssignment=FROM_SECTION)
p1 = mdb.models['InclusionModel'].parts['Matrix']
p = mdb.models['InclusionModel'].parts['Matrix']
f = p.faces
faces = f.getSequenceFromMask(mask=('[#1 ]', ), )
region = p.Set(faces=faces, name='Set-1')
p = mdb.models['InclusionModel'].parts['Matrix']
p.SectionAssignment(region=region, sectionName='MatrixSection', offset=0.0,
    offsetType=MIDDLE_SURFACE, offsetField='',
    thicknessAssignment=FROM_SECTION)
a = mdb.models['InclusionModel'].rootAssembly
a = mdb.models['InclusionModel'].rootAssembly
a.DatumCsysByThreePoints(coordSysType=CYLINDRICAL, origin=(0.0, 0.0, 0.0),
    point1=(1.0, 0.0, 0.0), point2=(0.0, 0.0, -1.0))
p = mdb.models['InclusionModel'].parts['Inclusion']
a.Instance(name='Inclusion-1', part=p, dependent=OFF)
a = mdb.models['InclusionModel'].rootAssembly
p = mdb.models['InclusionModel'].parts['Matrix']
a.Instance(name='Matrix-1', part=p, dependent=OFF)

#create subsection for meshing
a = mdb.models['InclusionModel'].rootAssembly
f11 = a.instances['Matrix-1'].faces
t = a.MakeSketchTransform(sketchPlane=f11[0], sketchPlaneSide=SIDE1, origin=(
    0, 0, 0.0))
s1 = mdb.models['InclusionModel'].ConstrainedSketch(name='__profile__',
    sheetSize=28.28, gridSpacing=0.7, transform=t)
g, v, d, c = s1.geometry, s1.vertices, s1.dimensions, s1.constraints
s1.setPrimaryObject(option=SUPERIMPOSE)
a = mdb.models['InclusionModel'].rootAssembly
a.projectReferencesOntoSketch(sketch=s1, filter=COPLANAR_EDGES)
s1.ArcByCenterEnds(center=(0, 0), point1=(0,
    5*(max(A,B))), point2=(5*(max(A,B)), 0), direction=CLOCKWISE)
a = mdb.models['InclusionModel'].rootAssembly
f1 = a.instances['Matrix-1'].faces
pickedFaces = f1.getSequenceFromMask(mask=('[#1 ]', ), )
a.PartitionFaceBySketch(faces=pickedFaces, sketch=s1)
s1.unsetPrimaryObject()
del mdb.models['InclusionModel'].sketches['__profile__']

```

```

#mesh control
a = mdb.models['InclusionModel'].rootAssembly
elemType1 = mesh.ElemType(elemCode=CAX4RH, elemLibrary=STANDARD,
    secondOrderAccuracy=OFF, hourglassControl=DEFAULT,
    distortionControl=DEFAULT)
elemType2 = mesh.ElemType(elemCode=CAX3H, elemLibrary=STANDARD,
    secondOrderAccuracy=OFF, distortionControl=DEFAULT)
a = mdb.models['InclusionModel'].rootAssembly
f1 = a.instances['Inclusion-1'].faces
faces1 = f1.getSequenceFromMask(mask=['#1'], )
f2 = a.instances['Matrix-1'].faces
faces2 = f2.getSequenceFromMask(mask=['#3'], )
pickedRegions = (faces1+faces2, )
a.setElementType(regions=pickedRegions, elemTypes=(elemType1, elemType2))
a = mdb.models['InclusionModel'].rootAssembly
f1 = a.instances['Inclusion-1'].faces
faces1 = f1.getSequenceFromMask(mask=['#1'], )
f2 = a.instances['Matrix-1'].faces
faces2 = f2.getSequenceFromMask(mask=['#2'], )
pickedRegions = faces1+faces2
a.setMeshControls(regions=pickedRegions, elemShape=QUAD)
a = mdb.models['InclusionModel'].rootAssembly
f1 = a.instances['Matrix-1'].faces
pickedRegions = f1.getSequenceFromMask(mask=['#1'], )
a.setMeshControls(regions=pickedRegions, elemShape=QUAD, technique=STRUCTURED)

fine=A/20
coarse=h/20

#seed inclusion edge
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Inclusion-1'].edges
pickedEdges = e1.getSequenceFromMask(mask=['#5'], )
a.seedEdgeBySize(edges=pickedEdges, size=0.01, deviationFactor=0.1,
    constraint=FINER)

#seed inclusion circle
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Inclusion-1'].edges
e2 = a.instances['Matrix-1'].edges
pickedEdges1 = e1.getSequenceFromMask(mask=['#2'], ) + \
    e2.getSequenceFromMask(mask=['#4'], )
a.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1, minSize=0.007,
    maxSize=0.013, constraint=FINER)
#a.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1, minSize=0.0035,
    #maxSize=0.025, constraint=FINER)

#seed matrix inner axis
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges1 = e1.getSequenceFromMask(mask=['#2'], )
a.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1, minSize=0.01,
    maxSize=0.05, constraint=FINER)

#seed matrix inner edge
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges2 = e1.getSequenceFromMask(mask=['#8'], )
a.seedEdgeByBias(biasMethod=SINGLE, end2Edges=pickedEdges2, minSize=0.01,
    maxSize=0.05, constraint=FINER)
#a.seedEdgeByBias(biasMethod=SINGLE, end2Edges=pickedEdges2, minSize=0.005,
    #maxSize=0.1, constraint=FINER)

#seed matrix middle circle
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges1 = e1.getSequenceFromMask(mask=['#1'], )
a.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1, minSize=0.02,
    maxSize=0.2, constraint=FINER)

#seed matrix outer axis
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges1 = e1.getSequenceFromMask(mask=['#80'], )
a.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1, minSize=0.05,
    maxSize=0.1, constraint=FINER)

```

```

#seed matrix outer edge
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges2 = e1.getSequenceFromMask(mask=('[#10 ]', ), )
a.seedEdgeByBias(biasMethod=SINGLE, end2Edges=pickedEdges2, minSize=0.05,
maxSize=0.1, constraint=FINER)

#seed matrix outside
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
pickedEdges = e1.getSequenceFromMask(mask=('[#40 ]', ), )
a.seedEdgeBySize(edges=pickedEdges, size=0.1, deviationFactor=0.1,
constraint=FINER)

#mesh
a = mdb.models['InclusionModel'].rootAssembly
partInstances = (a.instances['Matrix-1'], a.instances['Inclusion-1'], )
a.generateMesh(regions=partInstances)

#assign orientation
p = mdb.models['InclusionModel'].parts['Matrix']
f = p.faces
faces = f.getSequenceFromMask(mask=('[#1 ]', ), )
region = regionToolset.Region(faces=faces)
orientation=None
mdb.models['InclusionModel'].parts['Matrix'].MaterialOrientation(region=region,
orientationType=SYSTEM, axis=AXIS_3, localCsys=orientation,
fieldName='', additionalRotationType=ROTATION_NONE, angle=0.0,
additionalRotationField='', stackDirection=STACK_3)

p = mdb.models['InclusionModel'].parts['Inclusion']
f = p.faces
faces = f.getSequenceFromMask(mask=('[#1 ]', ), )
region = regionToolset.Region(faces=faces)
orientation=None
mdb.models['InclusionModel'].parts['Inclusion'].MaterialOrientation(
region=region, orientationType=SYSTEM, axis=AXIS_3,
localCsys=orientation, fieldName='',
additionalRotationType=ROTATION_NONE, angle=0.0,
additionalRotationField='', stackDirection=STACK_3)

#create connection
a = mdb.models['InclusionModel'].rootAssembly
s1 = a.instances['Matrix-1'].edges
side1Edges1 = s1.getSequenceFromMask(mask=('[#4 ]', ), )
region1=regionToolset.Region(side1Edges=side1Edges1)
a = mdb.models['InclusionModel'].rootAssembly
s1 = a.instances['Inclusion-1'].edges
side1Edges1 = s1.getSequenceFromMask(mask=('[#2 ]', ), )
region2=regionToolset.Region(side1Edges=side1Edges1)
mdb.models['InclusionModel'].Tie(name='Connect', master=region1, slave=region2,
positionToleranceMethod=COMPUTED, adjust=ON, tieRotations=ON,
thickness=ON)

#loop true strain matrix
for i in range(1, 11):

    eps=i/10.
    strainrate=0.1
    eps_eng=abs(exp(-eps)-1)

    print 'eng. strain= ', eps_eng

    #create steps
    mdb.models['InclusionModel'].StaticStep(name='Deformation1', previous='Initial',
timePeriod=eps_eng/strainrate, initialInc=0.5, minInc=5e-06, maxInc=0.9, nlgeom=ON,
maxNumInc=1000)
    session.viewports['Viewport: 1'].assemblyDisplay.setValues(step='Deformation1')

    #create output
    mdb.models['InclusionModel'].fieldOutputRequests['F-Output-1'].setValues(
variables=('PE', 'S', 'U', 'COORD', 'EVOL'))

#define BC

```

```

a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Inclusion-1'].edges
edges1 = e1.getSequenceFromMask(mask=('[#1 ]', ), )
e2 = a.instances['Matrix-1'].edges
edges2 = e2.getSequenceFromMask(mask=('[#18 ]', ), )
region = regionToolset.Region(edges=edges1+edges2)
mdb.models['InclusionModel'].YsymmBC(name='Symmetry', createStepName='Initial',
    region=region, localCsys=None)
a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
edges1 = e1.getSequenceFromMask(mask=('[#40 ]', ), )
region = regionToolset.Region(edges=edges1)
#apply velocity on top of matrix
mdb.models['InclusionModel'].VelocityBC(name='Deformation',
    createStepName='Deformation1', region=region, v1=UNSET, v2=-h*strainrate,
    vr3=UNSET, amplitude=UNSET, localCsys=None, distributionType=UNIFORM,
    fieldName='')

a = mdb.models['InclusionModel'].rootAssembly
e1 = a.instances['Matrix-1'].edges
edges1 = e1.getSequenceFromMask(mask=('[#82 ]', ), )
e2 = a.instances['Inclusion-1'].edges
edges2 = e2.getSequenceFromMask(mask=('[#4 ]', ), )
region = regionToolset.Region(edges=edges1+edges2)
mdb.models['InclusionModel'].XsymmBC(name='rot_sym', createStepName='Initial',
    region=region, localCsys=None)

#create jobname
A_string=str(2*A)
A_string=A_string.replace('.',',')
B_string=str(2*B)
B_string=B_string.replace('.',',')
eps_string=str(eps)
eps_string=eps_string.replace('.',',')

jobname='Inclusion_A1,0_B1,0_CAX4RH_jc25_165_0,75_eps'+eps_string

#create job
mdb.Job(name=jobname, model='InclusionModel', description='',
    type=ANALYSIS, atTime=None, waitMinutes=0, waitHours=0, queue=None,
    memory=90, memoryUnits=PERCENTAGE, getMemoryFromAnalysis=True,
    explicitPrecision=SINGLE, nodalOutputPrecision=SINGLE, echoPrint=OFF,
    modelPrint=OFF, contactPrint=OFF, historyPrint=OFF, userSubroutine='',
    scratch='', multiprocessingMode=DEFAULT, numCpus=2, numDomains=2)

# Run the job
mdb.jobs[jobname].submit(consistencyChecking=OFF)

# Do not return control till job is finished running
mdb.jobs[jobname].waitForCompletion()

#new_odb.close()

# *****
# CLASS DEFINITION
# *****

class FieldOutputRotations:

    def __init__(self, element_label_list, rotations_tuple_list):
        self.field_data_element_labels = element_label_list
        self.field_data_z_rotations = rotations_tuple_list

class FieldOutputDisplacements:

    def __init__(self, node_label_list, displacement_tuple_list):
        self.field_data_node_labels = node_label_list
        self.field_data_xyz_displacements = displacement_tuple_list

class FieldOutputOrientations:

    def __init__(self, element_label_list, orientations_tuple_list):
        self.field_data_element_labels = element_label_list
        self.field_data_orientations = orientations_tuple_list

```

```

# *****

from odbAccess import *
from abaqusConstants import *
import visualization
import math

# *****
# READ EXISTING OUTPUT DATABASE (.ODB)
# *****
Inc_Odb_Path = jobname+'.odb'
IncOdb = session.openOdb(name=Inc_Odb_Path)

Mat_Instance = IncOdb.rootAssembly.instances['MATRIX-1']
Inc_Instance=IncOdb.rootAssembly.instances['INCLUSION-1']

#read node data matrix
Mat_node_labels_and_coords = []
# This will hold node labels and coordinates for the Matrix as
# [(label1, xcoord1, ycoord1, zcoord1), (label2, xcoord2, ycoord2, zcoord2), ...]

Mat_nodes=Mat_Instance.nodes
no_of_Mat_nodes = len(Mat_nodes)
for i in range(no_of_Mat_nodes):
    node_label=Mat_nodes[i].label
    node_x_coord = Mat_nodes[i].coordinates[0]
    node_y_coord = Mat_nodes[i].coordinates[1]
    node_z_coord = Mat_nodes[i].coordinates[2]

    Mat_node_labels_and_coords.append((node_label, node_x_coord, node_y_coord, node_z_coord))

#read node data inclusion
Inc_node_labels_and_coords = []

Inc_nodes=Inc_Instance.nodes
no_of_Inc_nodes = len(Inc_nodes)
for i in range(no_of_Inc_nodes):
    node_label=Inc_nodes[i].label
    node_x_coord = Inc_nodes[i].coordinates[0]
    node_y_coord = Inc_nodes[i].coordinates[1]
    node_z_coord = Inc_nodes[i].coordinates[2]

    Inc_node_labels_and_coords.append((node_label, node_x_coord, node_y_coord, node_z_coord))

#read element data matrix
Mat_element_labels_and_connects = []
# This contains element labels and connectivity of Matrix
# (label, conn1, conn2, conn3, conn4)
Mat_elements=Mat_Instance.elements
no_of_Mat_elements = len(Mat_elements)
for j in range(no_of_Mat_elements):
    element_label = Mat_Instance.elements[j].label

    element_connector_1 = Mat_elements[j].connectivity[0]
    element_connector_2 = Mat_elements[j].connectivity[1]
    element_connector_3 = Mat_elements[j].connectivity[2]
    element_connector_4 = Mat_elements[j].connectivity[3]

    Mat_element_labels_and_connects.append((element_label,
                                            element_connector_1,
                                            element_connector_2,
                                            element_connector_3,
                                            element_connector_4))

#read element data inclusion
Inc_element_labels_and_connects = []
Inc_elements=Inc_Instance.elements
no_of_Inc_elements = len(Inc_elements)
for j in range(no_of_Inc_elements):
    element_label = Inc_elements[j].label

    element_connector_1 = Inc_elements[j].connectivity[0]
    element_connector_2 = Inc_elements[j].connectivity[1]
    element_connector_3 = Inc_elements[j].connectivity[2]
    element_connector_4 = Inc_elements[j].connectivity[3]

    Inc_element_labels_and_connects.append((element_label,

```

```

        element_connector_1,
        element_connector_2,
        element_connector_3,
        element_connector_4))

# Get names of all element sets of instance Matrix
MatSetNames = Mat_Instance.elementSets.keys()
# there is only one at position = 0
MatElSet = Mat_Instance.elementSets[MatSetNames[0]]
MatElementSet=Mat_Instance.elementSets[MatElSet.name]

# Get names of all node sets of instance Matrix
MatSetNames = Mat_Instance.nodeSets.keys()
# there is only one at position = 0
MatNoSet = Mat_Instance.elementSets[MatSetNames[0]]
MatNodeRegion=Mat_Instance.nodeSets[MatNoSet.name]

#loop particle diameter dp; subgrainsize dsg; recovery r
for dp in range(5, 6):
    for dsg in range(2, 3):
        for r in range(1, 11):

            recovery=r/10.0
            recovery_string=str(recovery)
            recovery_string=recovery_string.replace('.',',')

            fieldoutput_element_labels_Deformation1=[]
            fieldoutput_element_rot_Deformation1=[]
            fieldoutput_element_orient_Deformation1=[]
            no_of_frames = len(IncOdb.steps['Deformation1'].frames)
            MatRot_Deformation1 = []
            MatOrient_Deformation1 = []

            #store data; 0=initial, -1=last frame

MatCoords_init=IncOdb.steps['Deformation1'].frames[0].fieldOutputs['COORD'].getSubset(region=MatNodeRegion).values
MatCoords_frames=IncOdb.steps['Deformation1'].frames
PEEQvalues=IncOdb.steps['Deformation1'].frames[-1].fieldOutputs['PE'].getSubset(region=MatElementSet).values
MatCoords=MatCoords_frames[-1].fieldOutputs['COORD'].getSubset(region=MatNodeRegion).values
EVOLvalues=IncOdb.steps['Deformation1'].frames[-1].fieldOutputs['EVOL'].getSubset(region=MatElementSet).values

#set variables for 4 subgrain model
alpha_times_areal=0
areal=0
alpha_SG1=0
num_SG1=0
SG_e11=[]

alpha_times_area2=0
area2=0
alpha_SG2=0
num_SG2=0
SG_e12=[]

alpha_times_area3=0
area3=0
alpha_SG3=0
num_SG3=0
SG_e13=[]

alpha_times_area4=0
area4=0
alpha_SG4=0
num_SG4=0
SG_e14=[]

#set variables for 1 subgrain model
## alpha_times_area=0
## area=0
## alpha_SG=0
## num_SG=0
## SG_e1=[]

```

```

#loop every element to compute orientation
for iValue in PEEQvalues:
    element_label = iValue.elementLabel
    LocalSystem=iValue.localCoordSystem[0]
    x=LocalSystem[0]
    y=LocalSystem[1]
    if y<0:
        if x>0:
            alpha=-math.asin(math.fabs(y))
        else:
            alpha=-math.asin(math.fabs(x))-pi/2

    else:
        if x>0:
            alpha=math.asin(math.fabs(y))
        else:
            alpha=-math.asin(math.fabs(y))-pi
#get element nodes to compute coordinates
myNodes = Mat_elements[element_label-1].connectivity
El_x=0
El_y=0
for j in range(4):
    x=MatCoords[myNodes[j]-1].data[0]
    y=MatCoords[myNodes[j]-1].data[1]

    El_x+=x
    El_y+=y

El_x/=4
El_y/=4
El_r=sqrt(pow(El_x,2)+pow(El_y,2))

#define subgrain shape
r1=max(A,B)
t=(dsg/(dp*1.0))*0.5
r2=r1+t
At=A+t
Bt=B+t
phi_SG=(2*t)/(0.5*(2*r1+t))

phi_m=(pi/4)+(pi/4-phi_SG/2)*(pow(B,2)-pow(A,2))/(max(pow(B,2),pow(A,2)))
phi_m2=phi_m+phi_SG
phi_m3=phi_m-phi_SG

#define subgrain for 4 subgrain model
if El_r<r2+t:
    if El_r>r2:
        if El_y<El_x*tan(phi_m+phi_SG/2):
            if El_y>El_x*tan(phi_m-phi_SG/2):
                num_SG4+=1
                alpha_times_area4+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))
                area4+=EVOLvalues[element_label-1].data/(2*El_x*pi)
                SG_el4.append(element_label)

        if El_r<r2:
            if El_y<El_x*tan(phi_m+phi_SG/2):
                if El_y>El_x*tan(phi_m-phi_SG/2):
                    num_SG1+=1
                    alpha_times_area1+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))
                    area1+=EVOLvalues[element_label-1].data/(2*El_x*pi)
                    SG_el1.append(element_label)

            if El_x<El_y/tan(phi_m2-phi_SG/2):
                if El_x>El_y/tan(phi_m2+phi_SG/2):
                    num_SG2+=1
                    alpha_times_area2+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))
                    area2+=EVOLvalues[element_label-1].data/(2*El_x*pi)
                    SG_el2.append(element_label)

            if El_y<El_x*tan(phi_m-phi_SG/2):
                if El_y>El_x*tan(phi_m3-phi_SG/2):
                    num_SG3+=1
                    alpha_times_area3+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))

```



```

        area3+=EVOLvalues[element_label-1].data/(2*El_x*pi)
        SG_el3.append(element_label)

# define subgrain for 1 subgrain model
##
## rp=(dsg/(dp*1.0))*0.5
## xp=A*cos(phi_m)
## yp=A*sin(phi_m)
## if El_y-yp>0:
##     if pow(El_x-xp,2)<pow(rp,2):
##         if El_y < sqrt(pow(rp,2)-pow(El_x-xp,2))+yp:
##             num_SG+=1
##             alpha_times_area+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))
##             area+=EVOLvalues[element_label-1].data/(2*El_x*pi)
##             SG_el.append(element_label)
##         else:
##             if pow(El_x-xp,2)<pow(rp,2):
##                 if El_y > -sqrt(pow(rp,2)-pow(El_x-xp,2))+yp:
##                     num_SG+=1
##                     alpha_times_area+=alpha*57.3*(EVOLvalues[element_label-
1].data/(2*El_x*pi))
##                     area+=EVOLvalues[element_label-1].data/(2*El_x*pi)
##                     SG_el.append(element_label)
##             else:
##                 area+=EVOLvalues[element_label-1].data/(2*El_x*pi)
##                 SG_el.append(element_label)

# write initial orientation
fieldoutput_element_labels_Deformation1.append(element_label)
fieldoutput_element_rot_Deformation1.append((alpha*57.3,))

max=max1=max2=max3=max4=0
min=min1=min2=min3=min4=90
# overwrite orientation for the model with 4 subgrains
if areal>0:
    alpha_SG1=alpha_times_area1/areal
    for el_label in SG_el1:
        div=alpha_SG1-fieldoutput_element_rot_Deformation1[el_label-1][0]
        fieldoutput_element_rot_Deformation1[el_label-
1]=fieldoutput_element_rot_Deformation1[el_label-1][0]+recovery*div,
        if fieldoutput_element_rot_Deformation1[el_label-1][0]>max1:
            max1=fieldoutput_element_rot_Deformation1[el_label-1][0]
        elif fieldoutput_element_rot_Deformation1[el_label-1][0]<min1:
            min1=fieldoutput_element_rot_Deformation1[el_label-1][0]
    else:
        print 'areal=',areal
        min1=0
if area2>0:
    alpha_SG2=alpha_times_area2/area2
    if 2*num_SG2<num_SG1:
        alpha_SG2=0
    for el_label in SG_el2:
        div=alpha_SG2-fieldoutput_element_rot_Deformation1[el_label-1][0]
        fieldoutput_element_rot_Deformation1[el_label-
1]=fieldoutput_element_rot_Deformation1[el_label-1][0]+recovery*div,
        if fieldoutput_element_rot_Deformation1[el_label-1][0]>max2:
            max2=fieldoutput_element_rot_Deformation1[el_label-1][0]
        elif fieldoutput_element_rot_Deformation1[el_label-1][0]<min2:
            min2=fieldoutput_element_rot_Deformation1[el_label-1][0]
    else:
        print 'area2=',area2
        min2=0
if area3>0:
    alpha_SG3=alpha_times_area3/area3
    if 2*num_SG3<num_SG1:
        alpha_SG3=0
    for el_label in SG_el3:
        div=alpha_SG3-fieldoutput_element_rot_Deformation1[el_label-1][0]
        fieldoutput_element_rot_Deformation1[el_label-
1]=fieldoutput_element_rot_Deformation1[el_label-1][0]+recovery*div,
        if fieldoutput_element_rot_Deformation1[el_label-1][0]>max3:
            max3=fieldoutput_element_rot_Deformation1[el_label-1][0]
        elif fieldoutput_element_rot_Deformation1[el_label-1][0]<min3:
            min3=fieldoutput_element_rot_Deformation1[el_label-1][0]
    else:
        print 'area3=',area3
        min3=0
if area4>0:
    alpha_SG4=alpha_times_area4/area4
    for el_label in SG_el4:

```

```

        div=alpha_SG4-fieldoutput_element_rot_Deformation1[el_label-1][0]
        fieldoutput_element_rot_Deformation1[el_label-
1]=fieldoutput_element_rot_Deformation1[el_label-1][0]+recovery*div,
        if fieldoutput_element_rot_Deformation1[el_label-1][0]>max4:
            max4=fieldoutput_element_rot_Deformation1[el_label-1][0]
        elif fieldoutput_element_rot_Deformation1[el_label-1][0]<min4:
            min4=fieldoutput_element_rot_Deformation1[el_label-1][0]
    else:
        print 'area4=',area4
        min4=0

    #overwrite orientation for the model with 1 subgrain
    ##
    ##     if area>0:
    ##         alpha_SG=alpha_times_area/area
    ##         for el_label in SG_el:
    ##             div=alpha_SG-fieldoutput_element_rot_Deformation1[el_label-1][0]
    ##             fieldoutput_element_rot_Deformation1[el_label-
1]=fieldoutput_element_rot_Deformation1[el_label-1][0]+recovery*div,
    ##             if fieldoutput_element_rot_Deformation1[el_label-1][0]>max:
    ##                 max=fieldoutput_element_rot_Deformation1[el_label-1][0]
    ##             elif fieldoutput_element_rot_Deformation1[el_label-1][0]<min:
    ##                 min=fieldoutput_element_rot_Deformation1[el_label-1][0]
    ##
    ##         else:
    ##             print 'area=',area
    ##             min=0

    print 'Teilchdurchmesser', dp
    print 'Subkorndurchmesser', dsg
    print 'recovery', recovery
    print 'num_SG1', num_SG1
    print 'misorientation1', alpha_SG1
    print 'num_SG2', num_SG2
    print 'misorientation2', alpha_SG2
    print 'num_SG3', num_SG3
    print 'misorientation3', alpha_SG3
    print 'num_SG4', num_SG4
    print 'misorientation4', alpha_SG4
    ##
    ##     print 'num_SG', num_SG
    ##     print 'misorientation', alpha_SG

MatRot_Deformation1.append(FieldOutputRotations(fieldoutput_element_labels_Deformation1,
                                                fieldoutput_element_rot_Deformation1))

fieldoutput_element_labels_Deformation1 = []
fieldoutput_element_rot_Deformation1 = []

del MatCoords_init
del MatCoords_frames

fieldoutput_node_labels_Deformation1 = []
fieldoutput_node_disp_Deformation1 =[]

no_of_frames = len(IncOdb.steps['Deformation1'].frames)

MatDisp_Deformation1 = []
MatDisp_frames=IncOdb.steps['Deformation1'].frames

MatDisp=MatDisp_frames[-
1].fieldOutputs['U'].getSubset(region=MatNodeRegion).values
for q in range(no_of_Mat_nodes):
    field_label = Mat_nodes[q].label
    field_data_x = MatDisp[q].data[0]
    field_data_y = MatDisp[q].data[1]
    field_data_z = 0

    fieldoutput_node_labels_Deformation1.append(field_label)
    fieldoutput_node_disp_Deformation1.append((field_data_x,
field_data_y,field_data_z))

MatDisp_Deformation1 \
    .append(FieldOutputDisplacements(fieldoutput_node_labels_Deformation1,
                                    fieldoutput_node_disp_Deformation1))
fieldoutput_node_labels_Deformation1 = []
fieldoutput_node_disp_Deformation1 = []
del MatDisp_frames

```

```

del MatDisp

#####
#Inclusion
#####

# Get names of all element sets of instance Inclusion
IncSetNames = Inc_Instance.elementSets.keys()
# there is only one at position = 0
IncElSet = Inc_Instance.elementSets[IncSetNames[0]]
IncElementSet=Inc_Instance.elementSets[IncElSet.name]

# Get names of all node sets of instance Inclusion
IncSetNames = Inc_Instance.nodeSets.keys()
# there is only one at position = 0
IncNoSet = Inc_Instance.elementSets[IncSetNames[0]]
IncNodeRegion=Inc_Instance.nodeSets[IncNoSet.name]

fieldoutput_element_labels_Deformation1=[]
fieldoutput_element_rot_Deformation1=[]
fieldoutput_element_orient_Deformation1=[]
no_of_frames = len(IncOdb.steps['Deformation1'].frames)
IncRot_Deformation1 = []
IncOrient_Deformation1 = []

IncCoords_init=IncOdb.steps['Deformation1'].frames[0].fieldOutputs['COORD'].getSubset(region=IncNodeRegion).values
IncCoords_frames=IncOdb.steps['Deformation1'].frames

PEEQvalues=IncOdb.steps['Deformation1'].frames[-1].fieldOutputs['PE'].getSubset(region=IncElementSet).values

for iValue in PEEQvalues:
    element_label = iValue.elementLabel
    LocalSystem=iValue.localCoordSystem[0]
    y=LocalSystem[1]
    if y<0:
        alpha=-math.asin(math.fabs(y))
    else:
        alpha=math.asin(math.fabs(y))

    fieldoutput_element_labels_Deformation1.append(element_label)
    fieldoutput_element_rot_Deformation1.append((alpha*57.3,))

IncRot_Deformation1.append(FieldOutputRotations(fieldoutput_element_labels_Deformation1,
                                                fieldoutput_element_rot_Deformation1))

fieldoutput_element_labels_Deformation1 = []
fieldoutput_element_rot_Deformation1 = []
del IncCoords_init
del IncCoords_frames

fieldoutput_node_labels_Deformation1 = []
fieldoutput_node_disp_Deformation1 =[]

no_of_frames = len(IncOdb.steps['Deformation1'].frames)

IncDisp_Deformation1 = []

IncDisp=IncOdb.steps['Deformation1'].frames[-1].fieldOutputs['U'].getSubset(region=IncNodeRegion).values
for q in range(no_of_Inc_nodes):
    field_label = Inc_nodes[q].label
    field_data_x = IncDisp[q].data[0]
    field_data_y = IncDisp[q].data[1]
    field_data_z = 0

    fieldoutput_node_labels_Deformation1.append(field_label)
    fieldoutput_node_disp_Deformation1.append((field_data_x,
field_data_y,field_data_z))

IncDisp_Deformation1 \
    .append(FieldOutputDisplacements(fieldoutput_node_labels_Deformation1,

```

```

fieldoutput_node_labels_Deformation1 = []
fieldoutput_node_disp_Deformation1 = []

# Make a list of the step times for frames of 'Deformation1'
frame_step_times_Deformation1 = []
for x in range(len(MatRot_Deformation1)):
    frame_step_times_Deformation1.append(IncOdb.steps['Deformation1'].frames[x] \
        .frameValue)

# *****
# WRITE TO NEW OUTPUT DATABASE (.ODB)
# *****

#create a name for an output database
import os
dsg_string=str(dsg)
dsg_string=dsg_string.replace('.',',')
dp_string=str(dp)
dp_string=dp_string.replace('.',',')
output= 'Rotations_' + jobname+'_dp'+ dp_string+'_dsg'+
dsg_string+'_rec'+recovery_string

print output

if os.path.exists(output + '.odb'):
    os.remove(output + '.odb')

#create output database
new_odb = Odb('Element Rotations', analysisTitle='',
    description='', path= output + '.odb')
#create part
mat_part = new_odb.Part(name='Matrix Part', embeddedSpace=THREE_D,
    type=DEFORMABLE_BODY)
inc_part = new_odb.Part(name='Inclusion Part', embeddedSpace=THREE_D,
    type=DEFORMABLE_BODY)

#add nodes and elements
mat_part.addNodes(nodeData=Mat_node_labels_and_coords, nodeSetName='node set')
inc_part.addNodes(nodeData=Inc_node_labels_and_coords, nodeSetName='node set')

mat_part.addElements(elementData=Mat_element_labels_and_connects, type='CAX4RH',
    elementSetName='element set')
inc_part.addElements(elementData=Inc_element_labels_and_connects, type='CAX4RH',
    elementSetName='element set')

# Instance the part
mat_instance = new_odb.rootAssembly.Instance(name='Matrix Instance',
    object=mat_part)
inc_instance = new_odb.rootAssembly.Instance(name='Inclusion Instance',
    object=inc_part)

# Create a step 'Deformation1'
Deformation1 = new_odb.Step(name = 'Deformation1', description='', domain=TIME,
    timePeriod=1)

# Create frames with element rotations
for i in range(len(MatRot_Deformation1)):

    single_frame_Deformation1 = Deformation1.Frame(incrementNumber=i,
frameValue=frame_step_times_Deformation1[i],
        description='')

    rot_field_Deformation1 = single_frame_Deformation1.FieldOutput(name='ROT',
        description='Rotations',
        type=SCALAR)

    rot_field_Deformation1.addData(position=INTEGRATION_POINT,
instance=mat_instance,
        labels=MatRot_Deformation1[i].field_data_element_labels,
        data=MatRot_Deformation1[i].field_data_z_rotations)
    rot_field_Deformation1.addData(position=INTEGRATION_POINT,
instance=inc_instance,
        labels=IncRot_Deformation1[i].field_data_element_labels,
        data=IncRot_Deformation1[i].field_data_z_rotations)

```

```

disp_field_Deformation1 = single_frame_Deformation1.FieldOutput(name='U',
                                                                description='Displacements',
                                                                type=VECTOR)

disp_field_Deformation1.addData(position=NODAL, instance=mat_instance,
                                labels=MatDisp_Deformation1[i].field_data_node_labels,
                                data=MatDisp_Deformation1[i].field_data_xyz_displacements)
disp_field_Deformation1.addData(position=NODAL, instance=inc_instance,
                                labels=IncDisp_Deformation1[i].field_data_node_labels,
                                data=IncDisp_Deformation1[i].field_data_xyz_displacements)

# Make this the default field for visualization
Deformation1.setDefaultDeformedField(disp_field_Deformation1)

# Save the odb
new_odb.save()
new_odb.close()

#write to txt file
myString='eps ' + eps_string + " "
myLines=[]
myLines.append(myString)
myString='dp ' + dp_string + " "
myLines.append(myString)
myString='dsg ' + dsq_string + " "
myLines.append(myString)

myString='recovery ' + recovery_string + " "
myLines.append(myString)

alpha_SG1_string=str(alpha_SG1)
alpha_SG1_string=alpha_SG1_string.replace('.',',')
myString='Rot_SG1 ' + alpha_SG1_string + " "
myLines.append(myString)
min_SG1_string=str(min1)
min_SG1_string=min_SG1_string.replace('.',',')
myString='minRot_SG1 ' + min_SG1_string + " "
myLines.append(myString)
max_SG1_string=str(max1)
max_SG1_string=max_SG1_string.replace('.',',')
myString='maxRot_SG1 ' + max_SG1_string + " "
myLines.append(myString)

alpha_SG2_string=str(alpha_SG2)
alpha_SG2_string=alpha_SG2_string.replace('.',',')
myString='Rot_SG2 ' + alpha_SG2_string + " "
myLines.append(myString)
min_SG2_string=str(min2)
min_SG2_string=min_SG2_string.replace('.',',')
myString='minRot_SG2 ' + min_SG2_string + " "
myLines.append(myString)
max_SG2_string=str(max2)
max_SG2_string=max_SG2_string.replace('.',',')
myString='maxRot_SG2 ' + max_SG2_string + " "
myLines.append(myString)

alpha_SG3_string=str(alpha_SG3)
alpha_SG3_string=alpha_SG3_string.replace('.',',')
myString='Rot_SG3 ' + alpha_SG3_string + " "
myLines.append(myString)
min_SG3_string=str(min3)
min_SG3_string=min_SG3_string.replace('.',',')
myString='minRot_SG3 ' + min_SG3_string + " "
myLines.append(myString)
max_SG3_string=str(max3)
max_SG3_string=max_SG3_string.replace('.',',')
myString='maxRot_SG3 ' + max_SG3_string + " "
myLines.append(myString)

alpha_SG4_string=str(alpha_SG4)
alpha_SG4_string=alpha_SG4_string.replace('.',',')
myString='Rot_SG4 ' + alpha_SG4_string + " "
myLines.append(myString)
min_SG4_string=str(min4)
min_SG4_string=min_SG4_string.replace('.',',')
myString='minRot_SG4 ' + min_SG4_string + " "
myLines.append(myString)

```

```

max_SG4_string=str(max4)
max_SG4_string=max_SG4_string.replace('.',',')
myString='maxRot_SG4 ' + max_SG4_string + " "
myLines.append(myString)

##         alpha_SG_string=str(alpha_SG)
##         alpha_SG_string=alpha_SG_string.replace('.',',')
##         myString='Rot_SG ' + alpha_SG_string + " "
##         myLines.append(myString)
##         min_SG_string=str(min)
##         min_SG_string=min_SG_string.replace('.',',')
##         myString='minRot_SG ' + min_SG_string + " "
##         myLines.append(myString)
##         max_SG_string=str(max)
##         max_SG_string=max_SG_string.replace('.',',')
##         myString='maxRot_SG ' + max_SG_string + " "
##         myLines.append(myString)

myElFile.writelines(myLines)
myElFile.write("\n")
#myElFile.write("\n\n")
    session_name='C:/Users/DA/Documents/DA Loidolt/Thesis Simulation/4 Subgrains partial
recovered/' + jobname + '.odb'
    session.odbs[session_name].close()
myElFile.close()

```

## Literatur

- [1] Günter Gottstein. *Physikalische Eigenschaften von Materialien*. Springer Verlag, Berlin, 3<sup>rd</sup> edition, 2007.
- [2] Jrmantei. Versetzung im 2D-Kristall. Website, 2010-06-12. Available online at [http://commons.wikimedia.org/wiki/File:Versetzung\\_im\\_2D-Kristall.svg](http://commons.wikimedia.org/wiki/File:Versetzung_im_2D-Kristall.svg), visited on November 5th 2013.
- [3] F.J. Humphreys and M. Hatherly. *Recrystallization and Related Annealing Phenomena*. Elsevier, Oxford, 2<sup>nd</sup> edition, 2004.
- [4] Y. Huang, F.J. Humphreys, and M. Ferry. The annealing behaviour of deformed cube-oriented aluminium single crystal. *Acta Materialia*, 48:2543–2556, 2000.
- [5] Q. Liu, Z. Yao, A. Godfrey, and W. Liu. Effect of particles on microstructural evolution during cold rolling of the aluminum alloy AA3104. *Journal of Alloys and Compounds*, 482:264–271, 2009.
- [6] P. Loidolt. Cold Deformation and Recrystallization of the Aluminium Alloy 6082. Master’s thesis, Graz University of Technology, 2014.
- [7] O. Dmitrieva, J.V. Svirina, E. Demir, and D. Raabe. Investigation of the internal substructure of microbands in a deformed copper single crystal: experiments and dislocation dynamics simulation. *Modelling Simul. Mater. Sci. Eng.*, 18:085011 14 pp., 2010.
- [8] W. Mitter and C. Sommitsch. *On the Theory of Microstructure Changes in Metals: Hardening, Recovery, Grain Boundary Motion, and Recrystallization*, volume 1a of *Monographic Series TU Graz Advanced Materials Science*. Verlag der Technischen Universität Graz, Graz, 2011.
- [9] O. Engler, H.E. Vatne, and E. Nes. The roles of oriented nucleation and oriented growth on recrystallisation textures in commercial purity aluminium. *Materials Science and Engineering*, A205:187–198, 1995.
- [10] Dassault Systèmes Simulia Corp., Providence. *ABAQUS 6.11 Documentation*, 2011.
- [11] J.F Nye. Some geometrical relations in dislocated crystals. *Acta Metallurgica*, 1:153–162, 1953.
- [12] Gautam Puri. *Python Scripts for Abaqus*. S.N., 2011. ISBN: 978-0-615-52050-6.
- [13] W. Mitter C. Sommitsch. On modelling of dynamic recrystallisation of fcc materials with low stacking fault energy. *Acta Materialia*, 54:357–375, 2006.
- [14] Erik Nes. Modelling of work hardening and stress saturation in fcc metals. *Progress in Materials Science*, 41:129–193, 1998.

- [15] F. Roters. Advanced Material Models for the Crystal Plasticity Finite Element Method, 2011. Habilitationsschrift, RWTH Aachen.
- [16] A. Ma, F. Roters, and D. Raabe. A dislocation density based constitutive model for crystal plasticity FEM including geometrical necessary dislocations. *Acta Materialia*, 54:2169–2179, 2006.
- [17] F. Roters, P. Eisenlohr, L. Hantcherli, D.D. Tjahjanto, T.R. Bieler, and D. Raabe. Overview of constitutive laws, kinematics, homogenization and multiscale methods in crystal plasticity finite-element modeling: Theory, experiments, applications. *Acta Materialia*, 58:1152–1211, 2009.
- [18] D. Hull and D.J. Bacon. *Introduction to Dislocations*. Butterworth-Heinemann, Oxford, 4<sup>th</sup> edition, 2001.
- [19] U. Holzwarth, A. Barbieri, S. Hansen-Ilzhöfer, P. Schaaff, and M. Haaks. Positron annihilation studies on the migration of deformation induced vacancies in stainless steel AISI 316 L. *Applied Physics A*, 73:467–475, 2000.
- [20] G. Saada. Interaction de dislocations érouissage et production de défauts ponctuels dans les métaux c.f.c. *Acta Metallurgica*, 9:166–168, 1961.
- [21] E. Schafler, G. Steiner, E. Kornikova, M. Kerber, and M.J. Zehetbauer. Lattice defect investigation of ECAP-Cu by means of X-ray line profile analysis, calorimetry and electrical resistometry. *Materials Science and Engineering A*, 410-411:169–173, 2005.