ON THE ROLE OF LATTICE DEFECTS INTERACTIONS ON STRAIN HARDENING: A STUDY FROM DISCRETE DISLOCATION DYNAMICS TO CRYSTAL PLASTICITY MODELLING

A Thesis
Presented to
The Academic Faculty

by

Nicolas Bertin

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Mechanical Engineering

Georgia Institute of Technology
December 2015

ON THE ROLE OF LATTICE DEFECTS INTERACTIONS ON STRAIN HARDENING: A STUDY FROM DISCRETE DISLOCATION DYNAMICS TO CRYSTAL PLASTICITY MODELLING

Approved by:

Professor Laurent Capolungo, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Professor Hamid Garmestani
School of Materials Science and Engineering
Georgia Institute of Technology

Professor David McDowell
School of Mechanical Engineering & Materials Science and Engineering
Georgia Institute of Technology

Doctor Carlos Tomé
Materials Science and Technology Division
Los Alamos National Laboratory

Professor Surya Kalidindi
School of Mechanical Engineering & Materials Science and Engineering
Georgia Institute of Technology

Date Approved: October 16, 2015

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Laurent Capolungo, for his enthusiasm, his insightful guidance and his constant support during the last four years. Besides, I would like to thank Dr. David McDowell, Dr. Surya Kalidindi, and Dr. Hamid Garmestani, who have kindly accepted to serve as members of my dissertation committee, and who have had the formidable task of examining the present manuscript. My special thanks go to Dr. Carlos Tomé, for his warm welcome during my stay at Los Alamos, and for the enriching discussions and interactions we have had. Having such amazing and accomplished scientists in my committee was a great honor.

I would also like to thank Dr. Irene Beyerlein and Dr. Matthew Barnett for their precious advices and the fruitful discussions we have had during our collaborations, and Dr. Cédric Pradalier, for whom programming and algorithms hold no secrets.

I thank my friendly and talented labmates at GeorgiaTech Lorraine, for their stimulating discussions and all the fun we have had during the last years. I wish them great success in their future life and career.

And last but not least, I would like to thank my family for their constant and unfailing support during this journey. To my parents, who have always encouraged me to do my best. To my lovely wife, Céline, who had to support me during the last two years of this work, and who had probably heard the word *dislocation* too much.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This thesis focuses on the effects of slip-slip, slip-twin, and slip-precipitates interactions on strain hardening, with the intent of developing comprehensive modelling capabilities enabling to investigate unit processes and their collective effects up to the macroscopic response.

To this end, the modelling strategy adopted in this work relies on a two-way exchange of information between predictions obtained by discrete dislocation dynamics (DDD) simulations and crystal plasticity laws informed by DDD.

At the scale of lattice defects, a DDD tool enabling simulations on any crystalline structure is developed to model dislocation-dislocation, dislocation-twin and dislocation-particles interactions. The tool is first used to quantify the collective effect and strength of dislocation-dislocation interactions on latent-hardening, especially in the case of pure Mg. With regards to slip-twin interactions, an atomistically-informed transmission mechanism is implemented in the DDD framework so as to investigate the collective effects of dislocation transmission across a twin-boundary. With respect to slip-particles interactions, an efficient novel DDD approach based on a Fast Fourier Transform (FFT) technique is developed to include the field fluctuations related to elastic heterogeneities giving rise to image forces on dislocation lines. In addition, the DDD-FFT approach allows for the efficient treatment of anisotropic elasticity, thereby paving the way towards performing DDD simulations in low-symmetry polycrystals.

The information extracted from the collective dislocation interactions are then passed to a series of constitutive models, and later used to quantify their effects at the scale of the polycrystal. For such purpose, a constitutive framework capable

of receiving information from lower scales and establishing a direct connection with DDD simulations is notably developed.

CHAPTER I

INTRODUCTION

1.1 Motivation

Worldwide, the development of new material systems with advanced mechanical properties (yield strength, ductility, etc.) has become one of the top research priorities in response to environmental and economic challenges, as illustrated by the recent launch of the Materials Genome Initiative. In metals, present strategies to obtain materials with superior mechanical properties rely on taking action on several structural factors such as alloying elements, stacking-fault, grain sizes, etc. As a result, the complexity of new microstructures increases, which in turn leads to more complex behaviors, whereby multiple dissipative processes can be activated simultaneously. Dissipation is mediated by the different lattice defects present within the crystalline structure, such as point defects, line defects – also referred to as dislocations –, and interfaces such as grain boundaries.

Clearly, the mechanical behavior of metals (yield strength, strain hardening, strain-rate sensitivity, ...) directly depends on the activation of deformation mechanisms – whose selection is dictated by both intrinsic material properties such as crystallography and stacking-fault energies (SFE) [211, 105] and external conditions including strain rate and temperature [126, 127] – that induce microstructural changes resulting from nucleation, transport and interactions of defects. The latter for instance include the key mechanisms of dislocation-dislocation interactions, dislocation-interfaces interactions – of which dislocation-twin interactions are a particular case – and dislocation-precipitate interactions, for which TEM images are reported in figure 1.1, and whose role and effects on the mechanical behavior depend on the crystalline

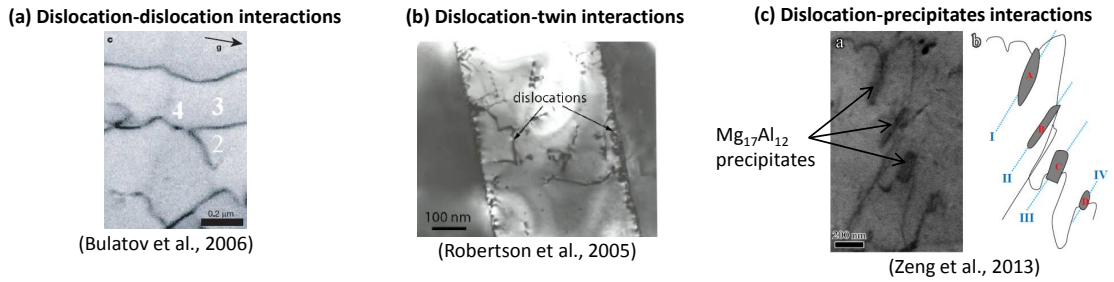


Figure 1.1: Examples of the types of defects interactions governing the microstructure evolution in metals. TEM micrographs of (b) dislocation-dislocation inetractions (reproduced from [31]), (c) dislocation-twin interactions (reproduced from [204]), (d) dislocation-particles interactions (reproduced from [268]).

structure and deformation regime (e.g. dislocation-dislocation interactions are dominating in slip-driven plasticity).

The case of the mechanical behavior of HCP crystals illustrates well the intricacy of the deformation process in metals. In figure 1.2(a), the spectrum of stress-strain responses as a function of the loading orientation in plane-stress compression of single crystal magnesium [121] is striking. Specifically, the rate of strain-hardening observed in orientations E and F between 6% and 8% strain significantly differs from that obtained in all other orientations and results from the simultaneous activation of slip and twinning deformation modes. Thus, although twin boundaries have been identified as important obstacles to dislocation motion in this process [213, 160, 222], a comprehensive understanding of their role and effects is still to be obtained [159, 158, 160].

As yet another example, the introduction of second-phase particles within crystal lattices has a direct impact on the mechanical response of metals. As illustrated in figure 1.2(b), while the presence of precipitates induces larger strain-hardening rates, the latter further differ depending on the alloying composition. Thus, although precipitation hardening has been extensively studied and the interactions between dislocations and precipitates have been evidenced as responsible for the improved strengthening [92, 15, 183], current approaches are not yet fully suited to predict

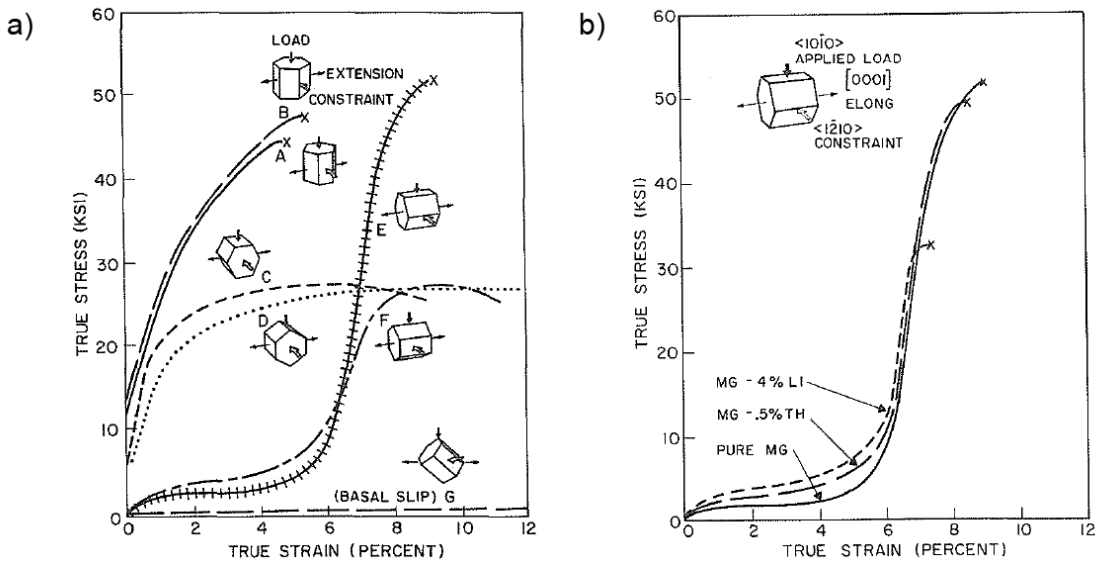


Figure 1.2: (a) Stress-strain responses in pure magnesium single crystals under plane-strain compression along different crystallographic directions at room temperature. (b) Stress-strain responses in the E orientation for different alloying compositions. Reproduced from the work of Kelley and Hosford [121].

the distinct effects induced by different particle compositions, whose assessment still largely relies on experiments [183, 185].

More generally, the increase in the complexity of the microstructures necessitates the use and development of complex modelling capabilities in order to interpret experimental data allowing to guide the development of new material systems.

1.2 Objectives and challenges

1.2.1 Multi-scale modelling approach

In the past decades, the development and refinement of constitutive laws to account for the increasing complexities of the microstructures has primarily relied on introducing sophistications in the internal state variables of the models. While such phenomenological approaches have in many cases demonstrated their ability to reproduce experimental results (e.g. [194, 33, 7, 162, 25, 123]), their predictive capabilities remain nonetheless limited, as attested by the persistent difficulties to predict mechanical responses in complex loading conditions [200, 122, 95]. Specifically, these

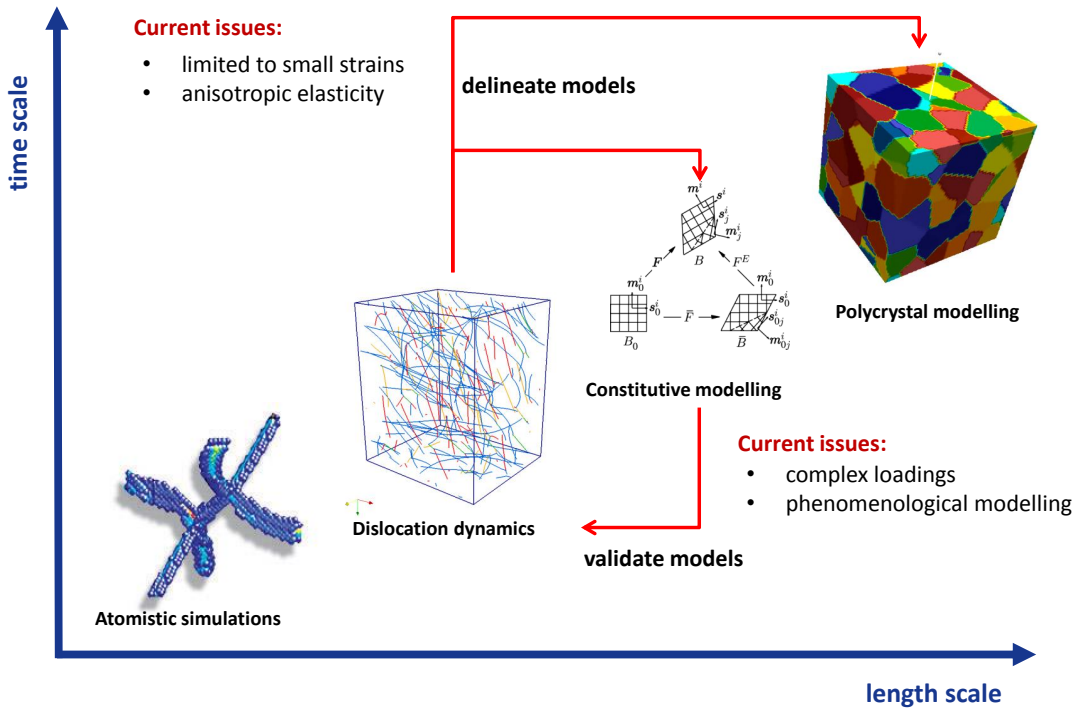


Figure 1.3: Diagram of the classical multiscale modelling approach in terms of time and length scales capabilities.

limitations are primarily associated with the difficulty in delineating approaches to establish connections between the evolution of the internal state variables and the activation and interactions of the different deformation modes driving microstructure evolution.

In this context, the core idea to circumvent the above limitations is to resort to a high hierarchical multiscale modelling approach, such as illustrated in figure 1.3. This diagram does not aim at representing all existing methods, but provides an overview of the methods used in the context of this work. Thus, at the lowest scale, atomistic simulations in which individual atoms are considered are used to investigate mechanisms associated with unit processes. Among them, Molecular Dynamics (MD) allow for simulating the motion of atoms whose interactions are modelled according to inter-atomic potentials. However, although computational capacities grow at a nearly exponential rate, modelling the deformation of a macroscopic sample of metal

with MD still remains an unrealistic objective at the present time. In other words, computational limitations impose restrictions on the domain of applicability of each approach in terms of time and length scales. Therefore, the modelling strategy consists in delineating homogenization techniques to establish a continuous connection between the different modelling scales as a way to exchange information. As a result, outcomes of atomistic simulations are typically used to inform dislocation dynamics simulations in which solely dislocation lines are represented. Following the same approach, discrete dislocations dynamics (DDD) simulation aiming at simulating the individual and/or collective behavior of dislocation lines provides a powerful tool to extract relevant microstructural parameters that can be incorporated into constitutive laws in which ensemble of dislocations are represented by means of densities. Delineating methods to achieve the exchange of information aforementioned is precisely the focus of this work.

Thus, crystal plasticity constitutive models informed by DDD have been developed at a higher scale [54, 7, 44, 161, 25]. A successful example of scale transitioning between DDD and constitutive models notably include the incorporation of the collective effect of dislocation-dislocation interactions by way of latent hardening coefficients $a^{ss'}$ that have been quantified using DDD for FCC [164, 165, 65] and BCC [198] materials. These latent hardening coefficients – related to the statistically representative effect of the strength of pairwise dislocations interactions between systems s and s' – have become a key ingredient of current dislocation density based constitutive models [112, 65, 132], and are typically accounted for in crystal plasticity frameworks to define the critical resolved shear stress (CRSS) τ_c^s on slip system s via the use of the Franciosi and Zaoui slip-hardening law [90]:

$$\tau_c^s = \tau_0^s + \mu b \sqrt{\sum_{s'} a^{ss'} \rho^{s'}} \tag{1.1}$$

where τ_0^s and ρ^s denote the lattice friction and the total dislocation density on system s, and where b and μ are the magnitude of the Burgers vector and the elastic shear modulus of the material, respectively. The use of these types of approaches has for instance allowed to refine orientation-dependent hardening rate predictions in single and polycrystals [7, 44, 25] and have permitted to account for complex slip mode selection processes [165, 64].

1.2.2 Challenges

In the context of this thesis, the hierarchical modelling approach needs to address the questions of dislocation-dislocation, dislocation-twin and dislocation-particles interactions. In the following, the shortcomings and challenges associated with current DDD approaches and with the information pass from DDD to constitutive laws to treat these defects interactions are presented.

Low-symmetry crystals: As a result of the anisotropic properties of dislocations in low-symmetry crystals (e.g. friction stress [37], mobilities [102]) arising from the presence of several slip modes, considerably less DDD studies have been dedicated to HCP metals compared to cubic crystals. Particularly, except for ice single crystal [62], hardening coefficients have not been calculated for hexagonal materials, for which generic latent parameters are currently used in constitutive laws.

In addition, most of current dislocation density based constitutive models rely upon two different slip hardening formulations, namely that proposed by Franciosi and Zaoui (FZ) whose expression is given in equation (1.1) [90], and that suggested by Lavrentev and Pokhil (LP) [142]. While the FZ law was specifically developed for FCC materials, the LP law appears to have been introduced for a broader variety of crystal structures. However, the validity of both these laws has never been assessed for HCP materials, and their accuracies have never been compared.

Anisotropic elasticity: Complementarily to the above limitation, current DDD simulations are generally limited to resort to elastic isotropic approximations when performing simulations on low-symmetry crystals, thereby not accounting for the anisotropic effects on dislocations stress fields [257, 14]. This limitation arises from the fact that, in contrast with isotropic elasticity, no closed-form solutions are available for the stress fields induced by dislocations in anisotropic media. Although substantial efforts have been undertaken in the past years to develop new numerical techniques allowing for savings in computation time [264, 12, 13], the cost of anisotropic calculations is typically one order of magnitude greater than that of isotropic calculations, and remains conditioned by the amount of anisotropy and the desired level of accuracy.

Heterogeneous elasticity: The study of dislocation-slip and dislocation-precipitates interactions requires to finely account for the effects of heterogeneous elasticity giving rise to image forces on dislocation lines. However, only a few DDD frameworks are capable of treating heterogeneous materials, such that only very few studies have been performed to investigate slip-particles interactions (e.g. [171, 218, 235, 199]), and these suffer from important limitations in terms of spatial and time resolutions. Particularly, the high computational burden associated with the finite element method (FEM) required in current approaches to enable inhomogeneous elasticity in DDD has precluded any fine investigations on such materials thus far.

Interfacial dislocation transmission: From a modelling standpoint, the transmission mechanisms of dislocations across twin-boundaries have mainly been investigated via atomistic simulations to examine unit processes (e.g. [119, 273, 265]). At the DDD scale, simulations have primarily relied on incorporating a line tension model to account for transmission events [56, 272], in place of incorporating dedicated local dissociation rules, thereby not accounting for important effects, especially

regarding interfacial plasticity.

Constitutive modelling: To account for the complexities associated with the presence of several deformation modes and address the remaining difficulties associated with complex loading conditions such as strain-path changes, sophisticated dislocation density-based constitutive models have emerged in the past years [195, 25, 196, 124, 189, 29, 122]. However, in addition to the limited predictive capabilities of phenomenological models, the incorporation of additional mechanisms often comes at the price of an increase in the number of fitting parameters, which may paradoxically lead to a disconnection from the physics of the dislocation. For instance, a given stress-strain response can be identically reproduced by the same model with two different sets of parameters, while predicted internal state variables evolve in drastically distinct manners [22]. Furthermore, it has been reported that predicted dislocation densities often do not concur with experimental observations, and are generally underestimated by one order of magnitude [49]. In addition, refinements of current models by ways of further decomposing the total density in a crystal into several distinct populations inevitably lead to additional complications. Specifically, their short-range interactions, physically taking the form of junctions or annihilations, are yet to be accounted for in constitutive laws. Doing so requires the development of novel approaches for their quantification.

Therefore, the development of comprehensive constitutive models with enhanced predictive capabilities still constitutes a challenging objective in material science. It follows that to finely capture intricate hardening mechanisms associated with low-symmetry crystals or complex loading conditions, constitutive models necessarily have to rely more closely on the physics of the dislocations. As a result, it is expected that constitutive models could greatly benefit from being informed by outcomes and parameters extracted at lower scales.

1.2.3 Objectives

The main objective of this work is to develop numerical tools to investigate defects interactions and quantify their collective effects on strain hardening in single crystals where deformation is primarily driven by slip. For such purpose, the philosophy adopted relies on a scale transitioning approach in which existing results extracted from atomistic simulations are used to inform and develop DDD simulation tools aiming at investigating unit processes associated with dislocation-dislocation, dislocation-twin, and dislocation-particles interactions. Further, massive DDD simulations are performed to assess their collective effects so as to validate, calibrate, and develop constitutive models accounting for defects interactions in a statistically representative fashion. Consequently, the strategy adopted in this work to contribute to the scale transition includes three aspects: 1) extracting parameters and laws from DDD simulations that would be incorporated in higher scales models, 2) developing a new efficient DDD approach that paves the way towards performing DDD simulations in heterogeneous / poly crystals, and 3) developing new constitutive laws establishing direct connections with lower scales modelling approaches.

In this thesis, the applications and studies are performed on various metals of different structures depending on the availability of critical data (e.g. atomistic results), for validation purposes, or to address specific needs of the material science community.

1.3 Scope of the thesis

In this thesis, the literature review is done within each chapter. In order to address the challenges presented in the previous sections, the manuscript is organized in the following manner:

Chapter 2 is dedicated to the development of a robust and efficient DDD simulation tool incorporating an explicit treatment of dislocation junctions and capable

of treating any kind of crystallography. For such purpose, the fundamental physical properties of dislocations are first reviewed and their continuum based representation is presented. Then, state-of-art numerical techniques are discussed and implemented. The code developed will then be used in Chapter 4 to compute latent hardening parameters in magnesium, and will serve as a framework for the novel FFT-based approach introduced in Chapter 3 and employed in Section 5.2, the transmission mechanism scheme implemented in Section 5.1, and the novel constitutive Hybrid Model developed in Chapter 6.

In Chapter 3, a novel formulation for DDD simulations based on an eigenstrain formalism and on fast Fourier transforms is developed for homogeneous and heterogeneous periodic microstructures. The objective of the DDD-FFT approach is to address current limitations of DDD simulations by enabling efficient and accurate calculations in anisotropic and heterogeneous elasticity, thereby paving the way towards performing DDD simulations in polycrystals.

Chapter 4 is dedicated to the quantification of dislocation-dislocation interactions parameters used in constitutive laws in the case of HCP magnesium, using the DDD simulation tool developed in Chapter 2. To this end, the crystallography of dislocation interactions in Mg is first analyzed. Then, latent hardening parameters are extracted and their effects on the macroscopic response predicted by polycrystalline models are investigated. Further, the validity of two widely used slip-hardening laws is assessed for HCP crystals.

Chapter 5 focuses on the investigation of dislocation interactions with different types of phases. Thus, the interaction between dislocations and coherent twin-boundaries is first examined by developing a transmission scheme to allow for the treatment of dislocation reactions upon intersection with coherent interfaces in DDD simulations. With this, the role of dislocation-twin interactions is investigated by focusing on dislocation multiplication and generation of internal strains. Then, the

heterogeneous DDD-FFT approach is employed to study dislocation-second phase particles in HCP Mg.

In Chapter 6, a novel strain hardening constitutive model for slip-driven plasticity is proposed. To address the limitations of current constitutive laws, the model distinguishes between glissile and stored dislocations, and establishes a direct connection with DDD simulations to quantify the evolution of dislocation densities due to glide and the transformation between the different populations by ways of interactions. The model is applied to FCC Al to demonstrate its capabilities.

Finally, conclusions and perspectives of this thesis are discussed in Chapter 7.

1.4 Notations

1.4.1 Mathematical notations

In this thesis, vectors are represented with an overhead arrow (e.g. $\vec{\boldsymbol{x}}$) and tensorial quantities are represented using a bold symbol (e.g. $\boldsymbol{X}$).

Both of these quantities can interchangeably be expressed in their Cartesian components form (e.g. x_i for the i-th component of a vector, X_{ij} for the component of a second-order tensor, Y_{ijkl} for the component of a fourth-order tensor, etc...).

Spatial derivatives in the component form are denoted using a comma followed by the component index (e.g. $x_{i,j}$ for the spatial derivative of the i-th component of vector $\vec{\boldsymbol{x}}$ with respect to the j-th direction).

Unless stated otherwise, Einstein summation convention is implicitly meant in the component form, so as to carry summations on repeated indices. For instance, one shall read the following equivalence:

$$A_{ij} = C_{ijkl}B_{kl} = A_{ij}\sum_{k}\sum_{l} C_{ijkl}B_{kl} \tag{1.2}$$

1.4.2 General variables

In the following, the notation of the general variables is presented. These notations are also introduced at the location of their first appearance in the text.

$\boldsymbol{I}_2$, δ_{ij} – Kronecker delta function

e_{ijk} – Levi-Civita permutation symbol

$\vec{\boldsymbol{u}}$, u_i – Displacement vector

$\boldsymbol{\epsilon}$, ϵ_{ij} – Total strain tensor

$\boldsymbol{\epsilon}^e$, ϵ^e_{ij} – Elastic strain tensor

$\boldsymbol{\epsilon}^p$, ϵ^p_{ij} – Plastic strain tensor

$\boldsymbol{\sigma}$, σ_{ij} – Cauchy stress tensor

$\boldsymbol{C}$, C_{ijkl} – 4th order stiffness tensor

$\boldsymbol{G}$, G_{ij} – Green's function

$\boldsymbol{\Gamma}$, Γ_{ijkl} – Modified Green's function

$\vec{\boldsymbol{x}}$, x_i – Coordinate in the real space

$\vec{\boldsymbol{\xi}}$, ξ_i – Coordinate in the Fourier space

ρ^s – Dislocation density on system s

τ^s – Resolved shear stress on system s

τ^s_c – Critical resolved shear stress on system s

$\dot{\gamma}^s$ – Shear strain rate on system s

V – Volume of a body

λ, μ – Lamé constants

CHAPTER II

DISCRETE DISLOCATION DYNAMICS

2.1 Introduction

At small scales, lattice defects and their interactions have long been determined as responsible for the plastic activity in metals [52, 136, 93]. Among these defects, linear defects – also referred to as dislocations – mediate the plastic slip through their motion and govern strain hardening through their interactions. The study of these defects and their role in the deformation process and in strain hardening behaviors has required over the years the development of dedicated numerical tools and techniques. These numerical tools have been particularly useful in providing insightful understandings of several mechanisms and, in specific cases, have allowed to go beyond what experimental capabilities can offer. Thus, to capture the physical mechanical behavior of slip-mediated plasticity, three-dimensional dislocation dynamics simulations, aiming at simulating the motion and interactions of dislocation lines, have emerged in the 1990s [134, 61, 67, 266, 30].

Several dislocation dynamics approaches have been developed, among which two classes of simulations can be distinguished. The first class is composed of front tracking methods in which dislocations are discretized into segments and tracked individually, while the second class regroups all alternative methods in which dislocations are not explicitly discretized. For instance, the second class includes the phase field approach in which dislocations are represented by density functions [246, 128, 116], and the level set method in which dislocation lines are represented by the intersection of higher order parametrized surfaces [262, 244]. In contrast, simulations in which dislocation lines are discretized are referred to as Discrete Dislocation Dynamics (DDD)

simulations, and are the subject of this chapter.

Since its development, DDD has proven to be a powerful tool aiding at understanding the collective behavior of large ensembles of dislocations at the mesoscale [67, 266, 30, 97, 5], and have successfully been used to investigate plasticity and strain hardening behaviors and processes at the scale of the dislocation. Well known applications include the study of plasticity in thin films [191, 242, 252, 272, 84] and micropillars [212, 255, 271], the quantification of the strength and formation likelihood of dislocation junctions in materials with different crystal symmetries [164, 39, 260], the unravelling of the formation process of triple dislocation junctions [31], the quantification of latent-hardening effects in cubics and hexagonal systems [165, 65, 198, 62, 22], and the quantification of the effect on dislocation interaction with irradiation-induced damage on strength [8, 84, 221]. DDD has also been successful in quantifying dislocation mean-free paths [19, 63, 132], in understanding the role of cross-slip in dislocation patterning [167, 249], in relating slip activity to plastic anisotropy [250], in elucidating the role of non-Schmid effects in tension/compression asymmetry [247], in quantifying the importance of elastic anisotropy [101, 40] and inertial effects [248], and in studying the interactions between mobile dislocations and grain boundaries [163, 152].

However, an exhaustive survey on the literature reveals that most of DDD studies have been performed on cubic materials. This is because supplementary complexities arise when dealing with low-symmetry crystals. For instance, performing DDD simulations on low-symmetry crystals such as HCP Mg necessarily requires the proper treatment of the anisotropic properties of dislocations [16, 37, 176], including friction stresses, mobilities [102], and anisotropic stress fields [108, 40, 85], each associated with the different slip modes. Although open-source codes such as ParaDis [5] or microMegas [69] are available – and could theoretically be modified to perform simulations on low-symmetry crystals –, these are usually distributed as black-boxes – especially regarding numerical efficiency purposes –, and are therefore not readily

designed to be edited. As a result, the choice is made for this thesis to develop a complete DDD code, for which the use for specific applications and studies would be simplified. Specifically, the code developed should enable the treatment of any crystallography. To this end, current DDD codes are analyzed and state-of-the-art techniques are implemented into the DDD code presented in this chapter, so as to develop a robust, efficient and competitive numerical tool.

The chapter is organized as follows. First, the general procedure in DDD and the complete DDD cycle are presented in Sections 2.1.1 and 2.1.2. Then, a review on the theory of dislocations laying the ground for the physical model underlying DDD simulations is given in Section 2.2. Finally, the physical and numerical aspects of each component of the regular DDD simulation approach are fully presented through Sections 2.3 to 2.9.

2.1.1 General procedure in DDD

Conceptually, the general procedure in DDD simulations is to adopt a discretization scheme of the dislocation network and derive thermodynamically consistent sets of equations of motion for each segment in the dislocation ensemble. For instance, the first approaches proposed a segment-based discretization scheme in which the orientation of segments was constrained to fixed sets (e.g. pure edge and pure screw segments [67, 240], mixed characters [164, 68]). Later, the nodal approach was introduced such as to allow for a continuous description of line orientation, using linear [266] or spline approximations [96, 97]. Once dislocation lines are discretized, the DDD procedure includes two main aspects: (i) the quantification of dislocation motion and (ii) the treatment of intersections between dislocation segments. Due to the non-linear behavior of dislocation motion and the complex interaction processes, DDD simulations are performed through an iterative process in which a time discretization is introduced.

As a result, steps (i) and (ii) are performed at each time step of the simulation. Further, as will be described in coming sections, the determination of dislocation motion (i) requires the precise knowledge of the mechanical state (stress state) throughout the whole simulation volume, which needs to be evaluated at each time step, since it depends on the position (or on the motion) of dislocation lines. On the other hand, intersections between dislocation lines (ii) that pertain to core reactions are generally treated via topological local rules based on static or dissipative criteria. Once steps (i) and (ii) have been performed, a new dislocation configuration is determined and the simulation can advance to the next time step. However, DDD simulations are complex numerical simulations that require supplementary stages to be implemented at each time step. Therefore, the full numerical DDD cycle is presented below.

2.1.2 DDD cycle

Numerically, the development and implementation of a DDD simulation is a complex task that requires the development of several components. Each of the main stages composing the basic DDD cycle are presented in the diagram in figure 2.1, and will be fully detailed throughout the coming sections of the present chapter. Sequentially, the main tasks to be performed at each time increment of the DDD cycle are the following:

(a) As any discrete approach, the evaluation of dislocation motion requires the dislocation network to be discretized. This procedure must be dynamically performed at each time increment since the topology of the network evolves from step to step: as dislocations move, the length (and shape) of dislocation segments changes and the connectivity between segments is modified by ways of dislocation intersections. The dynamic discretization (a), also referred to as segment remeshing, is required to ensure that, at each time step, the length of each dislocation segment allows for a correct representation of dislocation

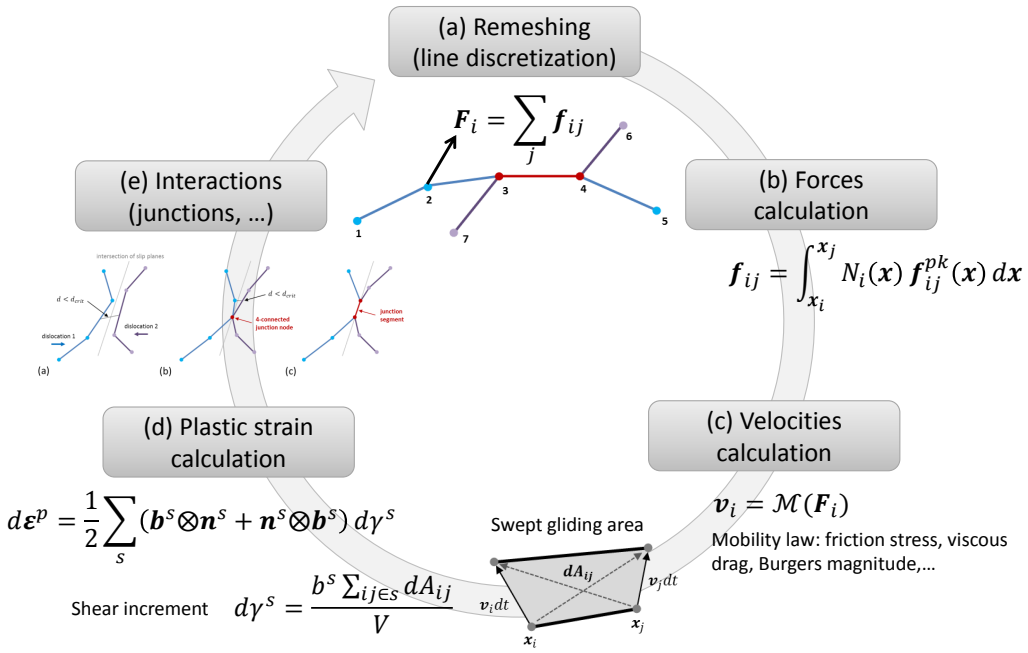


Figure 2.1: Diagram of the main stages composing the basic cycle performed at each time step in DDD simulations. (a) Dynamic discretization of dislocation lines: the topology of the dislocation network evolves at each time step requiring discretezation to be performed dynamically. (b) Forces calculation: the stress driving dislocation motion is evaluated from the spatial configuration of the dislocation network. (c) Velocities calculation: once forces on dislocations have been computed, the motion of dislocation lines can be calculated through the mobility law. (d) Plastic strain calculation: the areas swept by dislocation motion allow for the determination of the plastic activity. (e) Interactions: collisions between dislocations during glide are treated via topological rules. The DDD cycle (a) to (e) is repeated until simulation is completed.

lines and curvatures, and prevents the appearance of numerical instabilities. More details on the dislocation discretization schemes are provided in Section 2.3 while the different remeshing strategies are presented in Section 2.7.

(b) Once the dislocation network is well discretized, forces on dislocation segments resulting from the interactions of dislocations with the other dislocations and the applied loading are computed. Such forces correspond to the well-known Peach-Koehler forces [111] and are the driving forces for dislocation motion. The definition and coupling between the different contributions to the driving

force are presented in Section 2.4, while the calculation and assembly of nodal forces are detailed in Sections 2.5 and Appendix C.

(c) After nodal forces have been evaluated, nodal velocities are obtained from the use of the mobility law describing the gliding behavior of dislocation lines. This stage is explained in Section 2.5. Dislocation positions are then updated from dislocation velocities by virtue of time integration, thereby yielding a new dislocation configuration. Such procedure is described in Appendix C.4.

(d) Following the calculation of dislocation motion, the plastic activity is calculated by determining the areas swept by each dislocation segment, i.e. the sheared portions of the simulation volume in which slip has occur. This stage is required for the determination of the mechanical response of the simulated material and is described in Section 2.8.

(e) The last stage of a DDD simulation step pertains to dislocation core interactions. This includes the detection and treatment of dislocation collisions, leading to junction formations or annihilations, and the dissociation procedure. These procedures are presented in Section 2.6, while their numerical implementation is detailed in Appendix D.1.

Note that the order in which stages are performed may differ according to the different implementations. Thus, the interactions procedures (e) can be performed before the calculation of the plastic strain (d), for instance.

2.2 Theory of dislocations

In this section, the physical properties of dislocations and their effects are presented. Specifically, the continuum theory to derive stress fields associated with dislocation lines – whose evaluation is the core component of DDD simulations – is reviewed.

2.2.1 Concept

The concept of dislocation was first introduced by Volterra in 1907 [241]. A dislocation corresponds to a line defect within a crystal structure, associated with a displacement jump, and is characterized by its Burgers vector, usually denoted $\vec{\boldsymbol{b}}$, and its line direction $\vec{\boldsymbol{t}}$. The relative orientation between these two vectors defines the character of the dislocation, of which three types exist: edge, screw and mixed. An edge dislocation corresponds to a dislocation for which the Burgers vector is perpendicular to the line direction, while a screw dislocation is defined by the collinearity between its Burgers vector and its line direction. Therefore, edge and screw dislocations correspond to particular cases, and dislocations found in crystals are generally mixed dislocations, i.e. dislocations for which the Burgers vector form an angle $0° \leq \theta \leq 90°$ with its line direction.

To illustrate the concept of dislocation, an edge dislocation inserted in an initially perfect cubic crystal is depicted in figure 2.2. As drawn, an edge dislocation corresponds to the presence of an extra half plane of atoms, whose bottom termination delineates the line direction of the dislocation, usually represented with a $\perp$ symbol. In this case, the line direction is orthogonal to the paper (along the $\vec{\boldsymbol{z}}$ direction).

The definition of the Burgers vector associated with the dislocation depends on the sense of the line direction. Once a convention is adopted, the Burgers vector $\vec{\boldsymbol{b}}$, defining the magnitude and the direction of the displacement induced by the presence of the dislocation, is usually determined using the Burgers circuit procedure as introduced by Frank [91]. The procedure is as follows: a reference circuit $\mathcal{C}$ is delineated in the perfect crystal and is compared to the same circuit drawn in the dislocated crystal. The extra closure vector obtained in the case of the dislocated crystal is the Burgers vector. The existence of such closure vector arises from the elastic displacements of the atoms with respect to their perfect sites induced by the presence of the dislocation, as depicted in figure 2.2(c). Therefore, in an equivalent

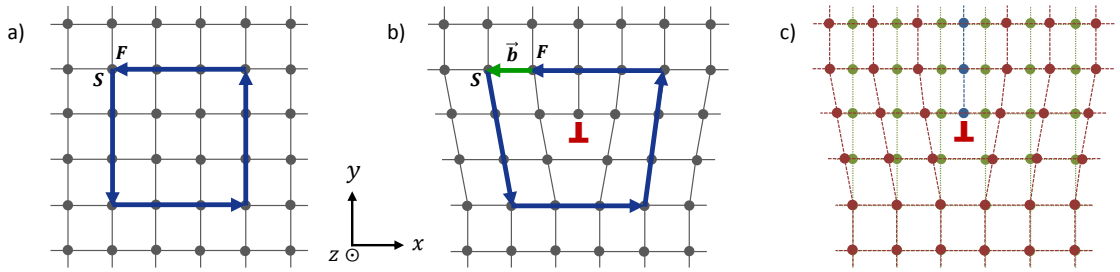


Figure 2.2: Schematic of an edge dislocation in a cubic crystal. (a) Perfect crystal: the reference Burgers circuit delineated in blue starts at position S and ends at the same position $F = S$. (b) Edge dislocation: when drawing the same circuit in the dislocated crystal, an extra vector $\vec{FS} = \vec{b}$ is required to close the circuit. Vector $\vec{b}$ is the Burgers vector that defines the direction and the magnitude of the displacement induced by the dislocation. The dislocation line located with a $\perp$ symbol is orthogonal to the paper (along the $\vec{z}$ direction). (c) Superposition of the atomic positions of the perfect (green) and dislocated (red) crystals highlighting the atom displacements induced by the presence of the dislocation. The extra plane of atoms characteristic of the edge dislocation is shown in blue.

manner, the Burgers vector is mathematically given as the line integral of the elastic displacements $\vec{u}$ along the Burgers circuit:

$$b_i = \oint_{\mathcal{C}} du_i = \oint_{\mathcal{C}} u_{i,j} dx_j \tag{2.1}$$

The integral in (2.1) vanishes when the crystal structure is not altered, or yields the Burgers vector when circuit $\mathcal{C}$ encircles a dislocation line. With this definition, it clearly appears that the presence of the extra plane, i.e. of the dislocation, induces a jump or a discontinuity in the displacement field, which cannot be well-defined everywhere in the crystal. Note that in order to obtain the true value of the Burgers vector, the circuit $\mathcal{C}$ should be taken along atoms that are sufficiently away from the dislocation line. This is to avoid dealing with the singularity of the displacements near the dislocation line, i.e. in the dislocation core. The dislocation core is more generally defined as the region where the elastic displacements of atoms are so large that the elastic approach is likely not to apply. The size of this region depends on

the lattice structure, and there is still substantial ongoing research to obtain a better understanding and description of dislocation cores [111, 16].

In contrast to the edge dislocation, the visualization of a screw dislocation is more difficult. As depicted in figure 2.3, a screw dislocation can be imagined as to introduce a cut along a plane in a perfect crystal along which one half of the crystal would have slipped by one atomic distance with respect to the other half. In this case, the Burgers vector $\vec{\boldsymbol{b}}$ required to close the reference Burgers circuit is parallel to the dislocation line L.

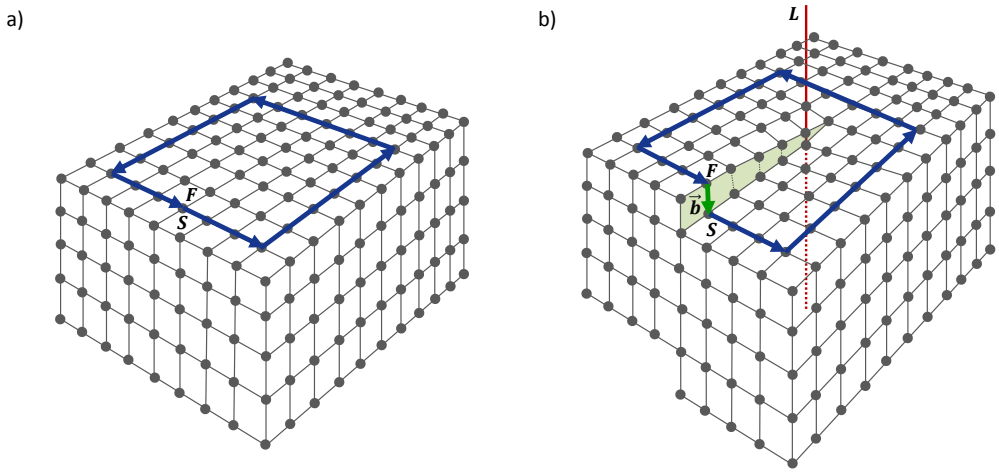


Figure 2.3: Schematic of a screw dislocation in a cubic crystal. (a) Perfect crystal: the reference Burgers circuit delineated in blue starts at position S and ends at the same position $F = S$. (b) Screw dislocation: when drawing the same circuit in the dislocated crystal, the Burgers vector $\vec{\boldsymbol{FS}} = \vec{\boldsymbol{b}}$ is required to close the circuit. In this case, $\vec{\boldsymbol{b}}$ is parallel to the dislocation line L shown in red. The surface in green delineates the plane of atoms on which slip has occurred.

As mentioned earlier, both edge and screw dislocations correspond to particular orientations. In general, the orientation of a dislocation found in a real material is not constant along its line, but typically continuously evolves from edge to screw characters, as illustrated in figure 2.4. When the Burgers vector $\vec{\boldsymbol{b}}$ is neither perpendicular

nor parallel to the line direction, the dislocation line is of mixed character.

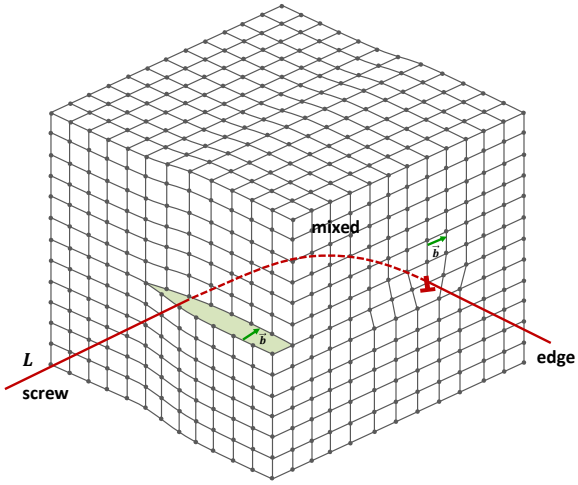


Figure 2.4: Schematic of a dislocation line L found in a cubic crystal. The line continuously evolves from edge to screw characters. When the Burgers vector $\vec{\boldsymbol{b}}$ is neither perpendicular nor parallel to the line direction, the dislocation line is of mixed character.

From a more general perspective, a dislocation can be imagined as resulting from a cut introduced over a surface $\vec{\boldsymbol{S}}$ within an otherwise continuous material. As illustrated in figure 2.5(a), let $\vec{\boldsymbol{S}}$ be the surface defined by its normal $\vec{\boldsymbol{n}}$ and bounded by line L (here $\vec{\boldsymbol{S}}$ denotes the ensemble of points $\vec{\boldsymbol{x}}$ located on the surface). If the domain S^+ of the crystal located above the surface of the cut $\vec{\boldsymbol{S}}$ is slipped by an amount $\|\vec{\boldsymbol{b}}\|$ in the direction of $\vec{\boldsymbol{b}}/\|\vec{\boldsymbol{b}}\|$ with respect to the domain S^- located below the cut, then a dislocation with Burgers $\vec{\boldsymbol{b}}$ and line L is introduced in the material. This is evidenced by the fact that taking the Burgers circuit $\mathcal{C}$ anywhere around dislocation line L will yield the Burgers vector $\vec{\boldsymbol{b}}$ in this case. Therefore, a dislocation can be seen as the boundary of a surface in which slip has occurred by an amount $\vec{\boldsymbol{b}}$, such that the jump in displacement $[\vec{\boldsymbol{u}}]$ across surface $\vec{\boldsymbol{S}}$, i.e. when going from S^- to S^+, is given by $[\vec{\boldsymbol{u}}] = \vec{\boldsymbol{b}}$. In other words, there exists a discontinuity in the displacement on surface $\vec{\boldsymbol{S}}$, such that the displacements are multi-valued on this surface. Further, the fact

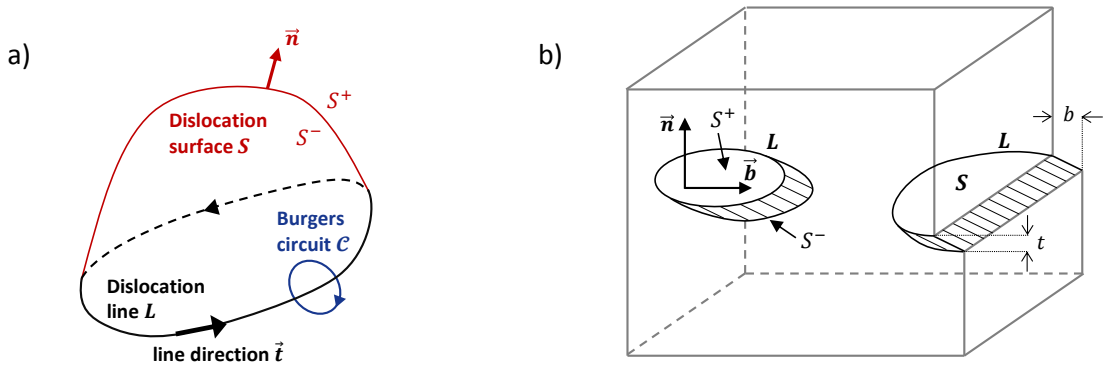


Figure 2.5: (a) Schematic of a dislocation loop L defined as the boundary of a cut introduced over a surface $\vec{S}$ within a continuous material. Here, S^+ denotes the region of the crystal above surface $\vec{S}$ while S^- denotes the region below the surface. A dislocation with Burgers vector $\vec{b}$ is introduced when the crystal in domain S^+ is slipped by an amount $b = \|\vec{b}\|$ in the direction of $\vec{b}/\|\vec{b}\|$ with respect to the crystal in domain S^-. With that, the Burgers circuit taken anywhere along L yields the Burgers vector $\vec{b}$ such that the discontinuity $[\vec{u}]$ in the displacement field across surface $\vec{S}$ when going from S^- to S^+ is given by $[\vec{u}] = \vec{b}$. (b) In the eigenstrain theory, dislocations are considered as plate-like Eshelbian inclusions of thickness t. Surface $\vec{S}$ corresponding to the slip plane of the dislocation is defined by the plane formed by the Burgers vector $\vec{b}$ and the line direction $\vec{t}$ such that $\vec{n} = \frac{\vec{b}\times\vec{t}}{\|\vec{b}\times\vec{t}\|}$. For a dislocation, t corresponds to the inter-atomic distance associated with its slip plane.

that a dislocation is solely defined by the boundary L of a surface induces two main consequences. First, it implies that a dislocation cannot end within a perfect crystal, but must be closed or terminate at the intersection with another defect, such as a dislocation, a disclination or a grain boundary. In other words, this implies that the Burgers vector must be conserved along the dislocation line. Second, it implies that the definition of the dislocation is independent of the shape of surface $\vec{S}$ bounded by L. In general, $\vec{S}$ is the plane formed by the Burgers vector $\vec{b}$ and the line direction $\vec{t}$ of the dislocation that together define its glide plane (see Section 2.2.4). As shown in figure 2.5(b), this definition directly leads to the representation of dislocations in the eigenstrain theory, in which they are considered as plate-like inclusions.

2.2.2 Statics of dislocations

The presence of a dislocation in a crystal primarily gives rise to internal strain and stress fields associated with the elastic displacement field induced by the dislocation. Therefore, a brief review of the elasticity theory is given here.

In continuum mechanics, the state at each material point of an elastic medium is primarily described by two quantities: the stress and the strain. Both these quantities are second-order tensors and their continuous distribution at each point over the medium defines the strain and stress fields, respectively. The stress tensor is usually denoted $\boldsymbol{\sigma}$, for which each component σ_{ij} represents the force on the i-th direction acting on a surface whose unit normal is parallel to the positive j-th direction. In a three-dimensional setting, indices i and j range from values 1 to 3. When the Cauchy definition of the stress tensor is used, the latter is symmetric, i.e. $\sigma_{ij} = \sigma_{ji}$, such that the mechanical equilibrium of an elementary volume element at rest is respected. Further, the equilibrium equation writes:

$$\sigma_{ij,j}(\vec{\boldsymbol{x}}) + f_i(\vec{\boldsymbol{x}}) = 0 \tag{2.2}$$

where f_i denotes the body force, and where the Einstein summation convention is used. In words, equation (2.2) states that the divergence of the stress must be balanced by the body forces at each material point $\vec{\boldsymbol{x}}$ of the medium.

In contrast to the stress which relates to forces, the strain is related to the deformation, i.e. to the response to the force. The definition of the total strain tensor, usually denoted $\boldsymbol{\epsilon}$, is given from the displacement $\vec{\boldsymbol{u}}$ by the compatibility equation as:

$$\epsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \tag{2.3}$$

Note that definition of the strain in equation (2.3) is an approximation valid under the small strain hypothesis. The constitutive equation relating the stress to the strain

is given by Hooke's law as:

$$\sigma_{ij} = C_{ijkl}\epsilon^e_{kl} \tag{2.4}$$

where ϵ^e_{kl} is the elastic strain and C_{ijkl} are the coefficients of the fourth-order stiffness tensor. From the symmetries of the strain and stress tensors, it follows that the stiffness matrix respects major and minor symmetries:

$$C_{ijkl} = C_{jikl} = C_{ijlk} = C_{klij} \tag{2.5}$$

In the presence of a dislocation, a plastic strain distribution $\boldsymbol{\epsilon}^p$ is prescribed in order to account for the inelastic deformation induced by the displacement discontinuity introduced in Section 2.2.1. In the case of infinitesimal deformations, the total strain can be written as the sum of the elastic and plastic contributions, i.e. $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}^e + \boldsymbol{\epsilon}^p$. With this, the elasto-plastic formulation of equation (2.4) rewrites:

$$\sigma_{ij} = C_{ijkl}\left(\epsilon_{kl} - \epsilon^p_{kl}\right) \tag{2.6}$$

Similarly, the gradient of the displacement $u_{i,j}$, also referred to as the total distortion β_{ji}, is decomposed as the sum of the elastic distortion β^e_{ij} and the plastic distortion β^p_{ij}, such that:

$$u_{i,j} = \beta_{ji} = \beta^e_{ji} + \beta^p_{ji} \tag{2.7}$$

$$\text{and} \quad \epsilon^p_{ij} = \frac{1}{2}\left(\beta^p_{ij} + \beta^p_{ji}\right) \tag{2.8}$$

In the absence of body forces, the equilibrium equation (2.2) applied to (2.6) yields the following expression that must hold at every point $\vec{\boldsymbol{x}}$ in the medium:

$$C_{ijkl}u_{k,lj}(\vec{\boldsymbol{x}}) = C_{ijkl}\epsilon^p_{kl,j}(\vec{\boldsymbol{x}}) \tag{2.9}$$

When the distribution of plastic strain $\epsilon^p_{kl,j}(\vec{\boldsymbol{x}})$ is known throughout the medium, expression (2.9) can be integrated to solve for the displacement field $\vec{\boldsymbol{u}}(\vec{\boldsymbol{x}})$ everywhere in the medium. Usually, such integration is performed via the introduction of the Green function G_{ij}, which defines the displacement in the i-th direction in response to a point force exerted in the j-th direction. More details on the Green function and a method to determine its expression are given in [72, 181] and in Section 2.2.3. Solving equation (2.9) leads to the expression of the displacement field as a function of the prescribed plastic strain as:

$$u_i(\vec{\boldsymbol{x}}) = -\int_V C_{klmn}\epsilon^p_{mn}(\vec{\boldsymbol{x'}})G_{ik,l}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})dV' \tag{2.10}$$

where the integration is performed over volume V where the plastic strain $\epsilon^p_{mn}(\vec{\boldsymbol{x'}})$ does not vanish. Complete details for obtaining equation (2.10) from (2.9) will be given in Section 3.3. In the case of the presence of a dislocation line defined by its Burgers vector $\vec{\boldsymbol{b}}$ and an arbitrary defect surface $\vec{\boldsymbol{S}}$ with unit normal $\vec{\boldsymbol{n}}$, the associated plastic distortion is given as [181]:

$$\beta^p_{ij} = -b_j n_i \delta(\vec{\boldsymbol{S}} - \vec{\boldsymbol{x}}) \tag{2.11}$$

where $\delta(\vec{\boldsymbol{S}} - \vec{\boldsymbol{x}})$ denotes the three-dimensional Dirac delta function that is unbounded on surface $\vec{\boldsymbol{S}}$ and 0 otherwise. Here, as depicted in figure 2.5, $\vec{\boldsymbol{S}}$ is the cut surface bounded by the dislocation line L and on which the displacements are multi-valued. Therefore, $\delta(\vec{\boldsymbol{S}} - \vec{\boldsymbol{x}})$ here accounts for the displacement discontinuity $[\vec{\boldsymbol{u}}] = \vec{\boldsymbol{b}}$ across $\vec{\boldsymbol{S}}$. From expression (2.8), the plastic strain associated with (2.11) writes:

$$\epsilon^p_{ij} = -\frac{1}{2}\left(b_i n_j + b_j n_i\right)\delta(\vec{\boldsymbol{S}} - \vec{\boldsymbol{x}}) \tag{2.12}$$

Combining expressions (2.10) and (2.12) leads to the following surface integral to define the total displacement field induced by the dislocation:

$$u_i(\vec{\boldsymbol{x}}) = \int_S C_{klmn} G_{ik,l}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_n n_m dS' \tag{2.13}$$

Equation (2.13) is the original formula obtained by Volterra [241]. Differentiating this equation to obtain the total distortion $u_{i,j}$ allows for an expression of the total strain using equation (2.3). The latter can be further converted to a line integral upon using Stokes' theorem [181]. From there, the stress induced by the presence of a dislocation line is obtained from equation (2.6) as:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \oint_L e_{lnh} C_{pqmn} G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_m dx'_h \tag{2.14}$$

where the integration is performed along the dislocation line L defined as the boundary of surface $\vec{\boldsymbol{S}}$ and with elementary line direction $d\vec{\boldsymbol{x'}}$, and where e_{ijk} is the permutation tensor. Equation (2.14) appears to have been first obtained by Mura [180] and constitutes the basis for evaluating the stress field of discrete dislocation lines. Note here that the stress field is given as a closed line integral, i.e. for a closed dislocation loop. In practice however, since a dislocation loop can be decomposed into the sum of parametric segments connected to one another, the stress field produced by a dislocation segment can be evaluated using equation (2.14). Nevertheless, difficulties arise when evaluating the stress of terminating segments, since the solution cannot be uniquely determined in that case [80, 14].

2.2.3 Elastic stress field of a static dislocation

As detailed in Section 2.2.2 and in [72, 181], the stress field $\boldsymbol{\sigma}(\vec{\boldsymbol{x}})$ of a dislocation line L with Burgers vector $\vec{\boldsymbol{b}}$ is obtained at any field point $\vec{\boldsymbol{x}}$ from expression (2.14) as:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \oint_L e_{lnh} C_{pqmn} G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_m dx'_h \tag{2.14 repeated}$$

where C_{ijkl} is the fourth order elastic stiffness tensor, e_{ijk} is the permutation tensor,

$\vec{\boldsymbol{x'}}$ is the coordinate that spans the dislocation, $G_{kp,q} = \partial G_{kp}/\partial x_q$ and $G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})$ is the static Green function for an homogeneous elastic medium. $G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})$ defines the displacement in the i-th direction at point $\vec{\boldsymbol{x}}$ due to an elementary force applied in the j-th direction at point $\vec{\boldsymbol{x'}}$. It can be shown that the Green function satisfies [99]:

$$C_{ijkl} G_{km,lj}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) + \delta_{im}\delta(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) = 0 \tag{2.15}$$

where δ_{im} denotes the Kronecker symbol and $\delta(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})$ is the three-dimensional delta function that vanishes everywhere expect where $\vec{\boldsymbol{x}} = \vec{\boldsymbol{x'}}$, i.e. on the dislocation line. In equation (2.14), the evaluation of the stress field of a dislocation requires an expression of the Green function which has not been given thus far. In the following, expressions for the Green function in the case of isotropic and anisotropic elasticity are provided.

2.2.3.1 Isotropic elasticity

There are several ways in which the Green function can be determined. Usually, such is done by solving for equation (2.15) in an infinite medium under the assumption that $G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})$ vanishes at infinity, i.e. $G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) \to 0$ when $\|\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}\| \to \infty$. The use of Fourier transforms techniques is then probably the most straightforward and elegant approach to obtain an expression for G_{ij} [181]. In the case of an elastically isotropic medium, the need of solely two elastic constants to describe the elastic properties of the medium greatly simplifies its determination. Thus, for isotropic elasticity, the static Green's function is analytically expressed as [99]:

$$G_{kp}(\vec{\boldsymbol{R}}) = \frac{1}{8\pi\mu}\left[\delta_{kp} R_{,qq} - \frac{1}{2(1-\nu)} R_{,kp}\right] \tag{2.16}$$

where μ and ν are the shear modulus and the Poisson's ratio, δ_{ij} is the Kronecker tensor and $\vec{\boldsymbol{R}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}$ is the radius vector of norm $R = \|\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}\| = \sqrt{\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{R}}} = \sqrt{R_i R_i}$.

Combining equations (2.14) and the first derivative of (2.16), the stress field of a dislocation line in an elastically isotropic infinite medium is expressed as [72]:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = \frac{\mu b_n}{8\pi} \int_L \Big[R_{,mpp} \left(e_{jmn} dx'_i + e_{imn} dx'_j \right) + \frac{2}{1-\nu} e_{kmn} \left(R_{,ijm} - \delta_{ij} R_{,ppm} \right) dx'_k \Big] \tag{2.17}$$

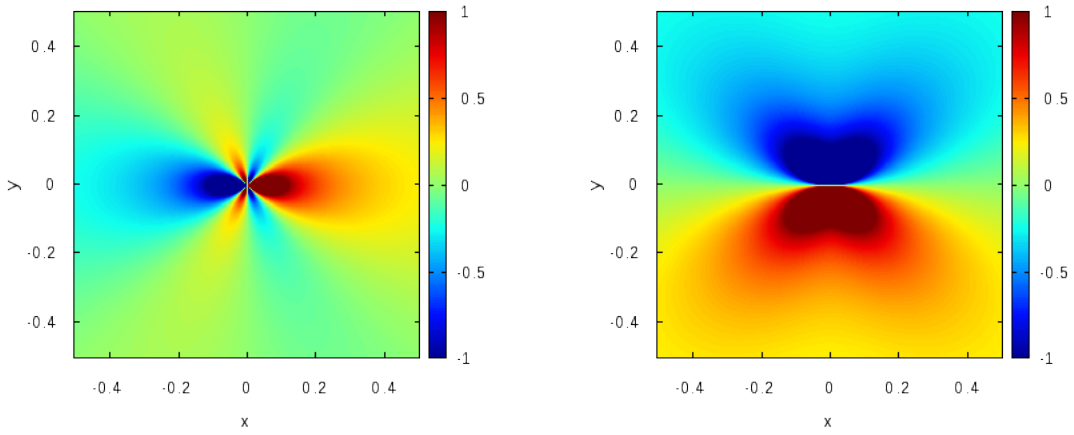
where the derivatives of the radius norm R in (2.16) and (2.17) are given by:

$$\begin{aligned} R_{,i} &= \frac{R_i}{R} \\ R_{,ij} &= \left(\delta_{ij} - \frac{R_i}{R} \frac{R_j}{R} \right) / R \\ R_{,ijk} &= \left[3 \frac{R_i}{R} \frac{R_j}{R} \frac{R_k}{R} - \left(\delta_{ij} \frac{R_k}{R} + \delta_{jk} \frac{R_i}{R} + \delta_{ki} \frac{R_j}{R} \right) \right] / R^2 \end{aligned} \tag{2.18}$$

Full details on the derivation of relations (2.18) are provided in Appendix A.

Here, it is important to notice that $R_{,ijk}$ becomes unbounded as $R \to 0$. In other words, equation (2.17) provides a singular expression of the stress field when evaluated on the dislocation line, i.e. when $\vec{\boldsymbol{x}} = \vec{\boldsymbol{x'}}$. As will be shown in the next paragraph, this results from the fact that the Green's function is not defined for $\vec{\boldsymbol{R}} = \vec{\boldsymbol{0}}$ in the case of general anisotropy, such that the general formulation (2.14) is similarly singular along the dislocation line L. This indicates that with this continuum approach, the stress field becomes unbounded in the dislocation core, which obviously does not occur in real crystals. To address this discrepancy, Cai et al. recently proposed a non-singular formulation, in which the dislocation core is spread according to an isotropic distribution [35]. Besides removing the singularity along dislocation line, this formulation allows for obtaining an explicit analytical formulation of the stress field induced by straight segments in isotropic elasticity. More details on the non-singular formulation are given in Appendix A.2.

For an infinite straight edge dislocation align with the z-axis and with Burgers



(a) σ_{xy} (b) σ_{xx}

(c) σ_{yy} (d) σ_{zz} (plane strain)

Figure 2.6: Non-zero components of the stress field produced by an infinitely long straight edge dislocation along the z-axis. The Burgers vector $\vec{\boldsymbol{b}}$ of magnitude $b = \|\vec{\boldsymbol{b}}\| = 1$ is parallel to the positive x direction and the medium is isotropic elastic with material parameters $\mu = 0.6$ and $\nu = 0.2$.

vector $\vec{\boldsymbol{b}} = [b00]$ parallel to the x-axis, the stress field components given by the classical singular theory in equation (2.17) reduce to:

$$\sigma_{xx} = -\frac{\mu b}{2\pi(1-\nu)}\frac{y(3x^2+y^2)}{(x^2+y^2)^2}$$

$$\sigma_{yy} = \frac{\mu b}{2\pi(1-\nu)}\frac{y(x^2-y^2)}{(x^2+y^2)^2}$$

$$\sigma_{xy} = \frac{\mu b}{2\pi(1-\nu)}\frac{x(x^2-y^2)}{(x^2+y^2)^2}$$

$$\sigma_{zz} = \nu(\sigma_{xx}+\sigma_{yy}) = -\frac{\mu b\nu}{\pi(1-\nu)}\frac{y}{x^2+y^2}$$

$$\sigma_{xz} = \sigma_{yz} = 0 \tag{2.19}$$

where component σ_{zz} is given for a plane strain situation ($\epsilon_{zz} = 0$). A graphical representation of the non-zero components of the stress field in a plane perpendicular to the line direction is given in figure 2.6.

In the case of an infinite screw dislocation along the z-axis with Burgers vector $\vec{\boldsymbol{b}} = [00b]$ parallel to the z-axis, the stress field components given by equation (2.17) reduce to:

$$\sigma_{xz} = -\frac{\mu b}{2\pi}\frac{y}{x^2+y^2}$$

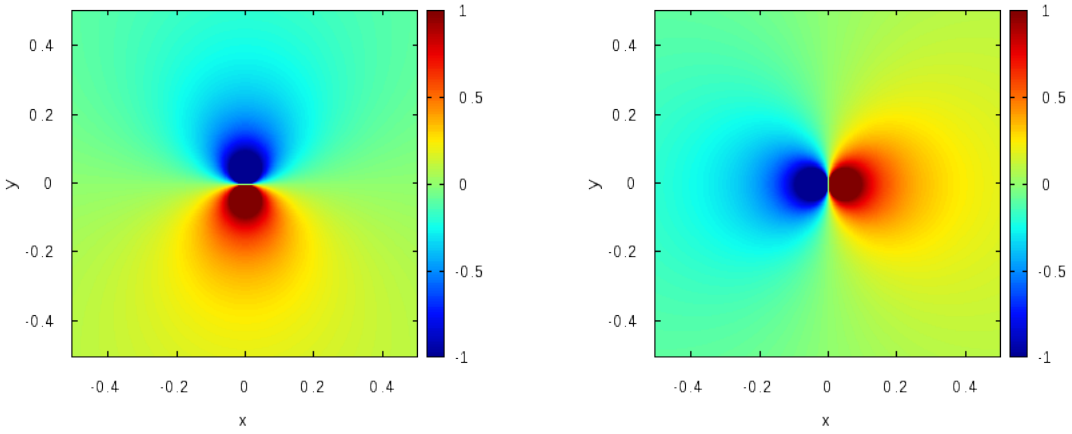
$$\sigma_{yz} = \frac{\mu b}{2\pi}\frac{x}{x^2+y^2}$$

$$\sigma_{xx} = \sigma_{yy} = \sigma_{xy} = \sigma_{zz} = 0 \tag{2.20}$$

for which a graphical representation is given in figure 2.7.

2.2.3.2 Anisotropic elasticity

In the case of general anisotropic elasticity, the determination of the Green function is more complex, and requires to solve a sextic equation in terms of the Stroh eigenvalues [230]. Following the formalism proposed by Ting and Lee [230], explicit formulations of first and second derivatives of the Green's function have been recently derived



(a) σ_{xz} (b) σ_{yz}

Figure 2.7: Non-zero components of the stress field produced by an infinitely long straight screw dislocation along the z-axis. The Burgers vector $\vec{\boldsymbol{b}}$ of magnitude $b = \|\vec{\boldsymbol{b}}\| = 1$ is parallel to the z direction and the medium is isotropic elastic with material parameters $\mu = 0.6$ and $\nu = 0.2$.

[146, 32], but their numerical evaluation cannot be directly obtained. Note that analytical solutions for the stress field of dislocations in transverse isotropy have been obtained by Pan and Chou [46, 190]. In the general case however, no closed form expression can be found. The first approaches to evaluate G_{ij} in an anisotropic medium concentrated on obtaining approximations of the solution for equation (2.15) by expanding G_{ij} in series form [182]. Later, Bacon and Barnett proposed an angular integral expression of the Green function and its derivatives [17, 14] for anisotropic elasticity. With this approach, the expression of G_{ij} is given as:

$$G_{kp}(\vec{\boldsymbol{R}}) = \frac{1}{4\pi^2 R} \int_0^{\pi} M_{kp}^{-1}(\vec{\boldsymbol{\xi}}) d\psi \tag{2.21}$$

where $\vec{\boldsymbol{R}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}$ is the radius vector of norm $R = \|\vec{\boldsymbol{R}}\|$. By denoting $(\vec{\boldsymbol{e}}_{R1}, \vec{\boldsymbol{e}}_{R2})$ any arbitrary plane orthogonal to $\vec{\boldsymbol{R}}$ and $\vec{\boldsymbol{T}} = \vec{\boldsymbol{R}}/R$ the unit vector direction of $\vec{\boldsymbol{R}}$, $\vec{\boldsymbol{\xi}}(\psi)$ denotes the unit vector on plane $(\vec{\boldsymbol{e}}_{R1}, \vec{\boldsymbol{e}}_{R2})$ making an angle ψ with $\vec{\boldsymbol{e}}_{R1}$. By definition, $\vec{\boldsymbol{\xi}}(\psi)$ satisfies $\vec{\boldsymbol{T}} \cdot \vec{\boldsymbol{\xi}} = 0$. With this, the tensor $\boldsymbol{M}^{-1}$ in equation (2.21) is expressed as:

$$M_{kp}^{-1}(\vec{\boldsymbol{\xi}}) = \frac{e_{ksm}e_{prw}K_{sr}(\vec{\boldsymbol{\xi}})K_{mw}(\vec{\boldsymbol{\xi}})}{2e_{lgn}K_{1l}(\vec{\boldsymbol{\xi}})K_{2g}(\vec{\boldsymbol{\xi}})K_{3n}(\vec{\boldsymbol{\xi}})} \tag{2.22}$$

where $K_{ik}(\vec{\boldsymbol{\xi}}) = C_{ijkl}\xi_j\xi_l$ and the elastic tensor $\boldsymbol{C}$ is expressed in the global frame $(\vec{\boldsymbol{e}}_1, \vec{\boldsymbol{e}}_2, \vec{\boldsymbol{e}}_3)$ in which the coordinates of $\vec{\boldsymbol{\xi}}$ are also expressed. The first derivative of the anisotropic Green's function expressed in equations (2.21) and (2.22) is given by [17]:

$$G_{kp,q}(\vec{\boldsymbol{R}}) = \frac{1}{4\pi^2 R^2}\int_0^{\pi}\left(-T_q M_{kp}^{-1}(\vec{\boldsymbol{\xi}}) + \xi_q C_{jrnw}M_{kj}^{-1}(\vec{\boldsymbol{\xi}})M_{np}^{-1}(\vec{\boldsymbol{\xi}})(\xi_r T_w + \xi_w T_r)\right)d\psi \tag{2.23}$$

for which no closed form solution exists. Therefore, the evaluation of the stress field of a dislocation in an anisotropic medium requires the calculation of nested numerical integrations given in equations (2.23) and (2.14). More details on the calculation and approximations of stress fields in anisotropic elasticity using expression (2.23) are given in Appendix B.

2.2.4 Dislocation glide

In Sections 2.2.2 and 2.2.3, the theory of static dislocations was introduced, i.e. dislocations were considered as static defects under no motion. However, when subjected to an applied stress – which may arise from a macroscopic loading or from the distortion induced by the presence of other dislocations or defects –, dislocations can glide and propagate. The resulting motion of an ensemble of gliding dislocations is generally referred to as dislocation slip. As a matter of fact, slip is one of the most common deformation modes in metallic materials and is characteristic of the plastic regime.

The permanent displacement of atoms, corresponding to the slipping of atomic planes with respect to each other, is associated with the plastic deformation and is facilitated by the presence of dislocation lines within the material: as depicted in

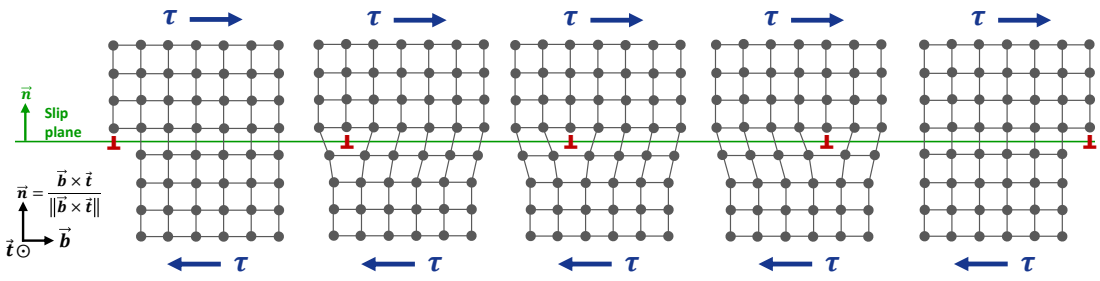


Figure 2.8: Schematic of the glide of an edge dislocation. The snapshots from left to right illustrate the glide of a dislocation line under an applied shear stress τ. The bonds between atoms successively break and reconnect as the dislocation propagates through the crystal. The slip plane on which gliding occurs is defined by the Burgers vector $\vec{\boldsymbol{b}}$ and the line direction $\vec{\boldsymbol{t}}$ (orthogonal to the paper in this case) of the dislocation such that its unit normal is given by $\vec{\boldsymbol{n}} = \frac{\vec{\boldsymbol{b}}\times\vec{\boldsymbol{t}}}{\|\vec{\boldsymbol{b}}\times\vec{\boldsymbol{t}}\|}$. At the end of the deformation, the dislocation has entirely sheared the crystal such that the part of the crystal above the slip plane has slipped by one lattice distance with respect to the bottom part.

figure 2.8, the glide of a dislocation line follows a step by step process in which atomic bonds around the dislocation are successively broken and reconnected, inducing a relative slip between both sides of the slip plane. Without the presence of such lattice defect, the slip of atomic planes would be much more difficult, since it will require the simultaneous breaking of all atomic bonds across the slip plane, thereby requiring a much higher energy. Besides, the atomic planes on which slip occurs generally correspond to the planes and directions in which the atoms are the most packed. This is because the step that needs to be overcome by atoms to be moved to their next lattice position is all the smaller than the plane and direction are dense, therefore requiring less energy. As a result, the combinations of slip plane and slip direction – referred to as slip systems – on which plastic deformation preferably occurs depend on the crystalline structure of the material. Thus, for instance, slip primarily occurs by motion of dislocations of the family of $\frac{1}{2}\langle\bar{1}10\rangle\{111\}$ slip systems in face-centered cubic (FCC) materials, where $\frac{1}{2}\langle\bar{1}10\rangle$ refers to the family of Burgers vectors and $\{111\}$ to the family of slip planes associated with these slip systems.

Note that different properties are associated with the slip systems depending on

the geometrical and physical properties of the dislocations that constitute them. For instance, the lattice friction resistance and the dislocation mobility governing dislocation motion are intrinsic properties of slip systems and crystalline structures.

When considering a single dislocation line as in figure 2.8, its slip plane, defined by its unit normal $\vec{\boldsymbol{n}}$, is determined by its Burgers vector $\vec{\boldsymbol{b}}$, giving the magnitude and the direction of the slip, and its line direction $\vec{\boldsymbol{t}}$ such that $\vec{\boldsymbol{n}} = \frac{\vec{\boldsymbol{b}} \times \vec{\boldsymbol{t}}}{\|\vec{\boldsymbol{b}} \times \vec{\boldsymbol{t}}\|}$. In other words a dislocation is generally constrained to move on its plane. However, the case of a screw dislocation is particular since no slip plane can be defined: the Burgers vector and the line direction are collinear in this case. For this reason, screw dislocations have the ability to cross-slip, i.e. to change slip planes during glide. Thus, when local conditions are met, a screw portion of a dislocation line can potentially glide on any slip planes containing its Burgers vector. This thermally-activated mechanism is for instance particularly encountered in the bypassing process of obstacles lying on the original slip plane of a dislocation.

As illustrated in figure 2.8, the propagation of a dislocation shears the crystal along its slip plane. The elementary plastic shear strain γ associated with the motion of a dislocation is given as:

$$\gamma = \frac{bA}{V} \tag{2.24}$$

where $b = \|\vec{\boldsymbol{b}}\|$ is the magnitude of the Burgers vector, A is the area swept by the dislocation during glide, and V is the volume of the sheared crystal. Without further details, equation (2.24) can be intuitively obtained from the purposely oversimplified following reasoning: let the crystal presented in figure 2.8 be of height h (along $\vec{\boldsymbol{n}}$), length l (along $\vec{\boldsymbol{b}}$) and width w (along $\vec{\boldsymbol{t}}$). When a dislocation of length w has fully propagated through the crystal from left to right, the upper part of the crystal is translated by an amount b with respect to the lower part such that the resulting amount of shear in the crystal is given by $\gamma = b/h$. In this process, the dislocation

has travelled a distance l and, consequently, swept an area $A = lw$. Therefore, the resulting shear can be expressed as $\gamma = bA/hA$ where $V = hA$ is the volume of the crystal, thereby recovering equation (2.24). Extending this reasoning to the collective glide of dislocations, the total plastic shear resulting from the global motion of dislocations on slip system s is given as:

$$\gamma^s = \sum_{i \in s} \frac{b^s A_i^s}{V} \tag{2.25}$$

where the sum is performed on all dislocations i on system s, A_i^s is the area sheared by dislocation i, and b^s is the magnitude of the Burgers vector associated with slip system s. With that, the plastic strain $\boldsymbol{\epsilon}^p$ and the plastic rotation $\boldsymbol{\omega}^p$ resulting from the collective dislocation slip on all slip systems are given as:

$$\boldsymbol{\epsilon}^p = \frac{1}{2} \sum_s \left(\vec{\boldsymbol{b}}^{\,s} \otimes \vec{\boldsymbol{n}}^{\,s} + \vec{\boldsymbol{n}}^{\,s} \otimes \vec{\boldsymbol{b}}^{\,s} \right) \gamma^s \tag{2.26}$$

$$\boldsymbol{\omega}^p = \frac{1}{2} \sum_s \left(\vec{\boldsymbol{b}}^{\,s} \otimes \vec{\boldsymbol{n}}^{\,s} - \vec{\boldsymbol{n}}^{\,s} \otimes \vec{\boldsymbol{b}}^{\,s} \right) \gamma^s \tag{2.27}$$

2.3 Discrete dislocation representation

In this section, the discrete representation of dislocation lines in DDD simulations is presented, and the numerical approach to model the physical properties of dislocations that are described in Section 2.2 is introduced.

2.3.1 Line discretization

2.3.1.1 Nodal approach

Several approaches have been developed to represent and discretize the dislocation network in DDD simulations. The first type of approaches rely on a discretization scheme in which dislocation segments – successively connected to one another –, were

restricted to a fix set of orientations. This representation resulted from the argument that the positions along dislocation lines should correspond to lattice sites, thereby geometrically constraining dislocation segments orientations. Following this lattice-based representation, the first simulations solely relied on pure edge and pure screw segments [67, 240]. Later, additional orientations were introduced, especially to better represent the formation of junctions whose orientations cannot be well described solely relying on pure edge or pure screw characters [164, 68]. Further, the need for a more accurate representation allowing for a continuous description of orientations led to the development of nodal approaches that have progressively gained interest and are currently widely used [266, 96]. A nodal-based scheme is adopted and implemented in the present work.

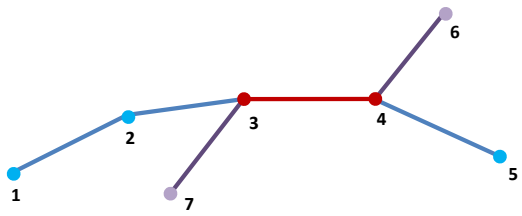


Figure 2.9: Schematic of nodal dislocation line discretization. The dislocation network is discretized into nodes connected to one another by parametric segments. Physical nodes correspond to nodes that have three or more connections, while nodes with two connections are referred to as discretization nodes. Nodes that possess only one connection correspond to end nodes associated with the artificial end of the defect within the crystalline structure.

In the nodal approach, dislocation lines are discretized into segments connected to one another through dislocation nodes, as depicted on figure 2.9. With this approach, the motion of dislocation lines is governed by the motion of their nodes, thereby allowing dislocation segments to attain any orientation. Each node can have one or several connections. A node that possesses only one connection is an end node and is physically constrained not to move, as it artificially corresponds to the end of a

defect within the lattice structure. A node that has three or more connections is a junction node as it corresponds to the intersection between two or more dislocations. In that sense, end nodes and junction nodes are considered as physical nodes, whereas nodes that possess two connections are referred to as discretization nodes, and are only introduced to allow for a correct description of the curvature of dislocation lines.

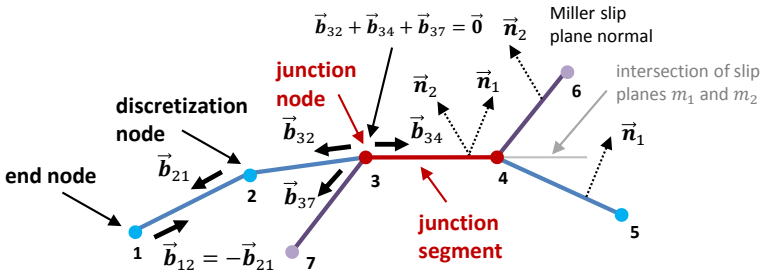


Figure 2.10: Schematic of a dislocation group and its properties. To each nodal connection is associated a Burgers vector and the normal of the glide plane(s) of dislocation segments. To maintain the Burgers continuity, each dislocation node i that is not an end node must respect $\sum_j \vec{\boldsymbol{b}}_{ij} = 0$, where the sum is performed on every connection j and $\vec{\boldsymbol{b}}_{ij}$ is the Burgers vector of oriented segment ij.

To accurately model the behavior of dislocations, discrete segments need to account for the physical properties presented in Section 2.2. Therefore, as depicted in figure 2.10, each dislocation segment is described by a Burgers vector and a glide plane defined by a Miller index. To account for the polarity of the line direction, the Burgers vector of a segment takes plus (+) or minus (−) signs depending on the orientation of the segment. Thus, for two nodes i and j that are connected together, segment ij, defined as oriented from node i to node j, possesses a Burgers vector $\vec{\boldsymbol{b}}_{ij}$ opposite to that of segment ji defined from node j to node i whose Burgers vector is $\vec{\boldsymbol{b}}_{ji} = -\vec{\boldsymbol{b}}_{ij}$. This directly results from the necessity to maintain Burgers continuity along dislocation lines [111]. In other words, with this convention, it follows that the sum of Burgers vectors at each node i connected to nodes j respects $\sum_j \vec{\boldsymbol{b}}_{ij} = 0$,

provided that node i is not an end node. In practise, it is easier to assign the Burgers vectors of the dislocation segments directly to the nodes. Thus, for each connection, a node i is associated with (1) the node j to which it is connected and (2) the Burgers vector $\vec{\boldsymbol{b}}_{ij}$ of segment ij going from node i to node j.

The Miller indices associated to dislocation segments define the normal of the slip planes dislocations are gliding on. Depending on the local configuration, the motion of dislocation segments can be constrained to lie in one to three slip planes. Thus, a non-junction segment is necessarily gliding on a single slip plane, and the pair formed by its slip plane and its Burgers vector defines its slip system. However, junction segments, resulting from the intersection of several dislocations gliding on different planes, are generally constrained to move along the intersection of the slip planes associated with the initial segments that formed it. Physically however, the slip plane of a dislocation is defined by the plane formed by its Burgers vector and its line direction (see Section 2.2.4). If such plane is dense and allows for slip – considering the friction resistance exerted by the lattice and the mobility of dislocations along that plane –, a dislocation will be able to glide on that plane under a sufficient resolved shear stress (RSS).

Note that, as will be discussed in Section 2.5, in DDD, the motion of a node connected to segments with two distinct slip planes is constrained to the line defined by the intersection of the two planes, while a node connected to segments with three distinct planes is geometrically constrained to be fixed, since the intersection between three non-coplanar planes reduces to a single point. In practice, glide planes are associated to the segments such that, when node i and node j are connected together, segment ij and segment ji are assigned the same Miller indices.

To allow any type of crystallography to be simulated, the DDD code developed in this work relies on a set of crystallographic planes $\{\vec{\boldsymbol{n}}\}$, slip directions $\{\vec{\boldsymbol{b}}\}$ and available slip system properties (magnitude of the Burgers vector, lattice friction

stress, dislocation mobilities), that define the available slip systems on which discrete dislocation segments can move. These information are incorporated into the DDD framework for common crystalline structures (FCC, BCC, HCP) or can be imputed to treat complex user-defined crystal structures.

2.3.1.2 Segments parametrization

Depending on the approaches, dislocation segments can be represented by straight segments or splines. In the general case, each dislocation segment ij can be described by a parametric line spanned with coordinate s for $0 \leq s \leq 1$, such that the position $\vec{\boldsymbol{x}}_{ij}(s)$ at coordinate s along the segment can be obtained from an interpolation between N_d coordinates as:

$$\vec{\boldsymbol{x}}_{ij}(s) = \sum_{a}^{N_d} N_a(s) \vec{\boldsymbol{q}}_{ij}^{a} \tag{2.28}$$

where $N_a(s)$ denotes the interpolation function associated with the a-th generalized coordinate $\vec{\boldsymbol{q}}_{ij}^{a}$ describing segment ij. The spline approximation is generally performed using a quadratic or cubic interpolation between dislocation node positions and requires the definition of the line tangent at each node [97, 251, 21]. The tangent vector $\vec{\boldsymbol{t}}_{ij}(s)$ along segment ij can be expressed as:

$$\vec{\boldsymbol{t}}_{ij}(s) = \sum_{a}^{N_d} N'_a(s) \vec{\boldsymbol{q}}_{ij}^{a} \tag{2.29}$$

where $N'_a(s)$ denotes the derivative of the shape function with respect to the curvilinear abscissa s. For a cubic spline interpolation, the generalized coordinates are given by [96]:

$$
\begin{aligned}
N_d &= 4 \\
\vec{\boldsymbol{q}}_{ij}^{\,1} = \vec{\boldsymbol{x}}_i \; ; &\quad \vec{\boldsymbol{q}}_{ij}^{\,2} = \vec{\boldsymbol{x}}_j \\
\vec{\boldsymbol{q}}_{ij}^{\,3} = \vec{\boldsymbol{t}}_i \; ; &\quad \vec{\boldsymbol{q}}_{ij}^{\,4} = \vec{\boldsymbol{t}}_j
\end{aligned}
\tag{2.30}
$$

where $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{t}}_i$ denote the position and the tangent vector at node i, respectively, while the interpolation functions and their derivatives are given by:

$$
\begin{aligned}
N_1(s) = 2s^3 - 3s^2 + 1 \quad \text{and} \quad N_1'(s) = 6s^2 - 6s \\
N_2(s) = -2s^3 + 3s^2 \quad \text{and} \quad N_2'(s) = -6s^2 + 6s \\
N_3(s) = s^3 - 2s^2 + s \quad \text{and} \quad N_3'(s) = 3s^2 - 4s + 1 \\
N_4(s) = s^3 - s^2 \quad \text{and} \quad N_4'(s) = 3s^2 - 2s
\end{aligned}
\tag{2.31}
$$

The spline approach allows for a precise description of dislocation lines, particularly because the curvature is defined everywhere along the line. However, when integrations are to be performed over the lines, such as when computing stress fields (see equation (2.14)), closed form expressions can generally not be obtained. For instance, evaluating the length of a dislocation segment ij described with a cubic spline can be achieved by calculating:

$$
l_{ij} = \int_0^1 \left(\sum_{a,b=1}^{4} \vec{\boldsymbol{q}}_{ij}^{\,a} \cdot N_a'(s) N_b'(s) \cdot \vec{\boldsymbol{q}}_{ij}^{\,b} \right)^{1/2} ds \tag{2.32}
$$

Numerically, equation (2.32) can be evaluated using a Gaussian quadrature approximation, for instance. However, any numerical integration technique becomes very costly when the number of integration points becomes large, i.e. when seeking for a good approximation. Thus, in general, closed form solutions are favored in numerical simulations, principally as a way to alleviate the computational cost while ensuring an optimal accuracy. Conveniently, a linear interpolation can be used to describe

straight segments between dislocation nodes. Using the general interpolation scheme introduced in equation (2.28), straight segments are obtained by setting:

$$\begin{gathered} N_d = 2 \\ N_1(s) = 1 - s \quad \text{and} \quad N_2(s) = s \\ \vec{\boldsymbol{q}}_{ij}^{\,1} = \vec{\boldsymbol{x}}_i \quad \text{and} \quad \vec{\boldsymbol{q}}_{ij}^{\,2} = \vec{\boldsymbol{x}}_j \end{gathered} \tag{2.33}$$

Such approximation is perfectly appropriate as long as the discretization size (i.e. the average length of dislocation segments) remains sufficiently small with respect to line curvature. Moreover, analytical formulations for stresses and nodal forces can be derived for straight segments [35, 5], thereby allowing for a considerable gain in computation time and accuracy with respect to numerical integrations. With such description, any property at each point along a segment line can be deduced from the nodal properties values through the linear interpolation. For the sake of clarity, the position $\vec{\boldsymbol{x}}_{ij}(s)$ of a point located at curvilinear abscissa s along the dislocation segment ij is for instance obtained from equations (2.28) and (2.33) as:

$$\vec{\boldsymbol{x}}_{ij}(s) = (1-s)\vec{\boldsymbol{x}}_i + s\vec{\boldsymbol{x}}_j \quad \text{with} \quad 0 \leq s \leq 1 \tag{2.34}$$

Note that with this description, the curvature is not defined at every point along the dislocation line, since it is not uniquely defined at each node i. Conversely, the tangent vector defined in equation (2.29) is constant over the segment length and given by:

$$\vec{\boldsymbol{t}}_{ij} = \frac{\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i}{\|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\|} \tag{2.35}$$

$$l_{ij} = \|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\| \tag{2.36}$$

where l_{ij} denotes the length of dislocation segment ij.

2.3.2 Periodic boundary conditions

In DDD simulations, periodic boundary conditions (PBC) are often employed as they are particularly adapted to study plasticity in the bulk. The use of PBC includes two aspects: (1) the unit cell, also referred to as the primary volume, and its dislocation content are replicated in all spatial directions (x, y, z), and (2) all dislocations exiting a surface are periodically reintroduced through the opposite face with the same velocity, so as to ensure the conservation of dislocation fluxes. As depicted for two dimensions in figure 2.11, the use of PBC allows for circumventing the difficulties associated with the treatment of free external surfaces. When PBC are prescribed in the three spatial directions (x, y, z), the primary volume is replicated 26 times. The replicated volumes are referred to as image volumes while the replicated dislocations are referred to as image dislocations. Note that when PBC are applied, primary dislocations elastically interact with their own images, as well as with the images of all other dislocations present in the primary volume.

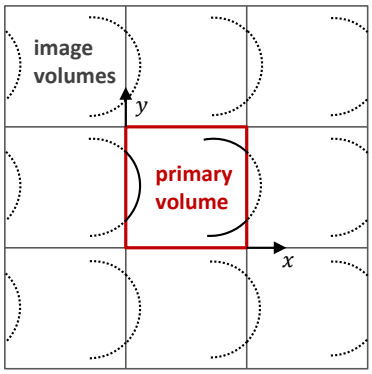


Figure 2.11: Schematic of 2D PBC in the (x, y) directions. The primary volume delimited in red is repeated in 8 times in all directions of the plane. The corresponding primary and image dislocations are drawn with solid and dashed lines, respectively. Dislocations that pass through one face of the primary volume are reintroduced on the opposite face by periodicity.

2.4 Boundary value problem in DDD

2.4.1 Mechanical boundary value problem

The framework used for the determination of the mechanical state in DDD simulations follows the fundamental boundary value problem of continuum mechanics. Essentially, the mechanical state at each point $\vec{\boldsymbol{x}}$ of the simulation volume V_s must respect the mechanical equilibrium (introduced in equation (2.2)) and satisfy the displacement and traction boundary conditions:

$$\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = \boldsymbol{C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}), \quad \forall \vec{\boldsymbol{x}} \in V_s \tag{2.37}$$

$$div(\boldsymbol{\sigma}) = 0 \tag{2.38}$$

$$\vec{\boldsymbol{u}} = \vec{\boldsymbol{u}}^{\,*} \text{ on } \partial V_u \tag{2.39}$$

$$\boldsymbol{\sigma} \cdot \vec{\boldsymbol{n}} = \vec{\boldsymbol{t}}^{*} \text{ on } \partial V_t \tag{2.40}$$

where $\boldsymbol{C}(\vec{\boldsymbol{x}})$ is the elastic stiffness tensor at point $\vec{\boldsymbol{x}}$, $\boldsymbol{\epsilon}^e$ is the elastic strain tensor, and ∂V_u and ∂V_t denote the external surfaces of volume V_s with normal $\vec{\boldsymbol{n}}$ subjected to displacement $\vec{\boldsymbol{u}}^{\,*}$ and traction $\vec{\boldsymbol{t}}^{*}$ boundary conditions, respectively.

Two main approaches have been developed in DDD simulations to solve for the boundary value problem expressed in equations (2.37) to (2.40). The first approach, which is currently widely used, consists in assuming that the medium V_s is elastic (outside of the dislocation cores). With this approach, the total stress $\boldsymbol{\sigma}$ is decomposed as the sum of different contributions that are added by virtue of the superposition principle [232]. In contrast, the second approach, the Discrete-Continuous Model (DCM), relies on an eigenstrain formalism in which the determination of the mechanical state follows an elasto-viscoplastic framework [147]. In the rest of this thesis, the method relying on the superposition principle will be referred to as the *regular* or *convential* DDD approach. This method, upon which most of current DDD simulations are grounded, is fully presented in the forthcoming sections. On the other hand, the

DCM approach will be fully presented in Section 3.2 as it constitutes the basis of the DDD-FFT method introduced in Chapter 3.

2.4.2 Superposition approach

In a vast majority of current conventional DDD simulations, the framework used to solve for the boundary value problem (2.37) to (2.40) follows the work of Van der Giessen and Needleman [232]. In this framework, the medium subjected to displacement and traction boundary conditions is considered to be linear elastic, outside of the dislocation cores. As a result, the initial complete problem is decomposed as two subsidiary problems that can be added making use of the superposition principle. As illustrated in figure 2.12, the original problem can be seen as the sum of a problem in which the dislocations are lying within an unbounded infinite elastic medium under no loading, and a problem in which the effect of the dislocations displacement and stress fields are removed from the original boundary conditions so as to define corrective boundary conditions. Following this decomposition, the total stress $\boldsymbol{\sigma}(\vec{\boldsymbol{x}})$ acting at each material point $\vec{\boldsymbol{x}}$ is given as the sum of (1) the internal stresses $\boldsymbol{\sigma}^{int}$ arising from the presence of dislocations in an infinite medium, and (2) external stress $\boldsymbol{\sigma}^{ext}$ coming from the imposed boundary conditions, such as to define:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{int} + \boldsymbol{\sigma}^{ext} \tag{2.41}$$

Depending on the loading and boundary conditions, the external contribution is also often referred to as the correction field required to ensure that the total stress $\boldsymbol{\sigma}$ respects the boundary conditions and the mechanical equilibrium stated in equations (2.37)–(2.40).

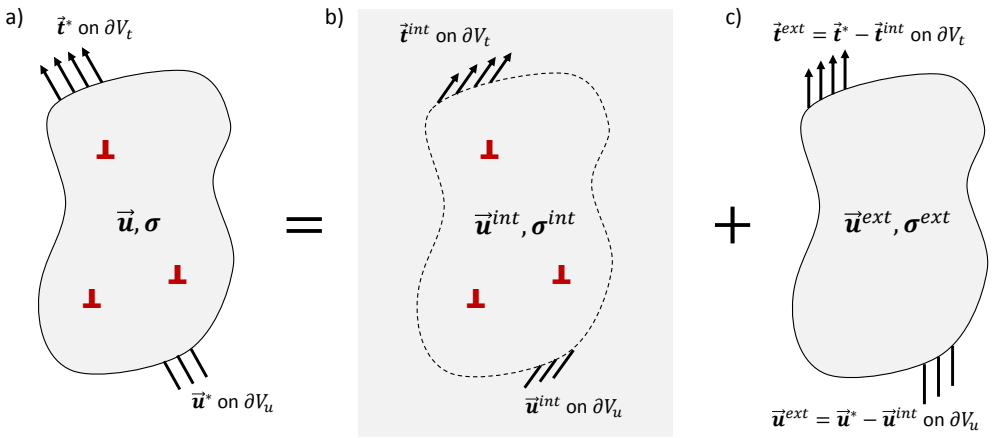


Figure 2.12: Decomposition of the boundary value problem as suggested in [232]. By considering the volume as linear elastic so as to make use of the superposition principle, the initial problem (a) defined through equations (2.37) to (2.40) can be decomposed as the superposition of two sub-problems: (b) a problem in which the dislocations are inserted into an infinite elastic medium under no load, and (c) a correction problem in which displacements and stresses induced by the dislocations are removed from the initial applied boundary conditions.

2.4.3 Internal stress contribution

In the regular approach, the internal stresses $\boldsymbol{\sigma}^{int}$ at each point $\vec{\boldsymbol{x}}$ in the volume are obtained by virtue of the superposition principle, whereby the contributions of the elastic stress field of each dislocation segment calculated in an infinite medium are added. As presented in Section 2.2.3, in its general form, the line integral expression of the stress induced by a dislocation loop with Burgers vector $\vec{\boldsymbol{b}}$ at any field point $\vec{\boldsymbol{x}}$ was obtained by Mura from the eigenstrain theory as [181]:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \oint_L e_{lnh} C_{pqmn} G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_m dx'_h \qquad \text{(2.14 repeated)}$$

where e_{ijk} is the permutation tensor, $G_{ij,k}$ is the derivative of the Green function, and $\vec{\boldsymbol{x'}}$ is the coordinate that spans the dislocation line L. As detailed in Section 2.2.2, this formulation stems from the boundary value problem (2.37) to (2.40) expressed in an infinite medium, i.e. for which no boundary conditions are prescribed, and in

which the stress results from the plastic distortion induced by the dislocation line L. Note however that equation (2.14) is singular along the dislocation line, such that the stress becomes unbounded. As a result, in the regular approach, the medium is considered elastic outside of the dislocation core. Note nonetheless that the non-singular formulation introduced by Cai and co-workers allows for a removal of the singularity by introducing a spreading of the Burgers vector [35] (see Appendix A). Following the discretization of dislocation lines used in DDD (see Section 2.3), equation (2.14) is generally decomposed as the sum of integrals over individual connected segments. As a result, the internal stress contribution $\boldsymbol{\sigma}^{int}$ is usually written as:

$$\boldsymbol{\sigma}^{int}(\vec{\boldsymbol{x}}) = \sum_{kl} \boldsymbol{\sigma}^{kl}(\vec{\boldsymbol{x}}) \tag{2.42}$$

where the sum is performed over all segments kl present in the volume and $\boldsymbol{\sigma}^{kl}$ denotes the stress field induced by segment kl at point $\vec{\boldsymbol{x}}$ obtained by equation (2.14). When PBC are prescribed, the summation also includes image segments pertaining to the replicated volumes. However, as indicated in (2.14), the classical theory provides an expression for the stress field of a dislocation line L when the latter is closed, i.e. when the integral is taken over a closed contour. In practice however, the dislocation network in regular DDD simulations is not enforced to be closed, and dislocations may intersect internal or external surfaces, if any. Using expression (2.14) to calculate the stress field of a dislocation intersecting a free surface will result in a violation of the mechanical equilibrium [154, 60]. Therefore, to ensure that the total stress remains divergence-free, the latter is corrected by the external contribution, when needed.

2.4.4 External stress contribution

In the literature, the external stress $\boldsymbol{\sigma}^{ext}$ contribution is often referred to as a correction field required for the total stress to satisfy the imposed loading conditions

[232, 253]. Particularly, this contribution is used to perform simulations in finite volume by imposing vanishing traction conditions on external surfaces such as to simulate the presence of free surfaces [253, 254]. Sometimes, this contribution is subdivided into two parts [77]:

$$\boldsymbol{\sigma}^{ext} = \boldsymbol{\sigma}^{u} + \boldsymbol{\sigma}^{t} \tag{2.43}$$

where $\boldsymbol{\sigma}^{u}$ pertains to the imposed displacement and $\boldsymbol{\sigma}^{t}$ to the surface traction. This decomposition is particularly apparent when PBC are used. In this case, the periodicity of the traction boundary conditions is ensured when contribution $\boldsymbol{\sigma}^{t}$ corresponds to the sum of the stress fields of image dislocations belonging to the replicated volumes [77]. In the general case however, $\boldsymbol{\sigma}^{ext}$ is determined by solving the boundary value problem illustrated in figure 2.12(c). The resolution of this problem requires the evaluation, at boundaries ∂V_t and ∂V_u, of internal stress and displacement fields induced by all dislocations in the volume, respectively. Such is done by using the superposition approach described in Section 2.4.3, whereby the stress can be calculated using equation (2.42). Using the non-singular formulation introduced by Cai et al., an analytical formulation for computing the internal stress field on a surface element has been recently developed [197]. However, special treatment for dislocations intersecting free surfaces must be included [254, 154, 252].

When PBC are used, a direct mean-field method can be used to solve for the external contribution. This is because in this case, simple displacement boundary conditions generate an homogeneous and uniform external stress throughout the simulation volume. As a result, the unique stress (respectively strain) value holding for the whole volume is readily obtained from the constitutive law (2.37), depending on the type of loading (a simulation can be alternatively driven in strain or stress, see Section 2.8.2 for complementary details). However, the direct method is only applicable to fully periodic simulations, and cannot be employed when boundary conditions

are complex or when elastic inhomogeneities are to be treated.

As an alternative to the mean-field method, the Finite Element Method (FEM) offers a full-field approach capable of overcoming the limitations of the direct method by providing a method to solve for the boundary value problem more rigorously. Expressing relations (2.37)–(2.40) in a weak form and applying the principle of virtual work for any arbitrary virtual displacement $\delta\vec{\boldsymbol{u}}$ yields the following expression:

$$\int_{V_s} \sigma_{ij}\delta u_{i,j} dV - \int_{\partial V_s} t_i \delta u_i dS = 0 \tag{2.44}$$

where ∂V_s denotes all external surfaces of simulation volume V_s. Consistently with relations (2.39)–(2.40), formulation (2.44) allows for applying both displacements and traction boundary conditions. Thus, $\vec{\boldsymbol{u}}^{ext}$ and $\boldsymbol{\sigma}^{ext}$ of the correction problem in figure 2.12(c) can be determined by discretizing simulation V_s into elements such as to numerically solve for (2.44). Full details on the FEM formulation and its implementation for linear elasticity are provided in Appendix E.

2.5 Motion of dislocations

2.5.1 Driving force and equation of motion

In metallic solids, dislocation slip constitutes one of the main plastic deformation modes. Slip corresponds to the generation and glide of dislocation lines, and occurs when local stress conditions are sufficient to activate dislocation propagation, as a way to accommodate the deformation (see Section 2.2.4). In other words, dislocation motion is mainly driven by the stress acting on dislocation lines. The stress contributions are either external, i.e. related to the imposed loading, or internal, i.e. induced by the presence of other defects within the lattice structure that give rise to internal displacement fields. As any crystallographic defect, dislocations contribute to the free energy of the crystal. As a result, a change in dislocation position through dislocation motion generally results in a change in the energy of the crystal. Thus,

from thermodynamical considerations, the driving force $\vec{\boldsymbol{f}}$ of dislocation motion is defined as the negative gradient of the energy E with respect to dislocation motion [73]:

$$\vec{\boldsymbol{f}} = -\frac{\partial E}{\partial \vec{\boldsymbol{x}}} = \vec{\boldsymbol{f}}^{pk} + \vec{\boldsymbol{f}}^{os} \tag{2.45}$$

where $\vec{\boldsymbol{f}}$ is the force per unit length. The general expression of the force $\vec{\boldsymbol{f}}$ on a dislocation includes a mechanical and a chemical contributions [73]. The former, corresponding to the well-known Peach-Koehler force, arises when the dislocation is subjected to a stress field $\boldsymbol{\sigma}$ and expresses as [111]:

$$\vec{\boldsymbol{f}}^{pk} = \left(\boldsymbol{\sigma} \cdot \vec{\boldsymbol{b}}\right) \times \vec{\boldsymbol{t}} \tag{2.46}$$

where $\vec{\boldsymbol{b}}$ denotes the Burgers vector corresponding to the elementary amount of displacement induced by the dislocation, and $\vec{\boldsymbol{t}}$ denotes its unit line tangent. The chemical force $\vec{\boldsymbol{f}}^{os}$, also referred to as the osmotic force, accounts for the presence of vacancies concentration and is given by the Bardeen-Herring equation [157, 73]. However, this contribution is usually neglected when modelling dislocation dynamics such that the following approximation is generally made:

$$\vec{\boldsymbol{f}} = -\frac{\partial E}{\partial \vec{\boldsymbol{x}}} \approx \vec{\boldsymbol{f}}^{pk} \tag{2.47}$$

Note however that a more general expression of the force per unit length practically used in DDD simulations such as to account for line tension and lattice friction stress will be given later in equation (2.55).

In response to a force, the motion of a dislocation differs depending on the material, the crystal structure, the slip system, the Burgers vector, the line orientation, etc. In a general setting, the relation between the force and the velocity along the dislocation line is modelled by a mobility function $\mathcal{M}$:

$$\vec{\boldsymbol{v}} = \mathcal{M}\left(\vec{\boldsymbol{f}}\right) \tag{2.48}$$

where $\vec{\boldsymbol{v}} = \frac{d\vec{\boldsymbol{x}}}{dt}$ is given as the time derivative of dislocation positions $\vec{\boldsymbol{x}}$. The mobility function $\mathcal{M}$ is chosen such that dislocation motion reproduces well experimental observations and atomistic simulations. In particular, the motion of dislocations in elastic crystals is generally approximated by an overdamped equation of motion mimicking the viscous drag arising from phonons interactions with moving dislocations. With such description, and neglecting the effects of inertia, equation (2.48) simply writes as a linear expression:

$$\boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}} = \vec{\boldsymbol{f}} \tag{2.49}$$

where $\boldsymbol{\mathcal{B}}$ denotes the viscous drag coefficient matrix. Details for the case where inertia is accounted for are given in Appendix C.1.1. Note that more complex mobility functions may be required to better reflect dislocations motion [36, 176]. For instance, sophisticated phenomenological approaches, such as those based on Arrhenius-type laws, have been developed to reproduce the kink-pair mechanism controlling the screw mobility in HCP crystals [176].

2.5.2 Determination of nodal dislocation motion

The linear equation of motion (EOM) (2.49) expresses the general relation between dislocation force and velocity from thermodynamical considerations. Using a variational approach, a weak formulation can be obtained to solve for dislocation motion when the dislocation network is discretized into segments. From equation (2.49), the EOM relating the force $\vec{\boldsymbol{f}}$ and the velocity $\vec{\boldsymbol{v}}$ at each position along a closed dislocation loop can be expressed as [97]:

$$\oint \delta\vec{\boldsymbol{x}} \cdot \left(\vec{\boldsymbol{f}} - \boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}}\right) |d\vec{\boldsymbol{x}}| = 0 \tag{2.50}$$

where $\delta\vec{\boldsymbol{x}}$ is a virtual displacement of the dislocation line and the integration is performed along the entire dislocation line. Using the segment parametrizations described in Section 2.3.1, a finite element approach based on the weak formulation (2.50) can be used to evaluate motion of the discrete dislocation network. Full details on the Garlekin variational method to discrete dislocation motion are provided in Appendix C. As a result, the determination of nodal velocities associated with dislocation segments at each time step is achieved by solving the system of equations defined by:

$$\vec{\boldsymbol{F}}_i = \sum_j \boldsymbol{B}_{ij}\vec{\boldsymbol{V}}_j \qquad \begin{array}{l} \forall \text{ node } i, \\ \forall \text{ node } j \text{ connected to node } i \text{ including } i = j \end{array} \tag{2.51}$$

where $\vec{\boldsymbol{V}}_j$ is the velocity of dislocation node j to solve for, $\vec{\boldsymbol{F}}_i$ is the force at node i, and $\boldsymbol{B}_{ij}$ is the resistivity matrix pertaining to the viscous drag of segment ij. Using the classical finite element approach, the forces $\vec{\boldsymbol{F}}_i$ at each dislocation node i required in system of equations (2.51) are assembled as:

$$\vec{\boldsymbol{F}}_i = \sum_j \vec{\boldsymbol{f}}_{ij} \tag{2.52}$$

where $\vec{\boldsymbol{f}}_{ij}$, denoting the force on segment ij acting at node i, is obtained by integration of the total force $\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}})$ along the segment length $|d\vec{\boldsymbol{x}}|$ as:

$$\vec{\boldsymbol{f}}_{ij} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}})\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}})\,|d\vec{\boldsymbol{x}}| \tag{2.53}$$

where $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ denote the positions of end nodes i and j delimiting segment ij, and $N_i(\vec{\boldsymbol{x}})$ is the interpolation function associated with dislocation node i. When using the segment linear interpolation introduced in relations (2.33) to map dislocation segments as straight lines, equation (2.53) reduces to:

$$\vec{\boldsymbol{f}}_{ij} = l_{ij} \int_0^1 (1-s)\vec{\boldsymbol{f}}\left(\vec{\boldsymbol{x}}_{ij}(s)\right) ds \tag{2.54}$$

where $l_{ij} = \|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\|$ denotes the length of segment ij, and $\vec{\boldsymbol{x}}_{ij}(s)$ defined in equation (2.34) is the straight line mapping dislocation segment ij with curvilinear abscissa $s \in [0,1]$. In order to account for the lattice friction resistance, the effective force $\vec{\boldsymbol{f}}_{ij}$ is practically obtained as (see Appendix C.2):

$$\vec{\boldsymbol{f}}_{ij} = \begin{cases} \vec{\boldsymbol{f}}_{ij}^{pk} - \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \cdot \left(\frac{\vec{\boldsymbol{f}}_{ij}^{pk}}{\|\vec{\boldsymbol{f}}_{ij}^{pk}\|}\right) & \text{if } \left|\vec{\boldsymbol{f}}_{ij}^{pk}\right| > \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \\ 0 & \text{if } \left|\vec{\boldsymbol{f}}_{ij}^{pk}\right| \leq \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \end{cases} \tag{2.55}$$

where $\vec{\boldsymbol{f}}_{ij}^{fric}$ is the force arising from the lattice friction resistance, which can be readily obtained by integration of the lattice friction stress over the dislocation segment length, and $\vec{\boldsymbol{f}}_{ij}^{pk}$ is the Peach-Koehler force acting at node i obtained from integration of $\vec{\boldsymbol{f}}^{pk}$ in expression (2.46) as:

$$\vec{\boldsymbol{f}}_{ij}^{pk} = l_{ij} \int_0^1 (1-s)\vec{\boldsymbol{f}}^{pk}\left(\vec{\boldsymbol{x}}_{ij}(s)\right) ds \tag{2.56}$$

From the expression of the Peach-Koehler force per unit length in (2.46) and the definition of the stress state using the superposition approach in expression (2.41), $\vec{\boldsymbol{f}}_{ij}^{pk}$ in equations (2.55) and (2.56) can be decomposed as:

$$\vec{\boldsymbol{f}}_{ij}^{pk} = \vec{\boldsymbol{f}}_{ij}^{int} + \vec{\boldsymbol{f}}_{ij}^{ext} \tag{2.57}$$

where the force $\vec{\boldsymbol{f}}_{ij}^{ext}$ arising from a uniform external applied stress $\boldsymbol{\sigma}^{ext}$ is given by (see details in Appendix C.2.4):

$$\vec{\boldsymbol{f}}_{ij}^{ext} = \frac{1}{2} l_{ij} \left(\left[\boldsymbol{\sigma}^{ext} \cdot \vec{\boldsymbol{b}}_{ij}\right] \times \vec{\boldsymbol{t}}_{ij} \right) \tag{2.58}$$

where $\vec{\boldsymbol{b}}_{ij}$ and $\vec{\boldsymbol{t}}_{ij}$ denote the Burgers vector and the unit tangent of segment ij. In

contrast, $\vec{\boldsymbol{f}}_{ij}^{int}$ corresponds to the force arising from the internal stress fields induced by all dislocations in the medium. Following the framework introduced in Section 2.4.3 whereby the contributions of the elastic stress fields of individual dislocation segments are added by virtue of the superposition principle, the internal force $\vec{\boldsymbol{f}}_{ij}^{int}$ can be expressed as a sum over all dislocation segments kl in the volume as:

$$\vec{\boldsymbol{f}}_{ij}^{int} = \vec{\boldsymbol{f}}_{ij}^{s} + \sum_{kl \neq ij} \vec{\boldsymbol{f}}_{ij}^{kl} \tag{2.59}$$

where $\vec{\boldsymbol{f}}_{ij}^{s}$ denotes the self-force for which details can be found in Appendix C.2.3, and where the force $\vec{\boldsymbol{f}}_{ij}^{kl}$ on segment ij due to the elastic interaction with segment kl is calculated for every segment kl distinct from ij.

As a result, the computation of all nodal forces is a very expensive $\mathcal{O}(N_{seg})$ process where N_{seg} is the total number of dislocation segments in the simulation volume, that constitutes a bottleneck in regular DDD simulations. Thus, special care needs to be taken so as to implement efficient methods for their computation. While segment-segment elastic interactions forces $\vec{\boldsymbol{f}}_{ij}^{kl}$ will be detailed in Section 2.5.3, approximations to improve the efficiency of their computation will be presented in Section 2.9.2.

Once the nodal forces have been computed, the resistivity matrix $\boldsymbol{B}_{ij}$ in system (2.51) is assembled and computed for each segment ij as described in Appendix C.3. After simplification and considering straight segments, system of equations (2.51) reduces to the following system:

$$\vec{\boldsymbol{F}}_i = \sum_{j \neq i} \frac{l_{ij}}{6} \boldsymbol{\mathcal{B}}(\theta_{ij})(2\vec{\boldsymbol{V}}_i + \vec{\boldsymbol{V}}_j) \tag{2.60}$$

where l_{ij} denotes the length of dislocation segment ij. Assuming the viscous drag exerted along a dislocation line continuously varies as a function of its character, a simple expression for the drag matrix $\boldsymbol{\mathcal{B}}(\theta_{ij})$ can be obtained as:

$$\boldsymbol{\mathcal{B}}(\theta_{ij}) = \left(\mathcal{B}^e_{ij} \sin^2 \theta_{ij} + \mathcal{B}^s_{ij} \cos^2 \theta_{ij}\right) \boldsymbol{I}_2 \tag{2.61}$$

where $\mathcal{B}^e_{ij}$ and $\mathcal{B}^s_{ij}$ are the edge and screw mobilities associated with the slip system of segment ij, and $\boldsymbol{I}_2$ is the second-order identity tensor.

With this approach, the velocity $\vec{\boldsymbol{V}}_i$ of each node i is determined by solving the system of equations given in (2.60). Note that the resulting velocities must be further projected on the glide planes of dislocation segments to ensure that glide of dislocations takes place in their slip planes (see Appendix C.3.2) and that a special treatment must be implement for junction segments (see Appendix C.3.3).

Finally, the new position $\vec{\boldsymbol{x}}_i^{\,t+\Delta t}$ of each dislocation node i at time $t + \Delta t$ can be obtained by time integration of velocities $\vec{\boldsymbol{V}}_i$. For instance, when an implicit backward Euler time integrator is used, the new positions are obtained from:

$$\vec{\boldsymbol{x}}_i^{\,t+\Delta t} = \vec{\boldsymbol{x}}_i^{\,t} + \vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right) \Delta t \tag{2.62}$$

where $\vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right)$ denote the nodal velocities obtained from system (2.60) as a function of nodal positions $\vec{\boldsymbol{x}}_i^{\,t}$ at time step t, and Δt is the time step increment. When using a more robust iterative time integration scheme, the forces and velocities need to be computed from (2.60) at each iteration as described in Appendix C.4.

At each time increment, once the new positions of the nodes have been computed, intersections between segments are considered (collisions, junction formations, node dissociations, etc.) and dislocation lines are dynamically remeshed. While the remeshing process is presented in Section 2.7, the treatment of dislocation interactions is discussed in Section 2.6.

2.5.3 Segment-segment elastic interactions

The computation of segment forces pertaining to internal stresses as presented in equation (2.59) has a computational complexity of $\mathcal{O}(N^2_{seg})$ where N_{seg} is the number

of dislocation segments in the simulation volume: the evaluation of $\vec{\boldsymbol{f}}_{ij}^{int}$ for each segment ij requires the computation of the elastic interaction force $\vec{\boldsymbol{f}}_{ij}^{kl}$ induced by every other segment kl. From a physical standpoint, this results from the fact that the presence of a dislocation line corresponds to the displacement of a certain number of atoms, thereby inducing a stress field throughout the elastic solid (see Section 2.2). As a result, each dislocation elastically interacts with all other dislocations present in the medium through the stress field that each of them produces. From a modelling standpoint, this translates by the fact that (1) the evaluation of all interaction forces $\vec{\boldsymbol{f}}_{ij}^{kl}$ is a function of the segment positions and needs to be calculated at every time step, and (2) the computational cost associated with their evaluation constitutes a bottleneck in regular DDD simulations. Thus, it becomes very expensive when dealing with large numbers of segments, especially considering that one of the particularities of DDD simulations lies in the colossal multiplication of dislocation segments during simulations (e.g. Frank-Read source mechanism). Besides, as detailed in Section 2.3.2, PBC are often in employed in DDD simulations to study plasticity in the bulk. In such case, the stress fields pertaining to the image dislocation segments belonging to the replicated volumes have to be further accounted in the evaluation of the internal stress state $\boldsymbol{\sigma}^{int}$, increasing thereby the computational burden by a factor 26. For these reasons, although approximation and optimization procedures have been developed and permit significant savings in computation time (see Section 2.9.2), dislocation-dislocation elastic interactions yet remain the most intensive procedure in regular DDD approaches, and particular care must be taken in the implementation of their numerical evaluation.

From expression (2.53) and the definition of the Peach-Kohler force in equation (2.46), the force $\vec{\boldsymbol{f}}_{ij}^{kl}$ acting at node i induced by the stress field of dislocation segment kl on segment ij is expressed as:

$$\vec{\boldsymbol{f}}_{ij}^{kl} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\boldsymbol{\sigma}^{kl}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| \tag{2.63}$$

where $\boldsymbol{\sigma}^{kl}(\vec{\boldsymbol{x}})$ denotes the stress field induced by dislocation segment kl at field point $\vec{\boldsymbol{x}}$, whose components are given in equation (2.14). Using the linear parametric segment representation introduced in relations (2.33), equation (2.63) reads:

$$\begin{aligned} \vec{\boldsymbol{f}}_{ij}^{kl} &= l_{ij} \int_0^1 (1-s) \left[\left(\boldsymbol{\sigma}^{kl}\left(\vec{\boldsymbol{x}}_{ij}(s)\right) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] ds \\ \text{or} \quad \vec{\boldsymbol{f}}_{ij}^{kl} &= l_{ji} \int_0^1 s \left[\left(\boldsymbol{\sigma}^{kl}\left(\vec{\boldsymbol{x}}_{ji}(s)\right) \cdot \vec{\boldsymbol{b}}_{ji} \right) \times \vec{\boldsymbol{t}}_{ji} \right] ds \end{aligned} \tag{2.64}$$

As mentioned in Section 2.2.3, the stress field induced by a dislocation becomes singular in its core when using the classical theory (2.14). To tackle this issue and ensure that integral (2.64) remains finite, several approaches have been adopted. For instance, the use of a cut-off radius has been one of the common techniques. However, the arbitrary choice for the cut-off value renders such approach more numerical than physical. Recently, Cai and co-workers proposed a more consistent non-singular approach for isotropic elasticity [35]. Besides removing the singularity in the dislocation core, this theory has also permitted to obtain fully analytical expressions of (2.64), allowing for an efficient numerical evaluation of isotropic segment-segment interactions $\vec{\boldsymbol{f}}_{ij}^{kl}$. While details of the non-singular approach are presented in Appendix A.2, the full analytical expressions of integral (2.64) for parallel and non-parallel interacting segments in isotropic elasticity are provided in reference [5].

In general anisotropic elasticity, no analytical formulation for (2.64) can be found since no closed-form solution exists for the Green's function in this case. The evaluation of interaction forces therefore requires the calculation of nested numerical integrations using expressions (2.23), (2.14) and (2.64) (see Section 2.2.3). Such direct approach has been reported to be extremely computationally prohibitive, inducing a cost ranging between 200 and 500 times that of isotropic elasticity [202, 108, 40].

Thus, to alleviate this computational burden, different anisotropic approaches have been developed, such as that based on the Willis-Steeds-Lothe formalism [264]. Nevertheless, the most important improvement was recently allowed by the development of an approach based on expansions in spherical harmonics [12], permitting significant savings in the computational cost. Details on the spherical harmonics approach for anisotropic media are provided in Appendix B.2.

2.6 Dislocation core-reactions and dissociations

During deformation, dislocation segments are likely to interact with one another as they propagate. Such core-dislocation reactions require a proper treatment as they constitute a very important feature of DDD simulations. Specifically, the formation of junctions that occur upon intersection of dislocation lines via the zipping, unzipping, and annihilation mechanisms, have been identified as crucial to accurately model strain hardening [31, 5]. In practice, there are several ways in which such reactions can be treated. The simplest way is to assume that dislocation-dislocation reactions are governed solely by elastic interactions. For instance, this method was used in [39] to compute the strength of junctions for pairs of dislocations in Mg. With this method, no additional local rules need to be implemented as junctions form by elastic attraction, and intersections of dislocation lines do not induce topological modifications of the dislocation network. Annihilations, however, cannot be treated directly with this method since it requires segments to be removed from the simulation. Another disadvantage of this method is that it requires extremely small time step increments and very precise calculations of the stress fields in order for dislocations to attain stable and realistic configurations while intersecting. Because this method is computationally intensive, it is generally not suitable for massive simulations, or when a large amount of plastic strain is desired [5]. To overcome these limitations, an approach in

which the formation of junction is treated explicitly is generally used in DDD simulations. Such approach is implemented via the introduction of local intersection rules and induces topological changes in the dislocation discretized network. While the explicit intersection approach implemented in this work is presented in Section 2.6.1, the topological operations implemented in the DDD simulation to support numerical topological changes are fully presented in Appendix D.1.1.

2.6.1 Junctions formation and annihilations

The dislocation-dislocation interaction local rules implemented in the DDD code should allow for junction formation, junction unzipping and annihilation operations. Therefore, the collision procedure implemented in the DDD simulation is as follows: as illustrated in figures 2.13(a) and 2.13(b), a junction node is inserted when two dislocation segments are within a predefined capture radius d_{crit} with respect to each other. Note however that an additional criterion on dislocation velocities is used, such that the creation of a junction node only occurs when both dislocation segments are moving towards one another, i.e. only in the case where a collision between both dislocations is predicted. To conserve the properties of the initial dislocation segments, the intersection node is constrained to lie on the intersection of the planes of the two intersecting dislocations. It is usually placed along this line at a position minimizing the distance from each initial segment. As a result, the two initial dislocation segments of interest are split and get pinned through a 4-connected junction node.

As depicted in figures 2.13(b) to 2.13(c), a junction segment is able to form following a "zipping" process: if two of the four arms of the junction node lie within their mutual capture radius, a new junction node is inserted following the same process as that described above. Consequently, the two junction nodes are connected by a junction segment, whose resulting Burgers vector is the sum of the Burgers vectors of the initial intersecting dislocation segments. Thus, the junction is glissile if its

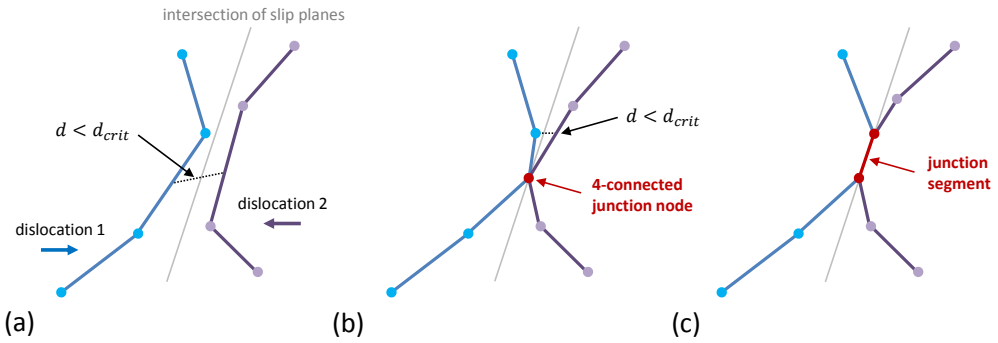


Figure 2.13: Schematic of dislocation-dislocation reaction: (a) when the distance between two dislocation segments moving towards one another is smaller than a critical capture radius d_{crit}, (b) a junction node is inserted at the intersection of the slip planes of both dislocations. (c) The repetition of process (a) to (b) between the arms of the junction node leads to the formation of a junction segment. The passage from steps (b) to (c) corresponds to the zipping process.

Burgers vector lies within one of the existing slip planes of the crystal – assuming motion is possible on this slip system –, sessile otherwise. A special situation occurs when two interacting dislocations have opposite Burgers vectors: in this case, the resulting junction possesses a null Burgers vector, such that the junction segment can be removed from the simulation. This case corresponds to the annihilation process. Therefore, the advantage of this approach is that it makes no distinction between junction formation and annihilation in the dislocation reaction treatment. In this process, one has to ensure that the formation of the new junction leads to a minimization of the strain energy [111]. This is done by checking whether the dissipation induced by the new junction is greater than that of the initial configuration. If not, the junction is not formed, and the interaction remains purely elastic. Such step is usually performed through the dissociation procedure presented in Section 2.6.2.

The unzipping of a junction is also handled by the DDD code through the following mechanism. As discussed in Appendix C.3.3, when a sessile junction is formed, its two end nodes are restricted to move along the junction direction only, i.e. along the intersection of the two planes of the dislocations that formed it. Therefore, unzipping

of a junction is implicitly handled by the remeshing procedure (see Section 2.7 for more details): when two nodes become too close to one another, they merge into a single node. As a result, the junction unzips. However, this unzipping mechanism functions until the junction is left with a single junction node, in which case it can no longer operate. Therefore, junction unzipping is also governed by the dissociation procedure presented in Section 2.6.2. Full details on the numerical implementation of junction formation are provided in Appendix D.1.2.

2.6.2 Dissociation procedure

As shown in figure 2.13, intersections of dislocation segments lead to the creation of 4-connected nodes. Those are of special interest as they can physically represent either the nucleation or the destruction of a junction, a crossed state or a repulsive state. To distinguish between these options, each 4-connected node is dissociated into 2 nodes, leading to the formation of an artificial junction segment connecting them. For the sake of comprehension, the sequence of operations associated with the dissociation of a 4-connected node is similar to that depicted in going from configuration in figure 2.13(b) to that in figure 2.13(c). For each node with four arms, three dissociation configurations are possible, given the different possibilities of arm connections (see Appendix D.1.3 for more details). The dissipation induced by each of these configurations is computed, and, as detailed in [5], the configuration inducing the greatest dissipation is kept. Thus, among every configuration jk corresponding to the dissociation of a 4-connected dislocation node i into two nodes j and k, the most favorable configuration is that inducing the maximum dissipation, i.e. the one that yields:

$$\max_{jk}\left(\mathcal{P}_{jk}^{diss}\right) = \max_{jk}\left(\vec{\boldsymbol{F}}_j \cdot \vec{\boldsymbol{V}}_j + \vec{\boldsymbol{F}}_k \cdot \vec{\boldsymbol{V}}_k\right) \tag{2.65}$$

where $\mathcal{P}_{jk}^{diss}$ is the dissipation induced by configuration jk and scalar quantity $\vec{\boldsymbol{F}}_j \cdot \vec{\boldsymbol{V}}_j$

is the contribution of node j to the total dissipation, where $\vec{\boldsymbol{F}}_j$ and $\vec{\boldsymbol{V}}_j$ are the nodal forces and velocities, respectively. If the dissipation induced by the original configuration is the most favorable, i.e. $\mathcal{P}_i^{diss} = \vec{\boldsymbol{F}}_i \cdot \vec{\boldsymbol{V}}_i > \max_{jk}\left(\mathcal{P}_{jk}^{diss}\right)$, the 4-connected node i is left undissociated, which, in that case, corresponds to a crossed or repulsive state. Full details on the numerical implementation of the dissociation procedure are provided in Appendix D.1.3.

2.7 Adaptive dislocation meshing

During DDD simulations, the dislocation configuration drastically evolves as a result of line increase via glide (e.g. Frank-Read source mechanism) and dislocation intersections. Therefore, a dynamical discretization of dislocation lines must be performed to maintain the dislocation network well discretized. In practice, the adaptive line remeshing procedure must prevent the occurrence of two problematic situations. First, it must be ensured that connected dislocation nodes are not too far apart, i.e. that dislocation segments do not become too long. Long dislocation segments inevitably lead to strong numerical inaccuracies in the overall dislocations behavior, especially because the curvature cannot be precisely defined (when using straight segments), and because it restricts the number of degrees of freedom for dislocation motion. Alternatively, it must also be ensured that dislocation segments are not too short. Practically, short segments induce harmful numerical vibrations that may bias the entire simulation behavior and lead to the collapse of the simulation, or require excessively small time step increments (small time step increments are also needed for short segments in order not to miss segment-segment intersections). For such reason, a remeshing procedure is performed on all dislocation segments at each time step. The two different remeshing approaches that have been implemented are presented below.

2.7.1 Segment remeshing

The simplest efficient remeshing approach preventing the rise of segments of problematic length (i.e. either too short or too long) consists in maintaining the length of every segment close to an average predefined value l_{avg}. This can be achieved by systematically merging close connected nodes, thereby removing short segments, and splitting long segments into shorter ones by inserting new nodes. Both these operations are performed thanks to the topological procedures introduced in Appendix D.1.1. Obviously, the use of this approach requires a wise choice for the average length l_{avg} of dislocation segments, that depends on the size of the simulation, the dislocation configuration, the time step increments, etc. Thus, dislocation segments whose length are greater than l_{avg} are split. Further, for the sake of consistency, connected nodes are merged when segments are shorter than $\frac{1}{4}l_{avg}$. This is to avoid that newly split segments immediately trigger the merge of their end nodes, thereby restoring the initial configuration.

2.7.2 Spline remeshing

Although the simple segment remeshing method presented in the above paragraph is suitable for most DDD simulations, it might become unsatisfactory for some specific applications or configurations. The two main limitations associated with this method are the following: the dislocation network ends up being unevenly discretized, i.e. the maximum relative length between dislocation segments typically reaches 2, and the curvature is not necessarily preserved as straight dislocation segments are split along their line. To overcome these limitations, a spline remeshing procedure has been implemented in the DDD code. The core idea of this approach relies on interpolating dislocation lines with a cubic spline whose control points are the dislocation nodes. Note that such interpolation is uniquely performed during the remeshing step, and follows the cubic spline parametrization introduced in Section 2.3.1. With this,

dislocation nodes can be moved along the spline such as to remain equally spaced and respect the curvature defined through the spline interpolation.

2.8 Mechanical response: stress and strain calculation

2.8.1 Plastic strain calculation

As discussed in Section 2.2.4, the motion of dislocations shears the crystal. From a physical standpoint, the shear induced by the glide of a dislocation line corresponds to the product between the magnitude of its Burgers vector and the area swept by the dislocation over the sheared volume V, as defined in equation (2.24):

$$\gamma = \frac{bA}{V} \qquad \text{(2.24 repeated)}$$

where $b = \|\vec{\boldsymbol{b}}\|$ is the magnitude of the Burgers vector, A is the area swept by the dislocation during glide, and V is the volume of the sheared crystal. Extrapolating this definition to a discrete dislocation representation, the total increment of shear $d\gamma^s$ induced by the collective movement of dislocation segments on system s can be evaluated as:

$$d\gamma^s = \frac{b^s \sum_{ij \in s} dA_{ij}}{V} \tag{2.66}$$

where V is the primary simulation volume, b^s is the magnitude of the Burgers vector of system s, and dA_{ij} corresponds to the incremental area swept by each dislocation segment ij of the primary volume belonging to system s. The determination of area dA_{ij} is obtained by integrating the segment displacement over its length, i.e.:

$$dA_{ij} = \int_0^1 \vec{\boldsymbol{V}}_{ij}(s) dt \cdot ds \tag{2.67}$$

where $\vec{\boldsymbol{V}}_{ij}(s)$ is the velocity at curvilinear abscissa $s \in [0, 1]$ along segment ij interpolated from nodal velocities $\vec{\boldsymbol{V}}_i$ and $\vec{\boldsymbol{V}}_j$, and dt is the time step increment. When using

a linear interpolation, the area swept by each dislocation segment forms a quadrilateral, such that the integration can be circumvented, and the incremental area can be analytically computed using the quadrilateral area formula:

$$dA_{ij} = \frac{1}{2} \| \left(\vec{\boldsymbol{x}}_j - \left(\vec{\boldsymbol{x}}_i - \vec{\boldsymbol{V}}_i dt \right) \right) \times \left(\vec{\boldsymbol{x}}_i - \left(\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{V}}_j dt \right) \right) \| \tag{2.68}$$

where $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ denote the initial positions of nodes i and j before motion has occurred, and operator $\times$ is the vectorial cross product. With this, the total plastic strain increment is computed from equation (2.26) as:

$$\widetilde{d\epsilon_{ij}^p} = \frac{1}{2} \sum_s \left(b_i^s n_j^s + n_i^s b_j^s \right) d\gamma^s \tag{2.69}$$

where b_i^s and n_i^s are the components of the unit Burgers vector and the unit normal of the plane of slip system s, and where the summation is performed over all slip systems. Thus, expression (2.69) provides a discrete quantification of the homogenized plastic strain tensor produced by the motion of the ensemble of dislocation segments present in the simulation volume during a time step increment.

2.8.2 Homogenized stress and strain response

In practice, the homogenized stress-strain $\boldsymbol{\sigma} - \boldsymbol{\epsilon}$ relation holding within the simulation volume is given by the classical three-dimensional Hooke's law:

$$\widetilde{\boldsymbol{\sigma}} = \boldsymbol{C} : \left(\widetilde{\boldsymbol{\epsilon}} - \widetilde{\boldsymbol{\epsilon}^p} \right) \tag{2.70}$$

where $\boldsymbol{C}$ denotes the forth-order elastic stiffness tensor, $\widetilde{\boldsymbol{\sigma}}$ is the stress tensor, and $\widetilde{\boldsymbol{\epsilon}}$ and $\widetilde{\boldsymbol{\epsilon}^p}$ are the total and plastic strain tensors, respectively. Since the plastic strain is solely computed over the whole volume in the regular approach (see Section 2.8.1), the quantities in equation (2.70) are therefore necessarily considered as uniform and homogeneous throughout the primary simulation volume, hence the use of symbol $\widetilde{\cdot}$

indicating homogenized quantities. Depending on the type of boundary conditions with which the simulation is driven, different schemes have been implemented to update the stress-strain response in the incremental time framework. Thus, when driving the simulation with a strain rate $\dot{\boldsymbol{\epsilon}}$, the total strain increment $d\widetilde{\boldsymbol{\epsilon}} = \dot{\boldsymbol{\epsilon}}dt$ is imposed at each time step $t+dt$ such that the stress $\widetilde{\boldsymbol{\sigma}}$ can be incrementally computed from (2.70) as:

$$\begin{aligned} \widetilde{\boldsymbol{\epsilon}}^{\,t+dt} &= \widetilde{\boldsymbol{\epsilon}}^{\,t} + d\widetilde{\boldsymbol{\epsilon}} = \widetilde{\boldsymbol{\epsilon}}^{\,t} + \dot{\boldsymbol{\epsilon}}dt \\ d\widetilde{\boldsymbol{\sigma}} &= \boldsymbol{C} : \left(d\widetilde{\boldsymbol{\epsilon}} - d\widetilde{\boldsymbol{\epsilon}^p}\right) \\ \widetilde{\boldsymbol{\sigma}}^{\,t+dt} &= \widetilde{\boldsymbol{\sigma}}^{\,t} + d\widetilde{\boldsymbol{\sigma}} \end{aligned} \tag{2.71}$$

where $\widetilde{\boldsymbol{\sigma}}^{\,t}$ and $\widetilde{\boldsymbol{\epsilon}}^{\,t}$ denote the previous stress and strain state at time t, respectively, and $d\widetilde{\boldsymbol{\epsilon}^p}$ is given by formulation (2.69). When driving the simulation with stress, a stress increment $d\widetilde{\boldsymbol{\sigma}} = \dot{\boldsymbol{\sigma}}dt$ is imposed at each time step $t + dt$ such that the new strain $\widetilde{\boldsymbol{\epsilon}}$ can be incrementally computed as:

$$\begin{aligned} \widetilde{\boldsymbol{\sigma}}^{\,t+dt} &= \widetilde{\boldsymbol{\sigma}}^{\,t} + d\widetilde{\boldsymbol{\sigma}} = \widetilde{\boldsymbol{\sigma}}^{\,t} + \dot{\boldsymbol{\sigma}}dt \\ d\widetilde{\boldsymbol{\epsilon}} &= \boldsymbol{C}^{-1} : d\widetilde{\boldsymbol{\sigma}} + d\widetilde{\boldsymbol{\epsilon}^p} \\ \widetilde{\boldsymbol{\epsilon}}^{\,t+dt} &= \widetilde{\boldsymbol{\epsilon}}^{\,t} + d\widetilde{\boldsymbol{\epsilon}} \end{aligned} \tag{2.72}$$

Thus, with schemes (2.71) and (2.72), macroscopic stress-strain curves corresponding to the mechanical response of the simulated crystal can be obtained.

2.9 Numerical implementation and optimizations

2.9.1 Development of the DDD code

Following the general procedure presented in Section 2.1.1, the development of a DDD code requires the numerical implementation of the different stages of the DDD cycle presented in figure 2.1 and detailed in all previous sections. For such purpose, a code

written in Fortran 90 has been developed during this work to incorporate all features mentioned through Sections 2.1 to 2.8. During the development, particular care has been taken to ensure that the code remains modular, i.e. allows for the addition of supplementary features in an easy fashion, while being robust and computationally efficient. Considering the prohibitive computational time associated with DDD simulations – essentially induced by the costly calculation of segment-segment elastic interactions and the significant multiplication of dislocation segments during strain hardening – particular care and efforts have been dedicated to develop an efficient parallel implementation. While the strategy adopted for the parallel implementation is summarized in Section 2.9.3, some of the various optimizations that have been incorporated to the DDD code are presented in Section 2.9.2.

2.9.2 Approximations

2.9.2.1 Long-range elastic interactions

As detailed in Section 2.5.3, the calculation of segment-segment elastic interaction forces is the most intensive computational task in DDD simulations. Since every dislocation segment interacts with every other segment present in the volume, this computational process scales with $\mathcal{O}(N_{seg}^2)$ where N_{seg} is the total number of segments in the simulation. Furthermore, the number of operations (multiplications, additions, logarithmic and sinusoidal functions) involved in the analytical computation of a single isotropic elastic interaction force amounts to about 1500 [12]. Consequently, the complexity associated with a full $\mathcal{O}(N_{seg}^2)$ calculation would preclude any relevant simulation in terms of achievable level of strain or dislocation density. For such reason, different approximation methods have been developed to alleviate the overall computational cost of elastic interactions. The two principal methods that are currently used take advantage of the physical decay in $1/R$ of the stress field induced by a dislocation, where R is the distance between the dislocation line and

the field point (see equations (2.14) and (2.17)). Thus, a distinction can be introduced between short-range and long-range elastic interactions, allowing to resort to approximated methods to evaluate the forces associated with long-range interactions, while accurately computing that for short-range interactions. Such distinction was first introduced via the Box Method [240], and such method is implemented in the DDD code. As depicted in two dimensions in figure 2.14, the Box Method regularly partitions the primary volume into a fixed total number of boxes N_{box}. With this, each dislocation segment ij is contained within a box α, and the union U_α of the 26 surrounding boxes and the box α itself defines the neighbor boxes of segment ij. With this, the ensemble of segments contained in U_α defines the short-range interactions of segment ij, while that outside of U_α are considered as long-range interactions. Following this partitioning, the stress fields associated with long-range interactions are approximated by computing them at the center $\vec{\boldsymbol{x}}_\alpha$ of each box α. Practically, the saving in computational cost results from the evaluation of a single long-range interaction force for all segments belonging to the same box. Therefore, the global complexity $\mathcal{C}_{box}$ of this approach can be estimated as:

$$\mathcal{C}_{box} \propto \underbrace{N_{box} \times \left(27 \frac{N_{seg}}{N_{box}} \times \frac{N_{seg}}{N_{box}} \right)}_{\substack{\text{short-range} \\ \text{interaction forces}}} + \underbrace{N_{box} \times \left((N_{box} - 27) \times \frac{N_{seg}}{N_{box}} \right)}_{\substack{\text{long-range} \\ \text{interaction stresses}}} \tag{2.73}$$

where N_{seg}/N_{box} is the average number of dislocation segments per box considering an homogeneous segment distribution. Further, since long-range interactions pertain to pairs of segments whose mutual distance is relatively large, their relative motion over a small increment of time is not expected to induce significant changes in the associated interaction forces. For such reason, long-range stresses may be solely evaluated at a certain frequency f_{box}. For instance, long-range stress fields can be updated at the center of each box α every 10 time steps, i.e. for $f_{box} = 1/10$. Including this, the complexity $\mathcal{C}_{box}$ of the Box Method can be estimated from (2.73) as:

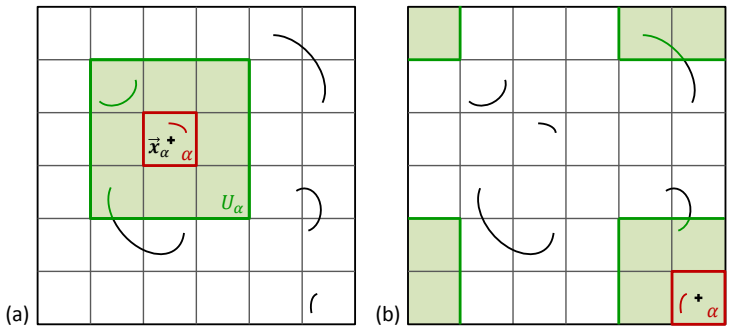


Figure 2.14: 2D schematic of the Box Method. The primary volume is partitioned into a predefined number of boxes. (a) Neighbor boxes of red box α are delineated by the green region U_α. For any red dislocation lying in box α, the elastic interactions induced by green dislocation segments in neighbor boxes will be accurately computed while that of long-range black dislocation segments will be approximated at the center of the box $\vec{\boldsymbol{x}}_\alpha$. (b) In this case, box α lies at an edge of the primary volume such that neighbor boxes are determined using periodic boundary conditions.

$$\mathcal{C}_{box} \propto \underbrace{27\frac{N_{seg}^2}{N_{box}}}_{short-range} + \underbrace{f_{box}(N_{box}-27)N_{seg}}_{long-range} \tag{2.74}$$

Clearly, when f_{box} is fixed, the computational savings allowed by the use of the Box Method is governed by the ratio N_{box}/N_{seg}. When it is low, the overall complexity is dominated by the evaluation of short-range interactions and tends to $\mathcal{O}(N_{seg}^2)$. In that case, the gain is minimal. On the contrary, when N_{box}/N_{seg} is large, the computation cost becomes dominated by the evaluation of long-range interactions. In that case, it must be ensured that the long-range approximation remains valid, as the critical distance defining the transition between precise short-range calculations and approximated long-range evaluations is fixed by N_{box}. Furthermore, selecting a number of box such that $N_{box}/N_{seg} > 1$ can lead to overrunning the initial $\mathcal{O}(N_{seg}^2)$ complexity. In other words, there exists a region of optimality for N_{box}/N_{seg}, such that the choice of the number of boxes N_{box} must be made wisely in order to optimize the computational load while ensuring that the long-range approximation remains

valid. Ideally, the number of boxes N_{box} partitioning the simulation volume should be dynamically determined every f_{box} iterations by optimization of the ratio N_{box}/N_{seg}. Note also that the estimation of the complexity given here does not account for PBC. When PBC are used (see Section 2.3.2), image dislocations would be directly accounted for in the long-range fields.

As another approach, the Fast Multipole Method (FMM) is also currently widely used in DDD simulations. This technique, whose original core idea initially relies on an expansion in Taylor series of the kernel (2.14) formed by the integral over the derivatives of the Green's function, has the advantage to reduce the computational complexity of long-range interactions to a $\mathcal{O}(N_{long})$ process, where N_{long} are the long-range dislocations [266, 148, 251, 5]. To make it fully compatible with a partitioning of the simulation volume into cells and subcells, a hierarchical FMM has been developed, in which translation operators are introduced to calculate and combine the multipole expansions between the different levels of the hierarchy [5]. Here again, there exists a region of optimality dynamically conditioning the choice of the number of subcells as a function of the number of dislocation segments. Specifically, the accuracy of the expansion to compute the stress of a segment increases with both the distance at which the stress is evaluated and the length of the segment. Consequently, when using the FMM, the discretization size l_{avg} of the dislocation network is voluntarily reduced as the number of segments increases so as to allow for a more important number of subcells (decreasing the average size of segments renders the expansion approximation valid at shorter distances, thereby allowing for a larger number of subcells). This process ensures that the number of short-range interactions remains relatively constant as segments multiply, while preserving the accuracy of the series expansion. Note also that the prefactor associated with the $\mathcal{O}(N_{long})$ complexity is fairly large due to the complex form of the translation operators. Recently, an improvement of the FMM technique has been achieved by using spherical harmonics

to diagonalize the translation operators [270]. Further, the FMM was adapted to stress calculations in anisotropic elasticity [263]. The FMM algorithm has not been implemented in the DDD code developed during this thesis.

2.9.2.2 Local velocities approximation

As detailed in Section 2.5, the determination of the nodal velocities – and consequently that of the nodal displacements – involves solving for the linear system of equations (2.60) for each dislocation group. As illustrated in figure 2.10, a dislocation group here refers to a connected ensemble of nodes, in the sense of graph theory. Thus, for large groups of dislocation nodes, system (2.60) may become computationally expensive to solve – even when sparse solvers are used – and approximations may be considered. Further, large groups of dislocation nodes are not uncommon in practice. As a matter of fact, as junctions form, the whole dislocation network may end up being lumped into a single dislocation group. Therefore, to alleviate the computational cost associated with the resolution of system (2.60), the following approximation can be used. If one assumes that the dislocation network is well discretized, then one can assume that the velocities of moving nodes connected together are roughly identical, i.e. $\vec{\boldsymbol{V}}_j \approx \vec{\boldsymbol{V}}_i$ for all moving nodes j connected to node i, and $\vec{\boldsymbol{V}}_k = 0$ for all fixed nodes k connected to node i. Using this local approximation, the velocity of node i can be directly obtained from equation (2.60) by solving the following equation:

$$\vec{\boldsymbol{F}}_i \approx \left[\sum_j \frac{l_{ij}}{2} \boldsymbol{\mathcal{B}}(\theta_{ij}) + \sum_k \frac{l_{ik}}{3} \boldsymbol{\mathcal{B}}(\theta_{ik}) \right] \vec{\boldsymbol{V}}_i \tag{2.75}$$

where indices j and k denote the moving and fixed nodes connected to node i, respectively. With equation (2.75), the force-velocity relation reduces to a 3×3 system which can be solved independently for each node. In that sense, it allows for a decoupling of equations (2.60) in which the initial full system of size $3N$ is now decomposed into N systems of size 3 for a group of N nodes. Furthermore, using such approximation,

better numerical stability is observed in the computation of nodal velocities.

2.9.3 Parallel computing

The computational cost of DDD simulations constitutes one of the principle limitations associated with this numerical method. Practically, simulations are currently limited to reach level of strains of the order of 1–2%, thereby preventing any direct connection with higher-scale models so far. Furthermore, the attainment of such level of deformation is currently only possible by means of parallel computation – when using the regular method –, and any sub-optimal implementation would necessarily preclude it. Therefore, the objective of this work is to build an efficient tool by adopting an appropriate parallel implementation, so that relevant simulations can be achieved in a reasonable amount of time, provided limited computational resources.

Different strategies can be envisioned for the parallel computing implementation of DDD simulations depending on the number of processors that are to be used. Thus, the goal of this thesis is not to compete with massively parallel algorithms such as ParaDis [5], that was designed to enable simulations on supercomputers up to 100,000+ CPUs [31], but rather to build a competitive and scalable DDD simulation tool for small clusters up to 100 CPUs. Besides, note that the alternative DDD-FFT method as a way to address the current computational limitations of the regular approach will be presented in Chapter 3.

As detailed in Section 2.5.3, the most computationally intensive task in regular DDD simulations pertains to the calculation of the elastic interactions between pairs of segments. This procedure generally accounts for more than 90% of the total time of the simulations. Therefore, particular efforts have been undertaken to optimize the numerical implementation and adopt a proper parallelization scheme for this process. In the context of parallel computing, the achievement of a good scalability is essentially conditioned by the achievement of an excellent load balance between

the CPUs, especially when communications between processors are required at each time step. This is because when the computational load is not perfectly balanced, the relative waiting time of the fastest CPU (i.e. that which has been attributed the lowest load, assuming all CPUs have the same frequency) to the slowest CPU (i.e. that which has been assigned the highest load) leads the overall efficiency of the algorithm to dramatically drop. In [5], the parallel strategy relies on a dynamical domain decomposition that partitions the simulation volume along all three spatial directions using a recursive k-d tree data structure. Further, at each time step, the boundaries of the different cells are moved so as to ensure that the computational load remains balanced between all domains as the dislocation configuration evolves. With this approach, each domain can be separately handled by a single CPU, provided that the information about dislocation segments close to the external boundaries of each computational domain are transferred across CPUs via cheap point-to-point communications. Further, such parallel architecture is conveniently coupled with a hierarchical Fast Multipole Method (FMM) [266, 148, 251, 5] so as to directly incorporate an approximation of long-range elastic interactions (see Section 2.9.2.1 for more details). Although such strategy is probably the most efficient algorithm for treating very large dislocation configurations since it provides excellent scalability, it can also become heavy for relatively small size simulations, and substantially increase the complexity of the numerical implementation, thereby hindering the overall versatility and modularity of the tool. For this reason, a different strategy was adopted in this work. It is detailed below.

As mentioned in Section 2.5.3, the bottleneck of DDD simulations pertains to the $\mathcal{O}(N_{seg}^2)$ process associated with the computation of the elastic interactions between pairs of individual segments, where N_{seg} is the total number of segments in the simulation. Recall that this quadratic complexity physically stems from the fact that each dislocation segment interacts with all other segments present in the volume through

their elastic stress fields. As detailed in Section 2.9.2.1, the decay in $1/R$ of the stress field can be advantageously exploited so as introduce approximation techniques to evaluate the long-range elastic interactions. With this, and assuming that the dislocation segments are homogeneously distributed over the volume, the computational complexity of analytical short-range elastic interactions reduces to $\mathcal{O}(N_{nei}^2)$, where $N_{nei} \leq N_{seg}$ is the average number of neighbor segments associated with short-range interactions. In practice however, assuming a constant number of neighbors N_{nei} for each segment so as to equally distribute the segments among the CPUs would lead to a sub-optimal load balancing strongly affecting the global scalability. As a matter of fact, a good load balance can only be achieved when the number of individual elastic interactions to compute is equal on each CPU. For such purpose, the strategy adopted here consists in distributing dislocation segments among processors so as to balance the total number of their actual neighbors. Full details on the parallel numerical implementation of the critical stages of the DDD cycle are presented in Appendix D.2.

2.10 Conclusion

In this chapter, a DDD nodal code relying on the superposition principle formalism was developed. The present tool includes an explicit treatment of junction formations and is able to treat any kinds of crystallographies, especially by handling the additional complexities associated with low-symmetry crystals. In addition, by adopting a rigorous parallel strategy and implementing several numerical optimizations, a robust, efficient and competitive numerical tool that shows good scalability up to 100 CPUs is obtained.

To assess the accuracy of the DDD code developed in this work, several validations were performed. Thus, it was ensured that the stress fields computed for infinitely long straight edge and screw dislocations match the analytical solutions. Further,

the activation stress of a Frank-Read source was assessed as a function of its length and of the segment discretization size, and was compared to analytical solutions and other DDD simulation tools results.

In this thesis, the DDD code developed in this chapter will be used in Chapter 4 to quantify dislocation-dislocation interaction coefficients in HCP Mg and test for the validity of slip-hardening laws used in current constitutive models. Prior to doing so, the present DDD code will be used to compute latent-hardening coefficients for FCC materials, and will yield similar values as those obtained in the literature, thereby providing a supplementary validation of the present implementation.

Further, the present DDD code was developed in a modular fashion such that additional features can be added easily. For instance, a transmission scheme will be implemented in Chapter 5.1 to study the interactions and reactions of dislocations across twin-boundaries. Finally, the present code will serve as a basis for the novel DDD FFT-based approach developed in Chapter 3 to address current limitations of regular DDD simulations pertaining to the treatment of anisotropic and heterogeneous elasticity.

CHAPTER III

DDD-FFT: A FAST FOURIER TRANSFORMED BASED FORMULATION FOR PERIODIC DISCRETE DISLOCATION DYNAMICS

3.1 Motivation

Through Chapter 2, a detailed overview of the physical model and the numerical approach and implementation underlying regular DDD simulations is given. Recall that in this work, as mentioned in Section 2.4, the designation of *regular* or *conventional* simulations is indifferently used to refer to DDD simulation approaches in which stress fields of dislocation segments are individually computed in an infinite elastic medium and added to one another by virtue of the superposition principle to define the internal stress at every material point. With this approach, it has also been emphasized in Sections 2.5.3 and 2.9 that the very expensive $\mathcal{O}(N_{seg}^2)$ computational cost associated with the calculation of elastic interactions was the bottleneck of those simulations. Despite the tremendous efforts that have been undertaken in the recent years, such as the introduction of a non-singular dislocation theory leading to analytical interaction force expressions for isotropic elasticity [35] and the development of approximation techniques such as the Box Method [240] or the FMM method, that have certainly permitted significant savings in the computational time [148, 5, 270], the evaluation of individual segment-segment interactions still remains the most costly procedure in regular DDD simulations.

More generally, the main limitations of current DDD codes lie in the treatment of anisotropic elasticity and elastic inhomogeneities. The first limitation arises from the fact that, in contrast to isotropic elasticity, no closed-form solution is available

for the stress field induced by a dislocation in an anisotropic medium. Thus, the non-singular theory developed by Cai et al. [35] cannot be simply extended to the case of anisotropic elasticity. Consequently, analytical expressions for elastic interaction forces cannot be derived, requiring to resort to substantially more costly numerical integration techniques such as the Gauss quadrature. Although substantial efforts have been undertaken in the past years to develop new numerical techniques allowing for savings in computation time [202, 108, 264], notably via the use of spherical harmonics [12, 13] (see Appendix B.2), the relative cost of anisotropic calculations usually amounts to at least one order of magnitude larger than that of isotropic calculations, and remains conditioned by the amount of anisotropy and the desired level of accuracy [12]. Practically, the significant increase in computation time associated with anisotropic calculations has precluded large simulations on such materials thus far. For this reason, to date, most of DDD simulations have been performed on FCC crystals, and the few studies performed on low-symmetry crystals have generally neglected the effects of elastic anisotropy [176, 62, 22]. The second limitation results from the high computational burden associated with the finite element method (FEM) (see Appendix E) required in current approaches to enable the incorporation of inhomogeneous elasticity in DDD. Further, in such simulations, special multilayer formulations of elastic interaction forces should be included to account for the elastic mismatch across the different media [107, 2].

From a practical perspective, such current limitations in terms of time and length scales (one is usually limited to 1–2% strain on 1–100 μm^3 samples with current approaches, provided important computational resources) have precluded large simulations on anisotropic and heterogeneous structures so far, limiting their application to the study of relatively small systems.

In parallel to the regular DDD approach, Lemarchand and co-workers [147] proposed a framework based on an eigenstrain formalism, the Discrete Continuous Model

(DCM), in which each dislocation is treated as a plate-like inclusion. By coupling this approach with a FEM framework, the stress state can be directly computed from a classical elasto-viscoplastic constitutive law in which the plastic strain is obtained from the motion of discrete dislocations [234]. In contrast with conventional DDD approaches relying on the superposition principle, the FEM mesh in the DCM approach accounts for both the boundary conditions and the long-range isotropic or anisotropic elastic interactions. Particularly, the intensive computation of internal stresses coming from dislocation-dislocation elastic interactions can potentially be reduced to a minimum with this approach, since it only concerns local pairs of segments whose interaction distance is smaller than half of the FEM mesh size. Besides, when using PBC, this approach does not require the computation of the elastic stress field of image dislocations [234]. In addition, this model has shown its capability to deal with inhomogeneous and anisotropic elasticity. However, the coupling of the DCM with a FEM framework eventually renders the approach computationally prohibitive (if not impractical for fine meshes), thereby limiting its use to coarse meshes and consequently requiring a significant a number of local interactions to be computed.

In the present thesis, a new approach for DDD simulations is proposed so as to simultaneously address the main limitations of current approaches, i.e. the computational cost of dislocation-dislocation interactions and the treatment of anisotropic and heterogeneous elasticity. For such purpose, a novel full-field approach based on Fast Fourier Transforms (FFT) and on the DCM is developed for the computation of mechanical fields in periodic DDD simulations. Of particular interest, the new DDD-FFT method allows for the treatment of both elastic anisotropy and heterogeneities in a very computationally efficient fashion.

As mentioned in Section 2.2, the use of Fourier techniques to solve problems in continuum mechanics has been long been identified as a convenient approach. It was already used by Green [99] and has been extensively employed by Mura in his seminal

work on micromechanics [181]. Later, the development of the FFT algorithm providing a very efficient technique to numerically compute discrete Fourier transforms has received a considerable echo in many scientific domains. In particular, the use of FFT techniques to solve problems in continuum mechanics has, over the past years, constituted an increasingly attractive technique and has shown to lead to significant computational gains [153]. Originally, the use of the FFT algorithm applied to mechanics of materials was introduced by Moulinec and Suquet [178, 179]. The authors introduced the first FFT-based formulation in computational mechanics of materials so as to enable fast computation of the local responses of composites. Interestingly, the FFT-based approach was first used for heterogeneous elasticity as a way to obtain the effective overall mechanical response. It was later extended by Lebensohn and co-workers to model the viscoplastic [143] and the elasto-viscoplastic [144] response of polycrystalline aggregates. Regarding dislocation dynamics, FFT-based formulations for computing dislocation elastic interactions have been used in approaches where dislocation lines are not tracked individually, such as in the level set method [262] and in phase field approaches [246, 116]. However, the DD level set method has been reported to be computationally costly due to the need of higher order parametrizations to represent dislocation lines [243], and the formation of junctions, crucial to accurately model strain hardening [31, 5], is not explicitly treated in both approaches. Particularly, the absence of discretization of dislocation lines either restricts junction formation to be purely governed by elastic interactions, generally requiring very small time step increments, or prevents its accurate description. Therefore, in the present thesis, a FFT-based approach was developed at the length scale of DDD simulations, such as to benefit from the computational efficiency of the FFT method while allowing for a discrete representation of dislocation lines. Thus, the proposed approach paves the way towards achieving scale transition from DDD to mesoscale plasticity. Specifically, with the ability of DDD-FFT formulation to account for heterogeneous

elasticity, it is to be expected that the method presented here could enable efficient DDD simulations on polycrystals.

The present chapter is organized as follows. First, the DCM model is fully presented and its advantages and limitations are discussed in Section 3.2. Then, the homogeneous formulation of the DDD-FFT method is introduced in Section 3.4, in which spectral formulations for continuum mechanics are first presented from a general perspective. Further, the development of an elastically heterogeneous model is detailed in Section 3.5. Finally, details related to the delicate numerical implementation of FFT-based approaches are provided in Section 3.6.

3.2 The Discrete-Continuous Model

3.2.1 General overview

In parallel to the development of the framework used in regular DDD simulations and presented in Chapter 2, Lemarchand and co-workers [147] have proposed a different approach to the DDD problem: the Discrete-Continuous Model (DCM). This approach was for instance used to study of the effect of elastic anisotropy in thin films [101], model the plastic deformation in metal matrix composites [100], and investigate plasticity in nickel-based single-crystal superalloys in which matrix channels were formed by the presence of precipitate phases [235, 236]. In this section, a brief overview of the DCM will be given, as it constitutes the basis on which the DDD-FFT model is built. For a complete presentation of the DCM, the reader is referred to references [147] and [234].

In contrast with the regular approach, the DCM model relies on an eigenstrain formalism, in which each dislocation is considered as an Eshelbian inclusion of thickness h, as depicted in figure 2.5(b). Such consideration has several implications, which will be discussed in details through the rest of this section. First, it requires that each initial dislocation should be introduced as a closed loop. As depicted in figure 2.5(a)

and discussed in Section 2.2.2, the definition of a dislocation line in the eigenstrain theory is given as the boundary of the surface defect, implying a closed line contour. Further, in contrast with the regular approach in which the medium is considered as purely elastic, the DCM involves an elasto-viscoplastic framework in which the plastic strain $\boldsymbol{\epsilon}^p$ results from the motion of dislocations. Thus, the constitutive law at each material point $\vec{\boldsymbol{x}}$ of the simulation volume is given by:

$$\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{x}}) = \boldsymbol{C} : (\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}})) \tag{3.1}$$

where tensors $\boldsymbol{\sigma}^{DCM}$ and $\boldsymbol{\epsilon}$ denote the stress associated with the DCM and the total strain, respectively, and $\boldsymbol{C}$ is the elastic stiffness matrix. Importantly, the numerical quantification of the plastic strain within the simulation volume requires a dedicated regularization procedure. Specifically, in [147] and [234] the DCM is coupled by its authors to a FEM framework. With this, the constitutive law that must hold at each integration point $\vec{\boldsymbol{p}}$ of the mesh is given from relation (3.1) as:

$$\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{p}}) = \boldsymbol{C} : (\boldsymbol{\epsilon}(\vec{\boldsymbol{p}}) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{p}})) \tag{3.2}$$

where $\boldsymbol{\epsilon}^p(\vec{\boldsymbol{p}})$ is the history-dependent accumulated plastic strain tensor resulting from the motion of all dislocations regularized at point $\vec{\boldsymbol{p}}$, whose increment over time is classically given from expression (2.26) as:

$$d\boldsymbol{\epsilon}^p(\vec{\boldsymbol{p}}) = \frac{1}{2}\sum_s \left(\vec{\boldsymbol{b}}^{\,s} \otimes \vec{\boldsymbol{n}}^{\,s} + \vec{\boldsymbol{n}}^{\,s} \otimes \vec{\boldsymbol{b}}^{\,s}\right) d\gamma^s(\vec{\boldsymbol{p}}) \tag{3.3}$$

where the summation is performed over all slip systems s with Burgers vector $\vec{\boldsymbol{b}}^{\,s}$ and unit normal $\vec{\boldsymbol{n}}^{\,s}$. The calculation of the plastic shear increment $d\gamma^s(\vec{\boldsymbol{p}})$ via the regularization procedure is explained in details in the following. Before, notice that by taking advantage of the time discretization used in DDD, the non-linear constitutive law (3.3) is often solved for via an explicit scheme, whereby the plastic strain state

$\boldsymbol{\epsilon}^p$ at step t is used to compute the stress and strain states at step $t + dt$.

Following the eigenstrain formulation, the DCM relies on considering each dislocation loop as a plate-like inclusion of thickness h. Obviously, the analogy with Mura's theory [181] giving rise to the derivation of the stress state associated with a dislocation loop (e.g. equation (2.14)) presented in Section 2.2.2 is evident. Thus, the core idea of the DCM is to directly obtain the stress state associated with the dislocation configuration from the calculation of the plastic strain associated with the motion of the ensemble of dislocation loops, following the procedure given from equations (2.10) to (2.14) to solve for the classical boundary value problem in continuum mechanics (2.37)–(2.40). Consequently, it is to be noticed that, in contrast with the regular approach, the DCM approach does not invoke the principle of superposition.

The apparent difficulty of the DCM resides in the calculation of the plastic strain at each integration point. As illustrated in figure 2.5(b), the inclusion thickness t associated with a dislocation physically corresponds to the inter-atomic distance associated with its slip plane. In the context of a numerical formulation however, the thickness h differs from t, and is related to the length scale of the numerical discretization. Thus, the shear produced by the motion of a dislocation can be determined as follows. As depicted in figure 3.1, it is considered that the elementary sheared area dS swept by the glide of a portion of a dislocation segment ij produces an elementary homogeneous plastic shear $d\gamma$ within an elementary spherical volume $d\phi$ of radius $h/2$ centered on the sheared area dS [234], such that:

$$d\gamma = \frac{b\,dS}{d\phi} = \frac{6b\,dS}{\pi h^3} \tag{3.4}$$

where b is the magnitude of the segment's Burgers vector, and $\pi h^3/6$ is the volume of any elementary sphere $d\phi$ of radius $h/2$. This approach is naturally consistent with the crystallographic definition of dislocation shear given in expression (2.24). However, one of the challenging aspects of this model lies in the distribution of the

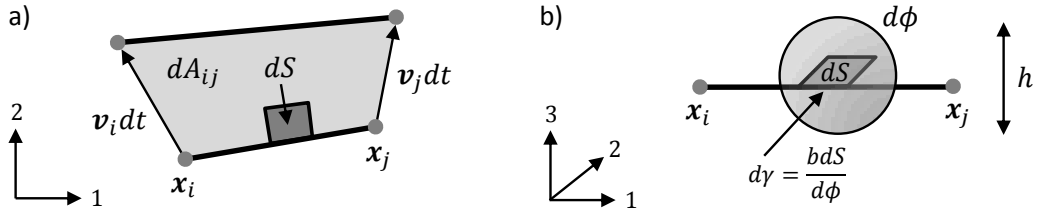


Figure 3.1: Sheared area dA_{ij} produced by the glide of a dislocation segment ij during time dt. The dislocation segment is defined by its end nodes i and j moving from their initial positions $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ at velocities $\vec{\boldsymbol{v}}_i$ and $\vec{\boldsymbol{v}}_j$, respectively. Schematic of (a) an elementary sheared area dS swept by the glide of a portion of a dislocation segment, and (b) the resulting elementary homogeneous plastic shear $d\gamma$ associated with an elementary spherical volume $d\phi$ of radius $h/2$ centered on the sheared area.

plastic strain to the mesh in order to perform accurate and realistic simulations. Specifically, the choice of the elementary sheared volume V in equation (2.24) – i.e. the choice of the regularization parameter h in (3.4) – is critical since it constitutes a key parameter of the elasto-viscoplastic model (3.2).

In the recent implementation of Vattré and co-workers [234], the numerical procedure to regularize the plastic shear contribution of a gliding dislocation segment at an integration point of a FEM mesh is as follows: the shear $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ induced by the glide of dislocation segment ij at integration point $\vec{\boldsymbol{p}}$ is not null only if $\vec{\boldsymbol{p}}$ lies within one or more elementary volume $d\phi(\vec{\boldsymbol{x}})$ centered in $\vec{\boldsymbol{x}}$ when $\vec{\boldsymbol{x}}$ spans the entire segment's glide area dA_{ij}, i.e.:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}} = \frac{6b}{\pi h^3} \int_{dA_{ij}} \chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}}) d\vec{\boldsymbol{x}} \tag{3.5}$$

where the characteristic function $\chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}})$ of integration point $\vec{\boldsymbol{p}}$ takes value $\chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}}) = 1$ if $\vec{\boldsymbol{p}}$ lies within the elementary sphere $d\phi(\vec{\boldsymbol{x}})$ centered in $\vec{\boldsymbol{x}}$, $\chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}}) = 0$ otherwise. By denoting:

$$dS_{ij}^{\vec{\boldsymbol{p}}} = \int_{dA_{ij}} \chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}}) d\vec{\boldsymbol{x}} \tag{3.6}$$

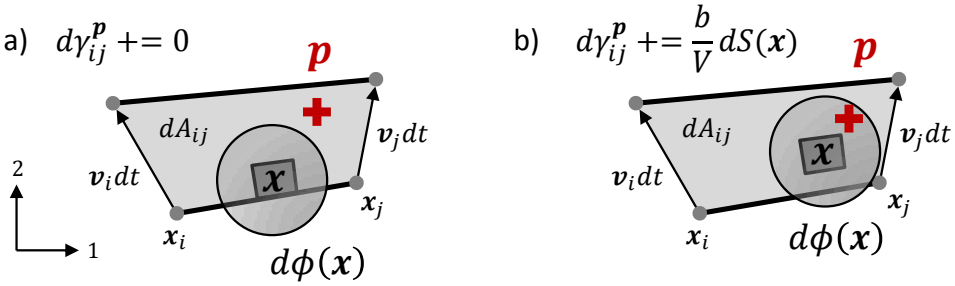


Figure 3.2: Numerical regularization procedure of the plastic shear produced by the glide of a dislocation segment when using the DCM as implemented in [234]. The surface integration in (3.5) is numerical calculated as follows. By successively discretizing the sheared area dA_{ij} into elementary swept areas $dS(\vec{\boldsymbol{x}})$, the increment of plastic shear $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ at integration point $\vec{\boldsymbol{p}}$ is: a) null when $\vec{\boldsymbol{p}}$ does not lie within the elementary sphere $d\phi(\vec{\boldsymbol{x}})$ centred in $\vec{\boldsymbol{x}}$, or b) equal to the shear associated with dS when $\vec{\boldsymbol{p}}$ lies in $d\phi(\vec{\boldsymbol{x}})$.

the integral of the characteristic function over the area swept by dislocation segment ij, equation (3.5) is rewritten as:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}} = \frac{6b}{\pi h^3} dS_{ij}^{\vec{\boldsymbol{p}}} = \frac{b}{V_e} dS_{ij}^{\vec{\boldsymbol{p}}} \tag{3.7}$$

where V_e is the volume of the elementary sphere $d\phi(\vec{\boldsymbol{x}})$. As a result, the total regularized shear increment $d\gamma^s(\vec{\boldsymbol{p}})$ on system s in equation (3.3) is calculated as the sum of the shear increments produced by all dislocation segments ij belonging to slip system s at integration point $\vec{\boldsymbol{p}}$:

$$d\gamma^s(\vec{\boldsymbol{p}}) = \sum_{ij \in s} d\gamma_{ij}^{\vec{\boldsymbol{p}}} \tag{3.8}$$

In [234], the numerical integration associated with $dS_{ij}^{\vec{\boldsymbol{p}}}$ in equation (3.6) is performed by successively discretizing the glide area dA_{ij} into elementary areas dS along the segment length and its glide direction (see figure 3.2). As discussed in [147], in this approach h is a parameter whose value must be chosen of the order of the mesh size L_{mesh} so as to ensure that the union of the elementary spheres $d\phi$ when dS spans

the entire sheared area dA_{ij} at least contains one integration point. This is to ensure that the shear produced by the motion of any dislocation segment will be entirely transferred to the mesh. Thus, in reference [147], the value $h = 3/2L_{mesh}$ has been determined as optimal and the same value has been used in reference [234]. More details on the numerical implementation of the DCM can be found in both these papers.

3.2.2 Advantages and limitations

With equations (3.2) to (3.8), the DDD boundary value problem (2.37)–(2.40) can be solved using the FEM method (see Appendix E.3 for details on the elasto-viscoplastic FEM formulation). Note here that the visco-plastic behavior directly follows from the plastic strain increment $d\boldsymbol{\epsilon}^p$ dependence on the stress. Indeed, the general framework laid out in Section 2.5 to determine dislocation nodal velocities is retained. However, the Peach-Koehler force is here directly taken as the integral of the stress $\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{p}})$ obtained in equation (3.2) thanks to the DCM-FEM coupling. This procedure is obviously valid since, thanks to the eigenstrain formalism, the resulting stress field $\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{x}})$ determined everywhere in the simulation volume encompasses both the external (from the applied loading and the boundary conditions) and the internal (from the dislocation microstructure) contributions without redundancies. More precisely, the DCM-FEM eventually provides a discrete stress state at each mesh node. From there, the Peach-Koehler force $\vec{\boldsymbol{f}}_{ij}^{pk}$ in equation (2.55) on any dislocation segment ij can be calculated by interpolating $\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{x}})$ along the segment length as:

$$\vec{\boldsymbol{f}}_{ij}^{pk} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\boldsymbol{\sigma}^{DCM}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| \tag{3.9}$$

where the different quantities in expression (3.9) have the same meaning as that used in Section 2.5. However, as explained in [234], the resort to such interpolation leads to lose in the evaluation of $\vec{\boldsymbol{f}}_{ij}^{pk}$ the interaction forces corresponding to pairs

of segments whose interaction distance is smaller than $h/2$. Consequently, solely the internal stresses pertaining to long-range dislocation-dislocation elastic interactions are inherently accounted for by the mesh. In this approach, long-range elastic interactions refer to pairs of segments whose interaction distance is larger than half of the regularization parameter h. In other words, the DCM model requires the additional computation of interaction forces for pairs of segments which are closer than $h/2$. Such interactions can be for instance computed using the non-singular analytical formulation [35] (see Section 2.5.3), and must be added as supplementary local contributions to the Peach-Koehler force.

The main interest of the DCM certainly lies in that it introduces an alternative approach that theoretically transforms the initial $\mathcal{O}(N_{seg}^2)$ problem of regular DDD simulations into a $\mathcal{O}(N_{el}^2)$ problem, where N_{el} is the number of elements of the FEM mesh (the best achievable complexity of the standard FEM approach is generally accepted to be $\mathcal{O}(N_{el}^2)$, see Appendix E). At first sight, the potential gain that it may allow is not obvious, and it would be inopportune to consider that choosing $N_{el} \leq N_{seg}$ would provide a direct improvement with respect to regular DDD simulations. This is for two reasons: first, because the equivalence between the complexity of both approaches is only valid when the number of supplementary local interactions to compute is small in the DCM method, i.e. when the mesh is relatively fine. Second, the prefactor associated with both complexities has a significant impact and prevents the direct comparison of both approaches. While the second reason cannot be assessed from a general perspective – the prefactor highly depends on the numerical implementation of the approaches –, the first point can provide an idea on the performance of the DCM-FEM method. When the mesh is coarse, the cost of the FEM may be reasonable, but the important number of local interactions to compute ($h/2$ is large in this case) reshapes the DCM approach to the original $\mathcal{O}(N_{seg}^2)$ problem. Alternatively, when the mesh is fine, the number of short-range interactions to

calculate can be reduced to a minimum. However, the cost of the FEM procedure generally becomes impractical in that case. For instance, a mesh of $N_{el} = 32^3$ elements is the upper limit in terms of resolution that can be achieved – provided reasonable computational resources –, and at this relatively low resolution, the DCM has been reported not to be competitive in comparison with the regular approach [234].

Nevertheless, one of the most attractive advantages of the DCM is its inherent ability to treat anisotropic and heterogeneous elasticity. This is because the general definition of the stiffness tensor can be easily used in the FEM formulation, and each integration point can be assigned a different material. Therefore, the long-range stresses accounted for by the mesh are calculated at the same cost in isotropic or anisotropic elasticity. However, the short-range interactions still remain an issue in this case since they must be explicitly computed using similar methods as in the regular DDD approach for anisotropic elasticity (e.g. spherical harmonics approach).

In the case of heterogeneous elasticity, the difficulty arises from the resolution that must be adopted in order to precisely describe the different phases. Thus, while the case of a bi-crystal will not pose any problem in general, the modelling of inclusions of small size or of complex shape will be impractical. For instance, a resolution of $N_{el} = 16^3$ was used to model precipitate phases in Ni-based single crystal superalloys [235, 236], restricting precipitates to be coarsely represented as large square inclusions. Notice that in order to alleviate the overall computational cost of the DCM-FEM, Vattré and co-workers suggested to perform the FEM computation according to a certain time step frequency. However, such cannot be done when a fine mesh is used, as dislocations may change cells in between two updates of the FEM, leading to dangerous artefacts such as double-counting their stress fields.

Furthermore, the use of a FEM framework is not the only computationally intensive process associated with the DCM. As mentioned in Section 3.2.1, the regularization procedure of the plastic strain involves the numerical integration introduced

in equation (3.6). As any numerical integration procedure, this process is costly and adds a significant computational burden to the simulation. This point is all the more important that the integration in expression (3.6) must be evaluated very precisely as the accuracy of the regularization is fundamental to the DCM. This characteristic finds its origin in the history-dependent formulation of the model. As a matter of fact, any inaccuracy will inevitably propagate and accumulate during the whole simulation, leading the internal stress state to depart from the actual dislocation configuration after some iterations. Notice that this issue does not exist as such in the regular approach since stresses are recomputed from the actual spatial positions of dislocation segments at each time step.

Finally, an interesting feature of the DCM should be mentioned at this point. The direct recourse to the eigenstrain theory implies that the initial mechanical state corresponding to the presence of the initial microstructure has to be taken into account in this approach (this stems from the history-dependent formulation). This can be done by introducing initial dislocation loops according to a Volterra-like process described in [234]. Although it imposes constraints on the initial dislocation microstructure that can be simulated, it also enforces the initial configuration to be relevant from a physical perspective [69]. One simple way to create a realistic initial microstructure in which dislocation sources are present consists in letting the initial Volterra loops relax. During this process, the formation of junctions creates physical pinning points for dislocations, thereby circumventing the necessity of introducing artificial pinning points (e.g. terminating segments) as usually employed in regular DDD.

In conclusion of this section, it is apparent that the DCM model presents an interesting alternative approach to the DDD problem. However, the method is rendered impractical by its computational cost. Specifically, the expensive computational load associated with the FEM procedure either restricts its use to coarse meshes – therefore requiring the computation of an important number of supplementary local

dislocation-dislocation interactions – or makes the procedure impractical due to the computational complexity of 3D FEM. To alleviate such complexity, a novel approach in which the DCM procedure is coupled to a FFT-based solver is proposed in this thesis, aiming at efficiently performing periodic DDD simulations in isotropic and anisotropic media.

3.3 Spectral methods in continuum mechanics

Over the course of the past decades, spectral methods have been widely employed to determine the mechanical state of a medium in continuum mechanics. As a matter of fact, Fourier-based methods are particularly adapted to derive analytical and numerical solutions for a medium subjected to distributed applied forces. Specifically, this results from the convenient use of the Green's function in the Fourier space, whose expression is straightforward. Further, Fourier-based methods have received much attention thanks to the development of the Fast Fourier Transform (FFT) algorithm, allowing to numerically calculate discrete Fourier transforms in a very efficient manner [51].

In this section, before the formulation for the DDD-FFT approach is introduced for an homogeneous and an heterogeneous medium, respectively, the solution of the boundary value problem in continuum mechanics using the spectral Green's function method is first detailed, and its application to the eigenstrain theory is discussed.

3.3.1 Boundary value problem solution using the spectral Green function method

As mentioned earlier, the resort to Fourier techniques has long been identified as a simple and convenient approach to solve for the fundamental boundary value problem in continuum mechanics. Thus, in linear elasticity, the mechanical state of a volume V subjected to traction and displacement boundary conditions can be determined by solving the boundary value problem introduced in equations (2.37)–(2.40), expressed

here in indicial notation as:

$$\sigma_{ij} = C_{ijkl}\epsilon^e_{kl} \tag{3.10}$$

$$\sigma_{ij,j} + f_i = 0 \tag{3.11}$$

$$u_i = u_i^* \text{ on } \partial V_u \tag{3.12}$$

$$\sigma_{ij} n_j = t_i^* \text{ on } \partial V_t \tag{3.13}$$

where the first equation refers to the elastic Hooke's law, the second to the mechanical equilibrium the stress must satisfy everywhere in the volume, and the last two equations correspond to the displacement and traction boundary conditions applied on the external volume surfaces ∂V_u and ∂V_t, respectively. Note that in addition to the divergence of the stress, the mechanical equilibrium here accounts for the body force f_i per unit volume defined throughout volume V. Using the definition of the stress (3.10) in equations (3.11) and (3.13) and that of the strain given in relation (2.3), the boundary value problem is expressed as:

$$\begin{aligned} C_{ijkl}u_{k,lj} + f_i = 0 \quad &\text{in} \quad V \\ u_i = u_i^* \text{ on } &\partial V_u \\ C_{ijkl}u_{k,l}n_j = t_i^* \text{ on } &\partial V_t \end{aligned} \tag{3.14}$$

Considering an homogeneous infinite volume V governed by linear elasticity and assuming that the body force f_i is localized and that the medium is traction-free at infinity, then for every spatial coordinate $\vec{\boldsymbol{x}}$ belonging to $V = \mathbb{R}^3$ one has:

$$\begin{aligned} f_i(\vec{\boldsymbol{x}}) \to 0 \quad &\text{when} \quad \vec{\boldsymbol{x}} \to \infty \\ C_{ijkl}u_{k,l}(\vec{\boldsymbol{x}})n_j = t_i^* \to 0 \quad &\text{when} \quad \vec{\boldsymbol{x}} \to \infty \end{aligned} \tag{3.15}$$

Under these assumptions, the displacement field in the medium is expected to be

bounded such that:

$$\begin{aligned} C_{ijkl}u_{k,lj} + f_i = 0 \quad \text{in} \quad V \\ u_i(\vec{\boldsymbol{x}}) \to 0 \quad \text{when} \quad \vec{\boldsymbol{x}} \to \infty \end{aligned} \tag{3.16}$$

Several approaches can be used to solve the boundary value problem expressed in expressions (3.14) to (3.16). Among them, the spectral technique based on the Fourier transforms constitutes an elegant and powerful approach to tackle such complex problem based on a system of partial differential equations. Such approach was for instance extensively used by Mura [181] in the context of the eigenstrain theory. As the use of Fourier techniques eventually involves Fourier transforms, the notations and definitions adopted in the rest of this thesis are introduced here. If $\vec{\boldsymbol{x}}$ denotes a coordinate in the 3D Cartesian space $\mathbb{R}^3$ and $\vec{\boldsymbol{\xi}}$ denotes a frequency of the reciprocate Fourier space $\mathcal{F}(\mathbb{R}^3)$, the Fourier transforms $\widehat{g}(\vec{\boldsymbol{\xi}})$ of any integrable function $g(\vec{\boldsymbol{x}})$ over $\mathbb{R}^3$ can be defined by:

$$\begin{aligned} \widehat{g}(\vec{\boldsymbol{\xi}}) = \mathcal{F}\left(g(\vec{\boldsymbol{x}})\right) = \int_{\mathbb{R}^3} g(\vec{\boldsymbol{x}}) e^{-\mathfrak{i}\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{x}} \\ g(\vec{\boldsymbol{x}}) = \mathcal{F}^{-1}\left(\widehat{g}(\vec{\boldsymbol{\xi}})\right) = \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} \widehat{g}(\vec{\boldsymbol{\xi}}) e^{\mathfrak{i}\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{\xi}} \end{aligned} \tag{3.17}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denotes the forward and inverse (or backward) Fourier transform operators, $\mathfrak{i} = \sqrt{-1}$ denotes the complex number, and where any quantities $\widehat{f}$ will denote a function expressed in the Fourier space, whose reciprocate function will be denoted f in the real space. Note that the definition of the Fourier transform is not unique and generally depends on the domain of application. One of the most remarkable properties associated with the use of the Fourier transform is that any spatial derivative of a function expressed in the real space becomes a simple multiplication in the Fourier space, i.e.:

$$\mathcal{F}\left(g_{,j}(\vec{\boldsymbol{x}})\right) = \mathfrak{i}\xi_j \widehat{g}(\vec{\boldsymbol{\xi}}) \tag{3.18}$$

This last result is simply obtained by performing an integration by parts using relations (3.17). With this, applying the Fourier transform to the equilibrium equation in (3.16) by using the relations defined in (3.17) and (3.18) leads to:

$$C_{ijkl}\xi_l\xi_j\widehat{u}_k(\vec{\boldsymbol{\xi}}) = \widehat{f}_i(\vec{\boldsymbol{\xi}}) \quad \text{in} \quad \mathcal{F}(V) \tag{3.19}$$

where the displacement field $\widehat{u}_k$ is expressed in the Fourier space $\mathcal{F}(V)$ associated with volume V. By introducing the following quantity:

$$K_{ik} = C_{ijkl}\xi_l\xi_j \tag{3.20}$$

the displacements in the Fourier space can be obtained from (3.19) as:

$$\begin{aligned} \widehat{u}_k(\vec{\boldsymbol{\xi}}) &= \widehat{G}_{ki}(\vec{\boldsymbol{\xi}})\widehat{f}_i(\vec{\boldsymbol{\xi}}) \\ \widehat{G}_{ki}(\vec{\boldsymbol{\xi}}) = K_{ik} = \left[C_{ijkl}\xi_l\xi_j\right]^{-1} &= N_{ik}(\vec{\boldsymbol{\xi}})D^{-1}(\vec{\boldsymbol{\xi}}) \end{aligned} \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.21}$$

where N_{ij} is the cofactor of K_{ij} and D its determinant, both given as:

$$\begin{aligned} N_{ij}(\vec{\boldsymbol{\xi}}) &= \frac{1}{2}e_{ikl}e_{jmn}K_{km}(\vec{\boldsymbol{\xi}})K_{ln}(\vec{\boldsymbol{\xi}}) \\ D(\vec{\boldsymbol{\xi}}) &= e_{mnl}K_{m1}(\vec{\boldsymbol{\xi}})K_{n2}(\vec{\boldsymbol{\xi}})K_{l3}(\vec{\boldsymbol{\xi}}) \end{aligned} \tag{3.22}$$

Full expressions of $\boldsymbol{N}$ and D for different crystal structures (i.e. for different shapes of the elasticity tensor $\boldsymbol{C}$) can be found in the literature (e.g. see reference [181]). In equation (3.21), G_{ki} refers to the spectral Green's function expressed in the Fourier space. By switching the indices in equations (3.21), the solution of the displacement field can be expressed in the Fourier space as:

$$\begin{aligned} \widehat{u}_i(\vec{\boldsymbol{\xi}}) &= \widehat{G}_{ik}(\vec{\boldsymbol{\xi}})\widehat{f}_k(\vec{\boldsymbol{\xi}}) \\ \widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) &= [C_{kjil}\xi_l\xi_j]^{-1} \end{aligned} \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.23}$$

Further, displacements in the real space can be obtained from equation (3.23) by taking the inverse Fourier transform of $\widehat{u}_i$ as:

$$\begin{aligned} u_i(\vec{\boldsymbol{x}}) &= \int_{\mathbb{R}^3} \widehat{G}_{ij}(\vec{\boldsymbol{\xi}})\widehat{f}_j(\vec{\boldsymbol{\xi}}) e^{\mathrm{i}\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{\xi}} \\ \widehat{f}_j(\vec{\boldsymbol{\xi}}) &= \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} f_j(\vec{\boldsymbol{x}}) e^{-\mathrm{i}\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{x}} \end{aligned} \tag{3.24}$$

Finally, combining the two relations in (3.24), the solution displacement field can be expressed as:

$$u_i(\vec{\boldsymbol{x}}) = \int_{\mathbb{R}^3} G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) f_j(\vec{\boldsymbol{x'}}) d\vec{\boldsymbol{x'}} \tag{3.25}$$

where

$$G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) = \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} \widehat{G}_{ij}(\vec{\boldsymbol{\xi}}) e^{\mathrm{i}(\vec{\boldsymbol{x}}-\vec{\boldsymbol{x'}})\cdot\vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{\xi}} \tag{3.26}$$

is the static Green's function for an infinite homogeneous linear elastic medium expressed in the real Cartesian space, whose expression for isotropic elasticity is given in equation (2.16). Interestingly, the Green function $G_{ij}(\vec{\boldsymbol{x}}-\vec{\boldsymbol{x'}})$ solely depends on the material properties and on the relative position of the source $\vec{\boldsymbol{x'}}$ and field $\vec{\boldsymbol{x}}$ points. In that sense, G_{ij} is independent of the loading and body force to which the medium is subjected to. Therefore, the convolution in (3.25) appears as a convenient manner to obtain the displacement field of a medium in response of the body force f_j. The result in (3.25) precisely constitutes the basis of the spectral approach to the eigenstrain theory, whose application is presented in Section 3.3.2. Further, it is interesting to notice from the comparison between expressions (3.23) and (3.25) that a convolution

in the real space corresponds to a simple multiplication in the Fourier space. Therefore, the interest of spectral methods in numerical applications via the use of the FFT algorithm will be presented in Section 3.3.3.

3.3.2 Application of the spectral method to the eigenstrain theory

In Section 3.3.1, the use of a spectral method to solve for the response of an infinite linear elastic medium to a body force was introduced. In this section, the framework is generalized to the case of material in which a distribution of eigenstrain exists. As detailed in [181], an eigenstrain is defined as an inelastic strain distribution arising from the presence of a local thermal expansion, a phase transition or a defect. From a general perspective, when the eigenstrain denoted ϵ^*_{ij} is prescribed within a finite domain Ω of medium V, region Ω is referred to an inclusion within the matrix $V - \Omega$. In regions where eigenstrains exist, the total strain ϵ_{ij} is decomposed as:

$$\epsilon_{ij} = \epsilon^e_{ij} + \epsilon^*_{ij} \tag{3.27}$$

where ϵ^e_{ij} denotes the elastic strain. Therefore, assuming a given distribution of eigenstrain throughout the whole medium V, the mechanical state at each material point $\vec{\boldsymbol{x}}$ must respect the constitutive law and the mechanical equilibrium:

$$\begin{aligned} \sigma_{ij}(\vec{\boldsymbol{x}}) &= C_{ijkl}\left(\epsilon_{kl}(\vec{\boldsymbol{x}}) - \epsilon^*_{kl}(\vec{\boldsymbol{x}})\right) \\ \sigma_{ij,j}(\vec{\boldsymbol{x}}) &= 0 \end{aligned} \qquad \forall \vec{\boldsymbol{x}} \in V \tag{3.28}$$

where the body forces are assumed to be vanishing everywhere. Combining both equations in (3.28) and using the small strain definition in (2.3), relations (3.28) can be written as:

$$C_{ijkl}u_{k,lj}(\vec{\boldsymbol{x}}) - \varphi_{ij,j}(\vec{\boldsymbol{x}}) = 0$$

$$\varphi_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl}\epsilon^{*}_{kl}(\vec{\boldsymbol{x}}) \tag{3.29}$$

where the quantity φ_{ij} referred to as the polarization tensor is introduced. Following the same procedure as that detailed in Section 3.3.1 for going from relations (3.16) to (3.23), the displacement field in the Fourier space can be expressed as:

$$\widehat{u}_i(\vec{\boldsymbol{\xi}}) = -\widehat{G}_{ik}(\vec{\boldsymbol{\xi}})\widehat{\varphi}_{kl,l}(\vec{\boldsymbol{\xi}}) \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.30}$$

Further, taking advantage of the properties of the derivation in the Fourier space given in (3.18), equation (3.30) can be conveniently written as:

$$\begin{aligned} \widehat{u}_i(\vec{\boldsymbol{\xi}}) &= -\mathrm{i}\xi_l\widehat{G}_{ik}(\vec{\boldsymbol{\xi}})\widehat{\varphi}_{kl}(\vec{\boldsymbol{\xi}}) \\ &= -\widehat{G}_{ik,l}(\vec{\boldsymbol{\xi}})\widehat{\varphi}_{kl}(\vec{\boldsymbol{\xi}}) \end{aligned} \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.31}$$

such as to express the resulting displacement field in the Fourier space as a function of the polarization tensor $\widehat{\varphi}_{ij}$ and the first derivative of the Green's function $\widehat{G}_{ij,k}$. Finally, taking the inverse Fourier transform of (3.31) leads to the following expression for the solution displacement field and its spatial derivative:

$$\begin{aligned} u_i(\vec{\boldsymbol{x}}) &= -\int_{\mathbb{R}^3} G_{ik,l}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})\varphi_{kl}(\vec{\boldsymbol{x'}})d\vec{\boldsymbol{x'}} \\ u_{i,j}(\vec{\boldsymbol{x}}) &= -\int_{\mathbb{R}^3} G_{ik,lj}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})\varphi_{kl}(\vec{\boldsymbol{x'}})d\vec{\boldsymbol{x'}} \end{aligned} \tag{3.32}$$

where the expressions of the derivatives of the Green's function can be directly obtained from (3.26). Obviously, the analogy with the previous section is complete when considering the divergence of the polarization as a body force so as to write $f_i = \varphi_{ij,j}$. Replacing the polarization tensor φ_{ij} by its complete expression given in (3.29), relation (3.32) is finally expressed as:

$$u_i(\vec{\boldsymbol{x}}) = -\int_{\mathbb{R}^3} C_{klmn}\epsilon^*_{mn}(\vec{\boldsymbol{x'}})G_{ik,l}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})d\vec{\boldsymbol{x'}} \qquad \forall \vec{\boldsymbol{x}} \in V \tag{3.33}$$

Expression (3.33) is the fundamental solution for the displacement field produced by an eigenstrain distribution in an infinite homogeneous medium. Of particular interest, expression (3.33) corresponds to the solution given in (2.10) for the displacement field produced by the presence of a dislocation in an infinite medium, i.e. when the eigenstrain ϵ^*_{ij} corresponds to the plastic strain ϵ^p_{ij} given in (2.12) associated with the dislocation, and for $V = \mathbb{R}^3$. Therefore, the procedure from going from equation (3.28) to (3.33) provides the complete derivation of equation (2.10) from constitutive relation (2.9) in Section 2.2.2. Further, using the small strain compatibility equation (2.3), the resulting strain field is expressed as:

$$\epsilon_{ij}(\vec{\boldsymbol{x}}) = -\frac{1}{2}\int_{\mathbb{R}^3} C_{klmn}\epsilon^*_{mn}(\vec{\boldsymbol{x'}}) \left(G_{ik,lj}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) + G_{jk,li}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})\right) d\vec{\boldsymbol{x'}} \qquad \forall \vec{\boldsymbol{x}} \in V \tag{3.34}$$

where $G_{ij,kl}$ denotes the second-order derivative of the Green function. Note that for the coming sections, it will be convenient to rewrite the convolution in (3.34) in tensorial form as:

$$\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) = -\boldsymbol{\Gamma} * [\boldsymbol{C} : \boldsymbol{\epsilon}^*](\vec{\boldsymbol{x}}), \quad \forall \vec{\boldsymbol{x}} \in V \tag{3.35}$$

where $*$ is the convolution operator in the real space, and $\boldsymbol{\Gamma}$ is the forth-order tensor associated with the minor and major symmetrizations of the second-order derivatives of the Green function in the real space, sometimes referred to as the modified Green operator. From expression (3.34), when $\vec{\boldsymbol{\xi}} \neq \vec{\boldsymbol{0}}$, the modified Green operator $\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}})$ in the Fourier space can be expressed as:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}) = \frac{1}{2}\left(\xi_l\xi_j\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) + \xi_l\xi_i\widehat{G}_{jk}(\vec{\boldsymbol{\xi}})\right) \tag{3.36}$$

where $\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) = [C_{kjil}\xi_l\xi_j]^{-1}$ is the static Green's function as introduced in (3.21). In a more general fashion, the modified Green's operator can be written as:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}) = \left\{\xi_j \left[\xi_m C_{kmin}\xi_n\right]^{-1}\xi_l\right\}_{\text{sym}}, \quad \forall\vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.37}$$

where $\{\}_{\text{sym}}$ denotes the major and minor symmetrizations of a given quantity. Naturally, equation (3.37) provides an expression for $\widehat{\Gamma}_{ijkl}$ that is valid for general anisotropy, and whose numerical evaluation does not induce any difficulty since it solely involves the inversion of a second-order tensor. Further, a simple fully analytical expression can be obtained in the case of elastic isotropy. First, it must be recalled from equation (3.21) that the Green's function can be conveniently written as:

$$\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) = N_{ik}(\vec{\boldsymbol{\xi}})D^{-1}(\vec{\boldsymbol{\xi}}), \quad \forall\vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.38}$$

where $N_{ik}(\vec{\boldsymbol{\xi}})$ and $D^{-1}(\vec{\boldsymbol{\xi}})$ are the cofactor and the determinant of quantity $K_{ik} = C_{ijkl}\xi_l\xi_j$, respectively, and whose general expressions are given by relations (3.22). For isotropic elasticity, the stiffness tensor C_{ijkl} can be expressed solely as a function of two materials parameters, such that:

$$C_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) \tag{3.39}$$

where λ and μ are the Lamé constants, such that $N_{ik}(\vec{\boldsymbol{\xi}})$ and $D^{-1}(\vec{\boldsymbol{\xi}})$ can be simply expressed from (3.22) as [181]:

$$\begin{aligned} N_{ik}(\vec{\boldsymbol{\xi}}) &= \mu\xi^2\big((\lambda+2\mu)\delta_{ik}\xi^2 - (\lambda+\mu)\xi_i\xi_k\big) \\ D(\vec{\boldsymbol{\xi}}) &= \mu^2(\lambda+2\mu)\xi^6 \end{aligned} \tag{3.40}$$

where $\xi^2 = \xi_q\xi_q$. As a result, the Green's function is analytically expressed as:

$$\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) = \frac{(\lambda+2\mu)\delta_{ik}\xi^2 - (\lambda+\mu)\xi_i\xi_k}{\mu(\lambda+2\mu)\xi^4}, \quad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.41}$$

and an expression of the modified Green's operator for isotropic elasticity is finally obtained by combining relations (3.37) and (3.41):

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}) = \frac{(\delta_{ik}\xi_l\xi_j + \delta_{il}\xi_k\xi_j + \delta_{jk}\xi_l\xi_i + \delta_{jl}\xi_k\xi_i)}{4\mu\xi^2} - \frac{(\lambda+\mu)\xi_i\xi_j\xi_k\xi_l}{\mu(\lambda+2\mu)\xi^4} \tag{3.42}$$

and where the condition $\widehat{\boldsymbol{\Gamma}}(\vec{\mathbf{0}}) = \mathbf{0}$ enforces $\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}) = \mathbf{0}$ at the highest frequencies in the current formulation.

Before concluding this section, an interesting perspective is worth mentioning here. As mentioned earlier, the presence of an eigenstrain in a finite region Ω of the medium V relates to the well-known case of an inclusion. When both the matrix $V-\Omega$ and the inclusion Ω have the same elastic properties, the inclusion is considered as homogeneous, and the displacement field in V due to the eigenstrain distribution ϵ^*_{ij} in Ω is given from (3.33) by:

$$u_i(\vec{\boldsymbol{x}}) = -\int_\Omega C_{klmn}\epsilon^*_{mn}(\vec{\boldsymbol{x'}})G_{ik,l}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}})d\vec{\boldsymbol{x'}} \qquad \forall \vec{\boldsymbol{x}} \in V \tag{3.43}$$

Note that in expression (3.43), the integral is solely carried out over region Ω since, by definition, the eigenstrain field ϵ^*_{ij} vanishes outside the inclusion, i.e. in $V-\Omega$. However, expression (3.43) provides an expression for the displacement field valid in the whole medium V, and is the basis of the Eshelbian micromechanics theory. Of particular interest, one of the most remarkable results of Eshelby is that, provided an uniform eigenstrain distribution in Ω, the resulting strain field – and consequently the resulting stress field – are uniform in the interior points of an homogeneous inclusion in an isotropic medium [79], such that one has:

$$\epsilon_{ij}(\vec{\boldsymbol{x}}) = S_{ijkl}\epsilon^*_{kl} \qquad \forall \vec{\boldsymbol{x}} \in \Omega \tag{3.44}$$

where S_{ijkl} denotes the Eshelby tensor, whose components are independent of the position $\vec{\boldsymbol{x}}$ and are given as a function of the shape of the inclusion and the elastic properties of the material. Thus, for particular geometries such as ellipsoidal inclusions, analytical expressions of S_{ijkl} have been obtained [181]. Although result (3.44) cannot be directly applied to the case of dislocations, the relevance of adopting a spectral approach to address problems based on an Eshelbian representation of dislocations as used in the DCM model (see Section 3.2.1) becomes apparent.

3.3.3 Comparison of FFT-based techniques with FEM approaches

In Sections 3.3.1 and 3.3.2, Fourier-based formulations for the boundary value problem (3.10)–(3.13) in continuum mechanics are presented. The advantage of such spectral method primarily lies in that it allows to conveniently obtain a solution for the displacement and strain fields in response to a body force or an eigenstrain distribution from their convolution with the Green's function. However, although analytical solutions can be obtained in particular cases, the calculation of a convolution is generally a daunting task when performed in the real space. Nevertheless, as noticed in Section 3.3.1, the convolution in the real space reduces to a simple multiplication in the Fourier space. Therefore, in numerical applications whereby the medium is discretized into a regular grid, fast algorithms such as the Fast Fourier Transform (FFT) algorithm – enabling the efficient computation of the discrete Fourier transform and its inverse – can be advantageously used to solve for the boundary value problem (3.10)–(3.13) directly in the Fourier space. Further, the transformation of spatial derivatives in the real space into simple multiplications in the Fourier space allows for a fast and simple numerical solution for the partial differential equations associated with problem (3.10)–(3.13). Consequently, FFT-based techniques provide an alternative full-field method to the FEM approach to solve for partial differential system of equations. Here, the notion of full-field refers to methods in which the

mechanical fields are inherently determined at every grid point – respectively mesh nodes – of the discretized simulation volume V.

Of particular importance, FFT-based and FEM formulations differ in two aspects, namely in their computational complexity and in the treatment of the boundary conditions. The first point is the principal reason spectral methods have received so much attention in the last years. In the standard FEM approach (assuming simple linear elements), a complexity of $\mathcal{O}(N_{tot}^2)$ can be achieved in the most favorable case, where N_{tot} is the total number of elements of the mesh, since the overall complexity in FEM is primarily associated with the cost of the linear solver that is used (see Appendix E). In comparison, for a regular Fourier grid made of the same total number of grid points N_{tot}, i.e. for the same resolution, the complexity drops to $\mathcal{O}(N_{tot} \log N_{tot})$. In this case, the cost is directly that of the efficient FFT algorithm (see Appendix F). Obviously, for large values of N_{tot} , the difference between $N_{tot} \log N_{tot}$ and N_{tot}^2 becomes significant, hence the gain in the computation time up to several orders of magnitudes generally attributed to FFT-based approaches with respect to their FEM counterparts [153]. Note however that more efficient FEM methods such that the hp-FEM that achieves exponential convergence rates exist and should be compared to FFT-based approaches. However, to the author's knowledge, such higher orders FEM methods have not been directly applied to mechanical stress-strain computation yet, probably because of their challenging numerical implementation.

Regarding the treatment of boundary conditions, the FEM method offers a flexible approach in which displacements can be directly prescribed to the mesh nodes and surface forces can be imposed on external surfaces. Thus, in general, mixed and complex boundary conditions can be prescribed, and the presence of surface boundary conditions specifically allows for the treatment of free surfaces. Comparatively, the treatment of boundary conditions in spectral methods offers less degrees of freedom. First, the formulations presented in Sections 3.3.1 and 3.3.2 are given for an infinite

medium. In discrete numerical applications, the use of finite simulation volumes V requires mechanical fields to be fully periodic for the formulation to remain valid (see Appendix F for more details). In that sense, no boundary conditions can be applied on external surfaces, as those do not exist as such. Furthermore, the imposition of the external loading in spectral methods can only be performed by prescription of "macroscopic" average quantities (e.g. average strain or stress, see next sections). Therefore, in FEM approaches, the simulation volume can be regarded as a finite region on which displacements and forces are locally applied, whereas in FFT-based methods, the simulation volume should be seen as a small region of an infinitely repeating domain on which a macroscopic loading is imposed.

The numerical superiority of FFT-based approaches with respect to the widely used FEM approach has led to the development of numerous numerical spectral formulations in various scientific areas where its application is relevant. The first FFT-based formulation for mechanics of materials was introduced by Moulinec and Suquet to efficiently compute the response of composites [178, 179]. It was later extended to polycrystalline simulations [143, 144]. In these works, the spectral formulations presented in Sections 3.3.1 and 3.3.2 were extended to account for inhomogeneous elasticity. In contrast with the the homogeneous formulation, heterogeneous spectral approaches involve iterative schemes, whose enhancement of the convergence has been the subject of many studies [81, 168, 267, 27, 174, 104, 177, 259, 258] (see Section 3.5.1 for the heterogeneous formulation). Note also that very recently, spectral approaches were developed as a way to solve Field Dislocation Mechanics (FDM) [26, 20].

3.4 Homogeneous FFT-based formulation for DDD

3.4.1 Homogeneous formulation

3.4.1.1 FFT-based solution of the boundary value problem

As an alternative full-field stress-strain computational approach, a FFT-based method to be used with the DCM procedure for periodic DDD simulations is developed in this work: henceforth it will be referred to as the DDD-FFT approach [23]. The core idea of the approach relies on coupling the DCM model presented in Section 3.2 with the spectral approach to the eigenstrain theory presented in Section 3.3.2, so as to develop an homogeneous discrete-continuous spectral model of plasticity. With this, it is expected that the DDD-FFT approach will benefit from the computational efficiency of the FFT algorithm while allowing for a discrete representation of dislocations. As mentioned earlier, the determination of the mechanical state throughout the entire simulation volume V_s in DDD simulations is achieved by solving the fundamental mechanical boundary value problem expressed in equations (2.37)–(2.40). As described in Section 3.2.1, the constitutive law $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = \boldsymbol{C} : (\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}))$ is linearized by considering the plastic strain $\boldsymbol{\epsilon}^p$, computed using the DCM, as a constant input of the FFT-based solver at each simulation step. With this, the application of the spectral method to the eigenstrain theory can be directly employed by considering the plastic strain field generated by dislocation motion as a particular inelastic eigenstrain distribution. Under this approximation and using the small strain hypothesis, the local mechanical equilibrium $\sigma_{ij,j}(\vec{\boldsymbol{x}}) = 0$ reads:

$$\begin{aligned} C_{ijkl} u_{k,lj}(\vec{\boldsymbol{x}}) - \varphi_{ij,j}(\vec{\boldsymbol{x}}) = 0 \\ \varphi_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \epsilon^p_{kl}(\vec{\boldsymbol{x}}) \end{aligned} \qquad \forall \vec{\boldsymbol{x}} \in V_s \tag{3.45}$$

where the polarization tensor φ_{ij} is here associated to the plastic strain ϵ^p_{ij}. Assuming the mechanical fields are periodic in the three spatial directions and denoting $\vec{\boldsymbol{\xi}}$ the (frequency) coordinate on the Fourier space, the equilibrium equation (3.45) in the

Fourier space writes:

$$C_{ijkl}\xi_l\xi_j\widehat{u}_k(\vec{\boldsymbol{\xi}}) + \widehat{\varphi}_{ij,j}(\vec{\boldsymbol{\xi}}) = 0 \tag{3.46}$$

where $\widehat{\boldsymbol{u}}$ and $\widehat{\boldsymbol{\varphi}}$ denote the Fourier transforms of the displacement field $\vec{\boldsymbol{u}}$ and the polarization tensor $\boldsymbol{\varphi}$ defined in the real space, respectively. From there, the displacements in the Fourier space can be obtained as:

$$\begin{aligned} \widehat{u}_i(\vec{\boldsymbol{\xi}}) &= -\widehat{G}_{ik}(\vec{\boldsymbol{\xi}})\widehat{\varphi}_{kl,l}(\vec{\boldsymbol{\xi}}) \\ \widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) &= [C_{kjil}\xi_l\xi_j]^{-1} \end{aligned} \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.47}$$

where $\widehat{G}_{ik}(\vec{\boldsymbol{\xi}})$ denotes the periodic Green's function expressed in the Fourier space, whose expression is valid for general anisotropic elasticity. Further, using the small strain compatibility equation (2.3), the strain $\widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}})$ can be directly computed in the Fourier space as:

$$\begin{aligned} \widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}}) &= \frac{1}{2}\left(\widehat{u}_{i,j}(\vec{\boldsymbol{\xi}}) + \widehat{u}_{j,i}(\vec{\boldsymbol{\xi}})\right) \\ &= \frac{1}{2}\left(\xi_l\xi_j\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) + \xi_l\xi_i\widehat{G}_{jk}(\vec{\boldsymbol{\xi}})\right)\widehat{\varphi}_{kl}(\vec{\boldsymbol{\xi}}) \end{aligned} \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.48}$$

where the property of the spatial derivative in the Fourier space expressed in (3.18) has been used. From expression (3.48), the following quantity can be introduced:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}) = \frac{1}{2}\left(\xi_l\xi_j\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) + \xi_l\xi_i\widehat{G}_{jk}(\vec{\boldsymbol{\xi}})\right) \qquad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.49}$$

The fourth-order tensor $\widehat{\Gamma}_{ijkl}$ denotes the modified Green's function in the Fourier space, whose expression in the real space has already been introduced in equation (3.36). With expressions (3.48) and (3.49), the total strain in the Fourier space is simply written as:

$$\begin{aligned} &\widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}}) = \widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}})\widehat{\varphi}_{kl}(\vec{\boldsymbol{\xi}}) \qquad && \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \\ &\widehat{\epsilon}_{ij}(\vec{\mathbf{0}}) = 0 \end{aligned} \tag{3.50}$$

where condition $\widehat{\epsilon}_{ij}(\vec{\mathbf{0}}) = 0$ is enforced by the numerical choice for the modified Green's function at the null frequency $\widehat{\Gamma}_{ijkl}(\vec{\mathbf{0}}) = 0$, for which the Green's function is not defined (the consequence of this choice will be discussed in the following). For DDD-FFT simulations in which the primary volume is discretized into a regular three-dimensional grid of N_i voxels $\{\vec{\boldsymbol{x}}_d\}$ in each direction $i = \{1, 2, 3\}$, the polarization tensor $\boldsymbol{\varphi}(\vec{\boldsymbol{x}}_d)$ related to the plastic strain via equation (3.45) can be determined at every grid point $\vec{\boldsymbol{x}}_d$ in the real space through the DCM procedure described in Section 3.2.1. Consequently, its discrete Fourier transform $\widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)$ at corresponding discrete frequencies $\vec{\boldsymbol{\xi}}_d$ required in the calculation of $\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)$ in equation (3.50) can be easily computed via the use of the FFT algorithm. Conversely, the FFT algorithm can be used to compute the inverse discrete Fourier transform of $\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)$. Note that the solution obtained from taking the inverse Fourier transform of equation (3.50) gives the fluctuations from the mean strain $\langle\boldsymbol{\epsilon}\rangle$, where symbol $\langle\cdot\rangle$ denotes the spatial average over the primary volume V_s. Therefore, by imposing the condition $\widehat{\boldsymbol{\epsilon}}(\vec{\mathbf{0}}) = \mathbf{0}$ on the average strain in equation (3.50), the corresponding strain field $\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d)$ in the real space is obtained by:

$$\epsilon_{ij}(\vec{\boldsymbol{x}}_d) = \mathcal{F}\mathcal{F}\mathcal{T}^{-1}\left(\widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}}_d)\right) + E_{ij} \tag{3.51}$$

where $\mathcal{F}\mathcal{F}\mathcal{T}^{-1}$ denotes the inverse discrete Fourier transform operation performed with the FFT algorithm and $\boldsymbol{E} = \langle\boldsymbol{\epsilon}\rangle$ corresponds to the average imposed macroscopic strain. With this approach, the primary volume can be regarded as being periodically replicated in all spatial directions and immersed in the material bulk subjected to a macroscopic strain $\boldsymbol{E}$.

Besides, note that the formulation in (3.51) is given for a simulation driven in strain. The simulation can be alternatively driven in stress by substituting the imposed value of the macroscopic strain $\boldsymbol{E}$ in equation (3.51) with its resultant when imposing a macroscopic stress $\boldsymbol{\Sigma}$ as:

$$E_{ij} = [C_{ijkl}]^{-1} \Sigma_{kl} + \langle \boldsymbol{\epsilon}^p \rangle_{ij} \tag{3.52}$$

3.4.1.2 Stress and segment force calculations

From equation (3.51), the stress field at each grid point $\vec{\boldsymbol{x}}_d$ is directly obtained from the compatible strain using the constitutive law:

$$\boldsymbol{\sigma}^{FFT}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : (\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)) \tag{3.53}$$

With this formulation, the stress field obtained with the FFT-based approach $\boldsymbol{\sigma}^{FFT}(\vec{\boldsymbol{x}}_d) = \boldsymbol{\sigma}^{int}(\vec{\boldsymbol{x}}_d) + \boldsymbol{\sigma}^{ext}(\vec{\boldsymbol{x}}_d)$ encompasses both the contributions of the dislocations elastic stress field and the stresses arising from the imposed loading conditions without redundancy. Further, by performing a simple interpolation between the grid points, the total stress state can be obtained at every point $\vec{\boldsymbol{x}}$ in the simulation volume. Of particular interest, the total stress can be readily obtained along every dislocation segments such that the Peach-Kohler force $\vec{\boldsymbol{f}}_{ij}^{pk}$ in equation (2.55) can be determined. However, as mentioned in Section 3.2.2, the grid does not account for short-range interactions when using the DCM approach. As a result, a local contribution must be added to the stress in equation (3.53) when computing the Peach-Koehler force along dislocation segments whose neighbors are closer than $h/2$, such that the Peach-Koehler force is readily expressed as:

$$\vec{\boldsymbol{f}}_{ij}^{pk} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\boldsymbol{\sigma}^{FFT}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| + \vec{\boldsymbol{f}}_{ij}^{loc} \tag{3.54}$$

where $\vec{\boldsymbol{f}}_{ij}^{loc}$ denotes the local force arising from the supplementary local interactions,

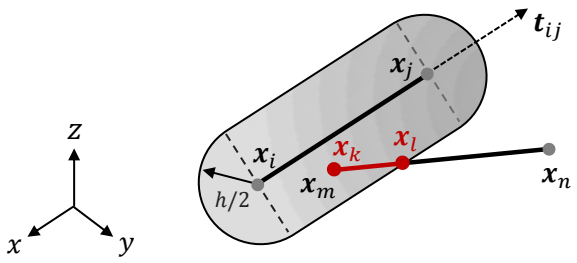


Figure 3.3: Schematic of portions of neighbor segments that must be accounted for as supplementary local interactions. Red sub-segment kl corresponds to the portion of neighbor segment mn whose distance to segment ij is closer than $h/2$. End points $\vec{\boldsymbol{x}}_k$ and $\vec{\boldsymbol{x}}_l$ can be analytically determined by computing the intersections between segment mn and (1) the finite cylinder of radius $h/2$ and of axis $\vec{\boldsymbol{t}}_{ij}$ bounded by normal planes at $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$, and (2) the two half spheres of radius h/2 centered in $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$.

and all other quantities in (3.54) have the same meaning as that used in Section 2.5. By considering all portions kl of neighbor dislocation lines whose distance to dislocation segment ij is smaller than $h/2$, the local force can be expressed as:

$$\vec{\boldsymbol{f}}_{ij}^{loc} = \sum_{kl} \vec{\boldsymbol{f}}_{ij}^{kl} \tag{3.55}$$

where $\vec{\boldsymbol{f}}_{ij}^{kl}$ denotes the interaction force between segment ij and sub-segment kl defined between end nodes $\vec{\boldsymbol{x}}_k$ and $\vec{\boldsymbol{x}}_l$. As depicted in figure 3.3, sub-segments kl correspond to portions of neighbor dislocation segments mn included in the union of spheres of radius $h/2$ centered in $\vec{\boldsymbol{x}}$ when $\vec{\boldsymbol{x}}$ spans dislocation segment ij, i.e. for $\vec{\boldsymbol{x}} = (1-s)\vec{\boldsymbol{x}}_i + s\vec{\boldsymbol{x}}_j, s \in [0,1]$. In order to avoid any computational burden associated with numerical integration, the determination of sub-segments kl is analytically performed by computing the intersections between (1) the finite cylinder of axis $\vec{\boldsymbol{t}}_{ij}$ and bounded at $s = 0$ and $s = 1$, and (2) the two half spheres of radius $h/2$ centered in $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$, respectively. With that, the local force $\vec{\boldsymbol{f}}_{ij}^{kl}$ can be computed by integration of the analytical stress induced by segment kl along segment ij given by Mura formula (2.14), or by the non-singular formulation for isotropic elasticity [5],

as described in Section 2.5.3. Naturally, the calculation of the Peach-Koehler force via expression (3.54) resembles that used in the regular DDD approach when using a long-range approximation method, such as the Box Method or the FMM (see Section 2.9.2.1). However, it is important to keep in mind that the definition of short-range interactions is not the same in both approaches. Specifically, the value of the regularization parameter h, which will be discussed in Section 3.6.1, is on the order of the mesh size in DCM approach, and does not depend on the segment discretization size. This implies that for fine grid resolutions, the calculation of short-range interactions generally does not scale with $\mathcal{O}(N_{nei}^2)$, since only few local interactions are needed to be evaluated. Further, in contrast with the Box Method and the FMM, no approximation is made for the calculation of long-range stress fields, and the latter comes at the same computational cost for isotropic and anisotropic elasticity.

3.4.1.3 Coincidence with analytical stress expressions

Note that the local force $\vec{\boldsymbol{f}}_{ij}^{kl}$ on segment ij cannot be directly obtained by integration of the stress induced by segment kl as expressed in (2.63), but requires an extra factor, for which a justification and an expression are given in the following. As mentioned earlier, the stress field $\boldsymbol{\sigma}^{FFT}$ in equation (3.53) obtained with the FFT-based approach encompasses both the contributions of the applied loading and the internal stress fields induced by the presence of dislocation lines within the simulation volume V_s and its replica. By definition, for any arbitrary dislocation configuration, the stress obtained with equation (3.53) for $\boldsymbol{E} = \boldsymbol{0}$, i.e. in the case where no loading is imposed, corresponds to the internal stresses produced by the dislocation segments in an infinite periodic free medium. Therefore, let us denote $\boldsymbol{\sigma}^{E=0}(\vec{\boldsymbol{x}})$ the stress obtained in (3.53) by the DDD-FFT method under no applied loading. An important characteristic of the present FFT-based formulation is that the internal stress field $\boldsymbol{\sigma}^{E=0}$ is not expected to coincide with the *Mura-based* solution $\widetilde{\boldsymbol{\sigma}}^{int}$ that would be

obtained for the same load-free microstructure. Here the *Mura-based* solution $\widetilde{\boldsymbol{\sigma}}^{int}$ refers to the solution used in regular DDD simulations in which the stress field is obtained by superposition of the stresses induced by individual dislocation segments and their images using the line integral formula (2.14) or the non-singular approach developed by Cai et al. [35] (the non-singular formulation is based on Mura formula and matches it outside of the dislocation cores). As pointed out in Section 3.3.3, this difference finds its origin in the fact that the imposed loading does not have the same physical meaning in both approaches: in the FFT-based formulation, the loading is prescribed by fixing the value of the average strain $\boldsymbol{E} = \langle \boldsymbol{\epsilon} \rangle$ (or stress) corresponding to a macroscopic loading. Thus, a load-free microstructure in the DDD-FFT approach corresponds to a microstructure for which $\boldsymbol{E} = \langle \boldsymbol{\epsilon} \rangle = \mathbf{0}$ is imposed, which is not the case when using the *Mura-based* formulation. In other words, the boundary value problem is not solved for the same boundary conditions in both approaches. As a result, since the only difference between both approaches lies in the value of the average total strain, the stress field $\boldsymbol{\sigma}^{E=0}$ obtained with the FFT-based approach for an arbitrary microstructure differs from the analytical solution $\widetilde{\boldsymbol{\sigma}}^{int}$ by a constant translation tensor $\overline{\boldsymbol{\sigma}}$ such that:

$$\boldsymbol{\sigma}^{E=0}(\vec{\boldsymbol{x}}) = \widetilde{\boldsymbol{\sigma}}^{int}(\vec{\boldsymbol{x}}) + \overline{\boldsymbol{\sigma}} \ , \qquad \forall \vec{\boldsymbol{x}} \in V_s \tag{3.56}$$

Note however that the *Mura-based* and the FFT-based solutions would coincide for a microstructure yielding $\langle \boldsymbol{\epsilon}^p \rangle = \mathbf{0}$. The condition $\langle \boldsymbol{\epsilon}^p \rangle = \mathbf{0}$ physically manifests by the presence of a dipolar configuration: in that case, $\langle \widetilde{\boldsymbol{\sigma}}^{int} \rangle$ reduces to $\mathbf{0}$ in the analytical solution, and as a result $\langle \widetilde{\boldsymbol{\epsilon}} \rangle = \boldsymbol{C}^{-1} : \langle \widetilde{\boldsymbol{\sigma}}^{int} \rangle + \langle \boldsymbol{\epsilon}^p \rangle = \mathbf{0}$ would match the FFT-based solution for $\langle \boldsymbol{\epsilon} \rangle = \boldsymbol{E} = \mathbf{0}$. With that in mind, it follows that $\boldsymbol{\sigma}^{E=0}$ and $\widetilde{\boldsymbol{\sigma}}^{int}$ are not expected to coincide during a DDD-FFT simulation, since the evolution of an arbitrary microstructure does not yield $\langle \boldsymbol{\epsilon}^p \rangle = \mathbf{0}$ in general.

Consequently, for the sake of consistency, both the stress obtained in equation

(3.53) and the *Mura-based* stress must coincide when computing the supplementary local contribution for segments whose interaction distance is smaller than half of the regularization parameter h. In other words, $\overline{\boldsymbol{\sigma}}$ must be determined for the superposition in expression (3.54) to remain valid. Since the translation tensor $\overline{\boldsymbol{\sigma}}$ is constant and hence does not depend on the spatial position on the grid, it can be easily calculated by comparing $\boldsymbol{\sigma}^{E=0}$ and the *Mura-based* stress solution $\widetilde{\boldsymbol{\sigma}}^{int}$ at any arbitrary point $\vec{\boldsymbol{x}}$ in the simulation volume. Therefore, in the current formulation, the determination of $\overline{\boldsymbol{\sigma}}$ requires the computation of the analytical stress solution at one arbitrary point in the volume at each simulation step. As a result, a translation force must be accounted for in the calculation of the local force, such that $\vec{\boldsymbol{f}}_{ij}^{kl}$ in equation (3.55) is decomposed as:

$$\vec{\boldsymbol{f}}_{ij}^{kl} = \widetilde{\vec{\boldsymbol{f}}}_{ij}^{kl} + \overline{\vec{\boldsymbol{f}}}_{ij}^{kl} \tag{3.57}$$

where $\widetilde{\vec{\boldsymbol{f}}}_{ij}^{kl}$, denoting the *Mura-based* nodal force for the pair of segment ij defined between end points $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ and sub-segment kl defined between end points $\vec{\boldsymbol{x}}_k$ and $\vec{\boldsymbol{x}}_l$, is given from expression (2.63) by:

$$\widetilde{\vec{\boldsymbol{f}}}_{ij}^{kl} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\widetilde{\boldsymbol{\sigma}}^{kl}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| \tag{3.58}$$

for which a non-singular analytical expression is given in [5] for isotropic elasticity, and $\overline{\vec{\boldsymbol{f}}}_{ij}^{kl}$ is the constant translation force given by:

$$\overline{\vec{\boldsymbol{f}}}_{ij}^{kl} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\overline{\boldsymbol{\sigma}} \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| = \frac{1}{2} l_{ij} \left[\left(\overline{\boldsymbol{\sigma}} \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] \tag{3.59}$$

where l_{ij} denotes the length of segment ij. When dealing with anisotropic elasticity, nodal force $\widetilde{\vec{\boldsymbol{f}}}_{ij}^{kl}$ in equation (3.58) can be obtained using recent efficient methods based on spherical harmonics expansions [12] (see Appendix B.2) or by numerical integration of the anisotropic stress field $\widetilde{\boldsymbol{\sigma}}^{kl}$ given in equation (2.14) [108].

3.4.1.4 Perspectives

In this section, equations (3.45) to (3.52) establish a complete framework to couple the DCM model presented in Section 3.2 with the very efficient FFT-based solver. As discussed in Section 3.7, such coupling allows for substantial gains in computation time with respect to both the DCM-FEM and the regular approaches. However, besides the prohibitive cost of the FEM procedure, two other important features pertaining to the original DCM approach were pointed out in Section 3.2.2. The first one was related to the expensive regularization procedure involving a numerical integration. To tackle this burden, a fully analytical regularization procedure has been developed and is presented in Section 3.6.1. The other point was related to the accuracy of the DCM model. As mentioned earlier, the history-dependent formulation may lead to an accumulation of errors during the simulation, leading the resulting stress state to eventually depart from the actual dislocation configuration. Obviously, replacing the FEM procedure with a spectral solver has no effect on this behavior, which is inherent to the DCM. Although a quantification of the numerical accuracy of the DCM method is difficult to evaluate in general, it is not expected to become a major issue as long as the regularization procedure remains accurate. However, to overcome this potential issue, an alternative DCM spectral method is proposed here. First, recall that the history-dependent formulation of the DCM model lies in that the plastic strain $\boldsymbol{\epsilon}^p$ directly results from the accumulated motion of dislocations over the time. Such is not the case in the regular DDD approach since the stress state is calculated at each step from the actual spatial position of dislocations. Precisely, the same could be done in the DCM by adopting a linear elastic model so as to directly regularize the dislocation field tensor $\boldsymbol{\alpha}$ associated with the spatial position of dislocation segments. As a matter of fact, the definition of the Burgers vector $\vec{\boldsymbol{b}}$ associated with a closed dislocation line L is given in Section 2.2.1 by:

$$b_i = \oint_{\mathcal{C}} du_i = \oint_{\mathcal{C}} u_{i,j} dx_j \qquad \text{(2.1 repeated)}$$

where $\mathcal{C}$ denotes any circuit irreducibly enclosing the dislocation line. By using the definition of the distortion $\beta_{ji} = u_{i,j}$ introduced in equation (2.7) and applying Stokes' theorem, definition (2.1) may be rewritten as:

$$b_i = \oint_{\mathcal{C}} \beta_{ji} dx_j = \int_S e_{klj} \beta_{ji,l} dS_k \equiv \int_S \alpha_{ki} dS_k \tag{3.60}$$

where S denotes the surface spanning the contour $\mathcal{C}$ and e_{ijk} is the permutation tensor. In equation (3.60), $e_{klj}\beta_{ji,l} = \alpha_{ki}$ provides an expression for the dislocation tensor $\boldsymbol{\alpha} = \mathbf{curl}\, \boldsymbol{\beta}$ [131, 72, 180]. Further, the Burgers vector $\vec{\boldsymbol{b}}$ can be defined as an integral over the same surface S as [139]:

$$b_i = \int_S b_i t_k \delta(\vec{\boldsymbol{L}}) dS_k \tag{3.61}$$

where $\vec{\boldsymbol{t}}$ is the line tangent to the dislocation line L whose position is located by vector $\vec{\boldsymbol{L}}$ and $\delta(\vec{\boldsymbol{L}})$ is the two-dimensional delta function in the plane perpendicular to $\vec{\boldsymbol{L}}$. Since the circuit $\mathcal{C}$ to define $\vec{\boldsymbol{b}}$ is arbitrary, the integrands in (3.60) and (3.61) are necessarily equals, such that one obtains:

$$\alpha_{ki} = b_i t_k \delta(\vec{\boldsymbol{L}}) \tag{3.62}$$

Thus, by regularizing the dislocation tensor $\boldsymbol{\alpha}$ whose components are given in (3.62) on the Fourier grid, the boundary value problem (3.10)–(3.13) would lead to a Poisson-type equation in which the elastic constitutive law is given as $\sigma_{ij} = C_{ijkl}\beta_{kl}$, where $\boldsymbol{\beta}$ is the elastic distortion. Such problem, that can be solved using a FFT-based method, would constitute a direct dynamical extension of the Field Dislocation Mechanics (FDM) whose solution for the static formulation was recently implemented using a

spectral approach [26, 20]. Further, by adopting the Stokes-Helmholtz decomposition of the distortion so as to differentiate between the compatible and the incompatible parts, it is expected that the resort to the translation stress $\overline{\boldsymbol{\sigma}}$ in (3.56) could be eliminated. Note finally that such formulation would become very similar to that used in the Level set method, where the elastic stress fields of the dislocation network are computed using (3.62), but in which the positions of dislocation lines are defined by the intersection of higher-order surfaces [262].

Finally, independently of the formulation used, the principal limitation of the present DDD-FFT approach lies in its prerequisite to deal with fully periodic simulations, i.e. with simulations for which PBC are prescribed in all spatial directions. In other words, the present formulation of the DDD-FFT approach cannot be used to simulate bounded simulations in which free surfaces are to be accounted for. However, the recent development of spectral methods to account for the effects of free surfaces [252] may be adapted to the DDD-FFT approach as a way to overcome the limitation associated with the periodicity. Implementing these is beyond the scope of the present work, and shall be undertaken in a separate study.

In the rest of this chapter, the validity of the DDD-FFT approach is first assessed by application to static and dynamic cases, and the method's ability to treat elastic anisotropy is demonstrated. Then, the current formulation is extended to heterogeneous elasticity in Section 3.5. Finally, the critical aspects of the numerical implementation of the DDD-FFT model are detailed in Section 3.6 while the performance of the approach is investigated and compared with the regular approach in Section 3.7.

3.4.2 Application and validation of the approach

In this section, the validity of the DDD-FFT approach is assessed. First, it is ensured that the internal stresses generated by the presence of one or multiple static dislocation loops in an isotropic medium match the analytical solution. Then, the results for several anisotropic ratios are investigated so as to assess the inherent ability of the method to capture the effects of anisotropic elasticity. Finally, the activation of a Frank-Read source is investigated to ensure the validity of the DDD-FFT approach in dynamic cases.

3.4.2.1 Static dislocations in isotropic elasticity

As a first validation of the DDD-FFT approach, the stress field obtained for a prismatic loop is assessed. For the basis of comparison, the set-up of the simulation is similar to that taken in reference [234] in which the original DCM-FEM method was revisited by its authors. The simulation cell is assumed to be fully periodic, i.e. infinitely replicated in all spatial directions. As depicted in figure 3.4, a prismatic square loop is introduced at the center of the simulation box. The loop is composed of four edge segments of length $l_0 = 1000b$ with Burgers vector $\vec{\boldsymbol{b}} = [010]$ and such that $\|\vec{\boldsymbol{b}}\| = b = 0.25nm$. The four segments are lying on the (010) plane of a simple cubic lattice. The prismatic dislocation loop is introduced in the simulation box following the Volterra-like process: two parallel edge segments of same Burgers vector but opposite line directions are initially introduced at the same position and are then moved apart along the (010) plane until forming the square prismatic loop (see figure 3.4(b)). The material is assumed to be elastically isotropic with shear modulus $\mu = 51GPa$ and Poisson's ratio $\nu = 0.37$. In order for the mesh size L_{mesh} to be identical in each direction, the primary simulation volume V_s is taken as a cube with side $L = 0.5\mu m$ such that $V_s = 0.5\mu m \times 0.5\mu m \times 0.5\mu m$, and discretized into N grid points in each direction, i.e. for a total of $N_{vox} = N \times N \times N$ grid points (or voxels).

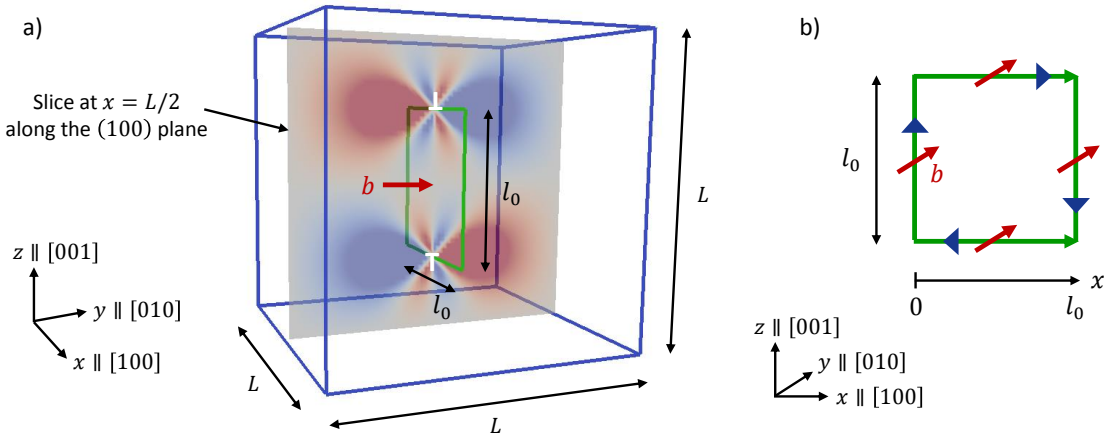


Figure 3.4: Set up used for evaluating the stress field of a prismatic loop using the DDD-FFT approach. Only the primary volume of the periodic simulation is shown here. (a) The prismatic dislocation loop is composed of four edge segments of length $l_0 = 1000b$ with Burgers vector $\vec{\boldsymbol{b}} = [010]$ and lying on the (010) plane. The loop is initially positioned at the center of the simulation box of side length $L = 0.5\mu m$. (b) The prismatic loop is introduced in the simulation following the Volterra-like process: two initial vertical edge segments with same Burgers vector $\vec{\boldsymbol{b}} = [010]$ but opposite line directions are introduced at position $x = 0$. One of the segments is then held at its original position while the second segment is moved apart at a distance l_0 on the (010) plane thereby creating the closed prismatic loop. The blue arrows indicate the line direction of each segment.

The resulting stress field obtained for the prismatic dislocation loop through this process is compared to the analytical solution obtained with the non-singular formulation of Cai and co-workers [35]. Figure 3.5 shows the σ_{23} component of the resulting stress fields on a (100) slice taken at the center of loop ($x = L/2$) for a Fourier grid made of $64 \times 64 \times 64$ voxels. The oscillations observed in figure 3.5(b) pertain to the well-known Gibbs phenomenon associated with spectral methods: this phenomenon occurs in taking Fourier transforms of discontinuous fields. Because of the discontinuities in the plastic strain field induced by dislocation motion – the glide of a dislocation segment produces a jump in the displacement field across its slip plane, see Section 2.2 –, the undesirable Gibbs effect is to be expected.

In order to tackle this undesirable effect, several approaches have been recently

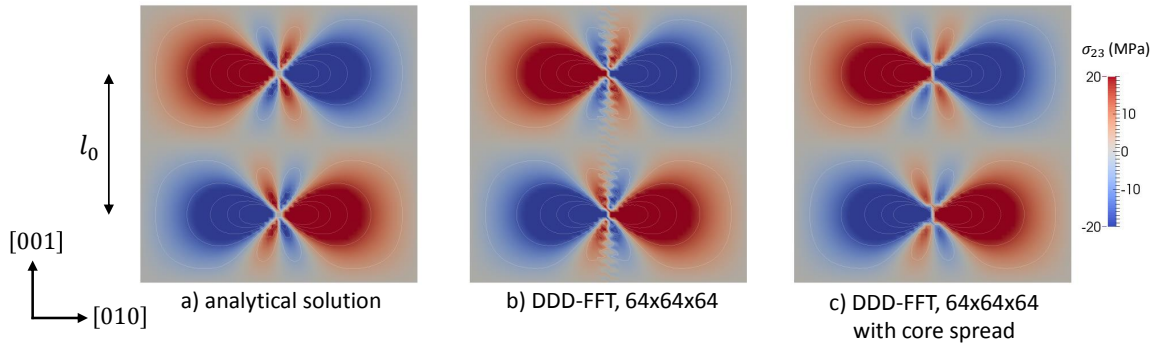


Figure 3.5: σ_{23} shear component of the stress field of a prismatic loop along a (100) slice taken at position $x = L/2$ (see figure 3.4(a)) obtained: (a) using the analytical solution, (b) using the DDD-FFT approach with a grid of $64 \times 64 \times 64$ voxels. Numerical oscillations pertaining to the Gibbs phenomenon are occurring, and (c) using the DDD-FFT approach with a grid of $64 \times 64 \times 64$ voxels and by spreading the regularized plastic strain over $3 \times 3 \times 3$ voxels. Numerical oscillations are removed and the dislocation core spreads as a result.

proposed. For instance, an approach based on a *numerical spreading* was proposed in the work of Brenner et al. [26]. In their two-dimensional implementation of static field dislocation mechanics, the authors proposed to spread any dislocation density $\boldsymbol{\alpha}$ defined at a pixel across its neighboring pixels and studied the effect of different distribution schemes. It was found that spreading the core of a dislocation over 3×3 pixels using a triangular distribution was sufficient to strongly attenuate the oscillations produced from Gibbs phenomenon. As another approach, the use of discrete gradient operators in the Fourier space has provided an efficient manner to attenuate the oscillations [259, 20]. In addition, such approach has been reported to substantially accelerate the convergence of iterative heterogeneous spectral methods [258]. Extensive details on the different solutions for the removal of the Gibbs oscillations are presented in Appendix G.1. In the rest of this section, a method based on a numerical spreading is used, while the approach based on discrete gradients will be presented in details in Section 3.5 when introducing the iterative schemes associated with the heterogeneous DDD-FFT formulation.

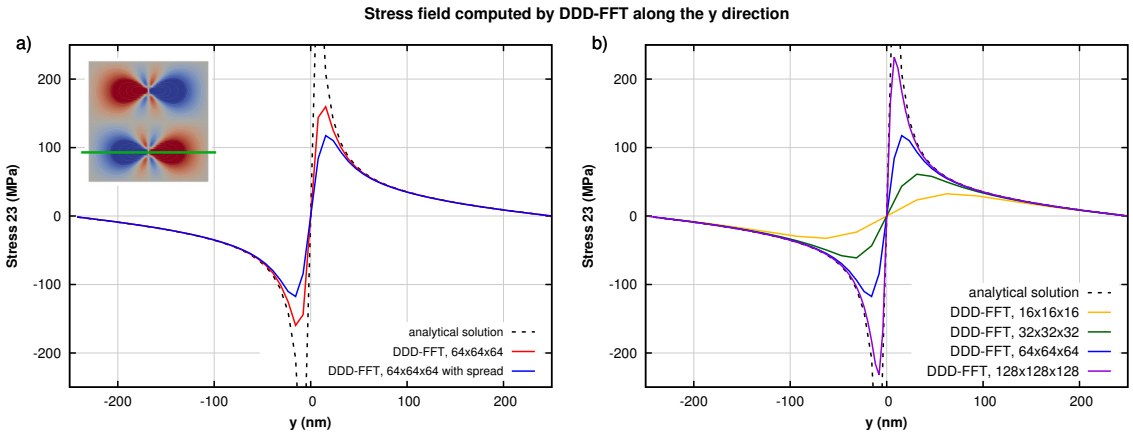


Figure 3.6: Evolution of the σ_{23} shear component of the stress field of a prismatic loop along the green line showed in the inset. In this frame, the origin of the y-axis is aligned with the position of the dislocation core. (a) Comparison between the analytical solution and the DDD-FFT approach with and without *numerical spreading* for a grid of $64 \times 64 \times 64$ voxels. (b) Comparison between the analytical solution and the result obtained with the DDD-FFT approach including the numerical spreading for different grid sizes.

Consequently, a similar *numerical spreading* as that introduced in [26] is used to distribute the plastic strain with a 3D triangular distribution over $3 \times 3 \times 3$ voxels, for which details are provided in Appendix G.1.1. Practically, the plastic shear computed at each grid point from the numerical regularization in equations (3.5) to (3.8) (for which an analytical alternative procedure is developed in Section 3.6) is distributed over the 27 neighboring voxels using a triangular distribution. This numerical spreading is somehow analogous to the isotropic core spread introduced in [35], expect that the core width is related to the mesh size in this case.

The stress field obtained with the spreading is shown in figure 3.5(c). One can observe that the oscillations have been removed and that the dislocation core is apparently spread as a consequence. For the sake of comparison, the stress component σ_{23} is plotted along a line passing through the core of the dislocation, as shown in figure 3.6(a). While the stress fields obtained with the DDD-FFT approach perfectly match the analytical solution away from the core, the numerical spreading naturally

induces a smearing out in the vicinity of the core. Nevertheless, oscillations as shown in figure 3.5(b) are not acceptable in DDD – the stress constitutes the driving force of dislocation segments – and must be eliminated. In other words, if such numerical oscillations were present in the resulting stress field, they would inevitably propagate throughout the motion of dislocations, thus leading to non-physical behaviors, and eventually provoking the collapse of the whole simulation.

In line with the examination of the numerical accuracy of the method, the effect of the grid resolution is plotted in figure 3.6(b) for grids ranging from resolutions of $16 \times 16 \times 16$ to $128 \times 128 \times 128$ voxels. As expected, a finer grid produces more accurate results as we approach the dislocation core. In addition, a finer mesh will induce a smaller spatial spread of the dislocation core. Besides, it must be recalled that from the DCM procedure, the stress field in the vicinity $h/2$ of the core of a dislocation cannot be accounted for by the mesh [234], such such that solely the the *long-range* stress is plotted in figure 3.6. Obviously, providing a careful numerical implementation, the analytical solution in figure 3.6 is recovered in the vicinity $h/2$ of the dislocation core when adding the supplementary local short-range contribution to the *long-range* stress computed with the DDD-FFT approach for all resolutions.

Furthermore, the validity of the DDD-FFT approach must be assessed when multiple dislocations are present in the simulation. To this end, the stress field produced by a dislocation dipole is examined. The prismatic dislocation dipole is created by simultaneously introducing two prismatic loops of opposite Burgers vectors. To ensure that the procedure will be valid in a more general context, the two prismatic loops forming the dipole are positioned in between grid points, respectively at distances $d = L_{mesh}/4$ and $d = 3L_{mesh}/4$ according to the frame depicted in figure 3.7(a). The resulting stress field shown in figure 3.7(b) is seen to perfectly match the analytical solution given in [35] away from the direct vicinity of the dislocation cores. This

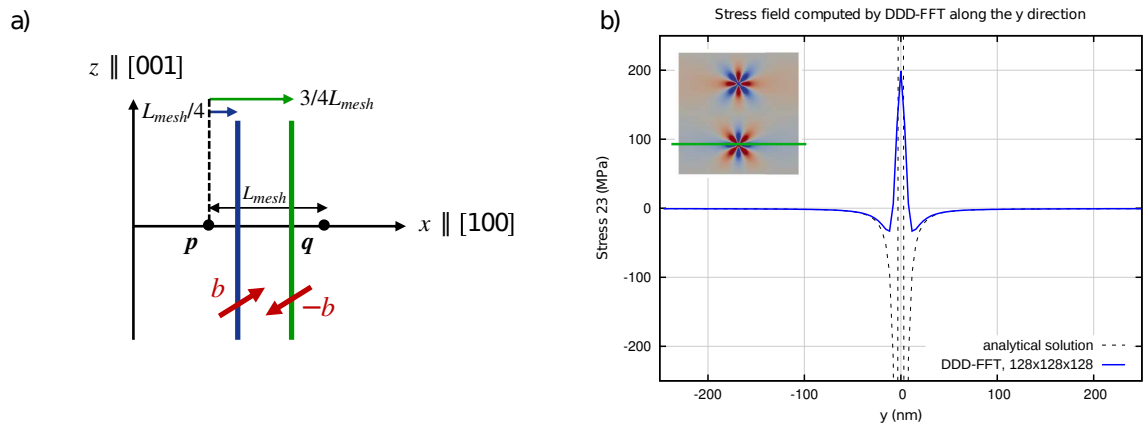


Figure 3.7: (a) Schematic of the positioning of the two prismatic dislocation loops of opposite Burgers vectors $\vec{\boldsymbol{b}} = [010]$ forming a dislocation dipole. (b) Comparison of the evolution of the σ_{23} shear component of the stress field of a prismatic dipole along the green line showed in the inset between the analytical solution and the result obtained with the DDD-FFT approach for a resolution of $128 \times 128 \times 128$ voxels. In this frame, the origin of the y-axis is taken at the middle of the two dislocation loops.

example assesses the ability of the FFT-based technique to achieve direct superposition of the mechanical fields of distinct dislocations, hence validating the DDD-FFT approach to simultaneously handle multiple dislocations.

3.4.2.2 Anisotropic elasticity

As detailed in Section 3.4.1, solving for the mechanical equilibrium in the Fourier space enables for the direct use of the periodic Green's function through its definition, expressed in equation (3.26) and valid for general anisotropy. Therefore, computing the exact stress state – outside of the direct vicinity of the core – associated with periodic dislocations in an anisotropic medium comes at no supplementary cost than for an isotropic medium. In this section the aim is to show the applicability of the DDD-FFT method developed in this work to the case of anisotropic elasticity.

Several measures on the elastic constants can be associated with the deviation from elastic isotropy. Among them, the anisotropy ratio defined as:

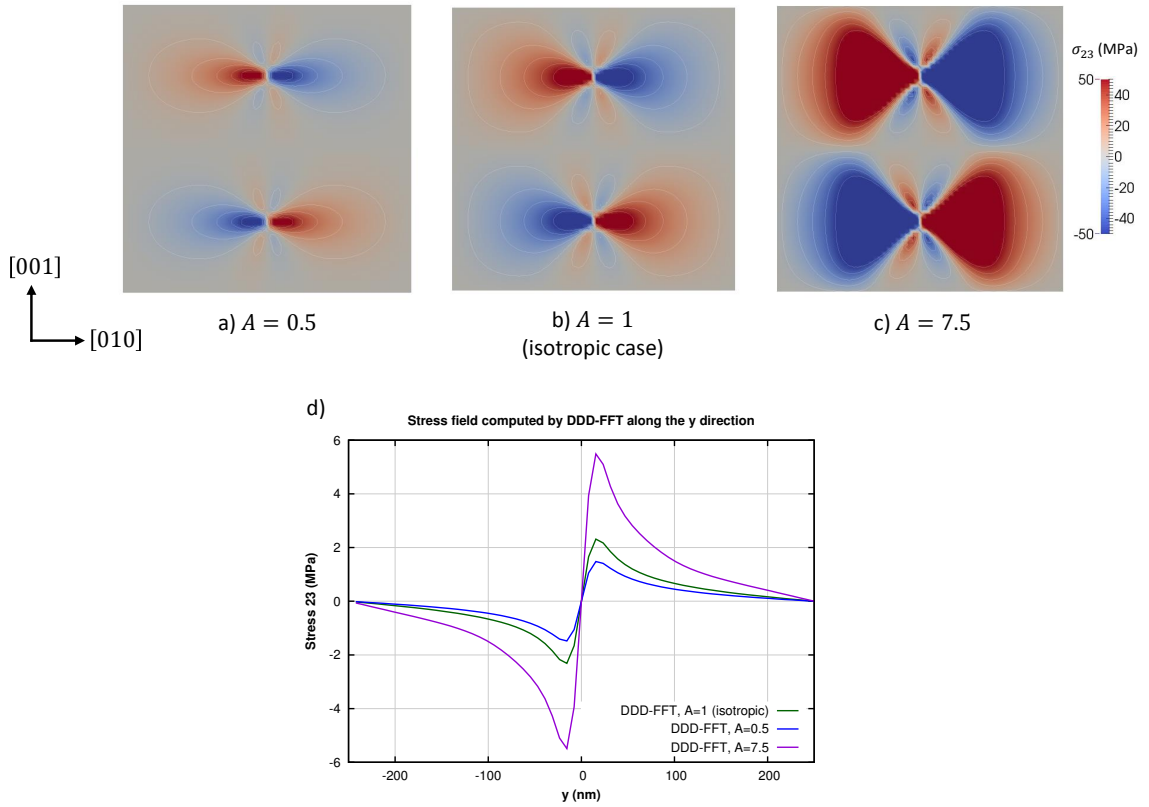


Figure 3.8: σ_{23} shear component of the stress field of a prismatic loop along a (100) slice taken at position $x = L/2$ (see figure 3.4(a)) for different anisotropic ratios: (a) $A = 0.5$ (b) $A = 1$ (isotropic case) and (c) $A = 7.5$. (d) Evolution of the σ_{23} component normalized with $\mu b/l_0$ as a function of the distance to the dislocation core. The results have been obtained for a Fourier grid made of $64 \times 64 \times 64$ voxels. Note that the computational cost associated with cases (a), (b) and (c) is identical when using the DDD-FFT approach.

$$A = \frac{2C_{44}}{(C_{11} - C_{12})} \tag{3.63}$$

is commonly taken as a quantification of the anisotropy of cubic crystals, where $A = 1$ is characteristic for isotropy [111]. For the prismatic loop examined in Section 3.4.2.1, the isotropic material parameters chosen as $\mu = 51GPa$ and $\nu = 0.37$ corresponds to elastic constants of $C_{11} = 247GPa$, $C_{12} = 145GPa$, and $C_{44} = (C_{11} - C_{12})/2 = 51GPa$. In order to provide insights on the current approach's ability to treat anisotropic elasticity, the effect of anisotropy ratios on the resulting

stress of the prismatic loop is compared here. To cover a wide range of possible departures from isotropy, three different anisotropy ratios have been selected: (a) $A = 0.5$ ($C_{44} = 25.5GPa$) which would correspond to Nb crystals [111], (b) $A = 1$ ($C_{44} = 51GPa$) corresponding to an isotropic case, and (c) $A = 7.5$ ($C_{44} = 382GPa$) that is found for α-Fe at high temperatures [12]. The σ_{23} components of the resulting stresses for the configuration depicted in figure 3.4 are reported in figure 3.8. In comparing figures 3.8(a)-(c), the change in shape of the stress fields induced by the variation of the anisotropy ratio is clearly visible, as attested by the isocontours delineated by the thin white lines, going from elliptical-like shape for $A = 0.5$ to triangular-like shape for $A = 7.5$. In figure 3.8(d), where the stresses are normalized by $\mu b/l_0$, the increase in the stress magnitude associated with the increase in the anisotropy ratio is even more apparent.

Of particular interest, note that the computational cost associated with each of these three cases is identical when using the DDD-FFT approach to compute the *long-range* stress fields. Nevertheless, it must be recalled that the DCM procedure requires the calculation of additional local stress fields in the vicinity $h/2$ of the core of a dislocation (see Section 3.4.1). This supplementary contribution is crucial for a proper treatment of junction formation. In the original DCM procedure, the additional intensive computational layer associated with the FEM solver is restraining its use to relatively coarse meshes, thus requiring the computation of a substantial number of local interactions. Conversely, the efficiency of the FFT-based approach allows for the use of finer grids, thus significantly reducing the number of local interaction calculations. Therefore, in the case where the mesh size is chosen sufficiently small such that a marginal number of local interactions have to be computed, the treatment of anisotropic elasticity comes at a roughly equivalent cost as for isotropic elasticity.

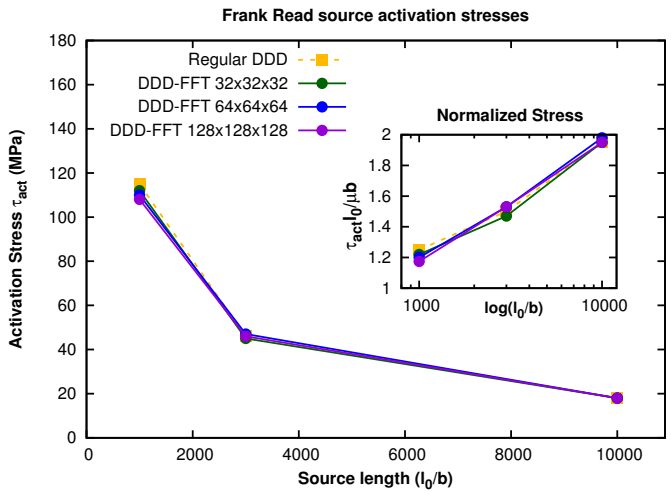


Figure 3.9: Activation stresses τ_{act} of Frank Read sources of different length obtained with the regular DDD approach used in [22] and the DDD-FFT approach for different grid sizes. In the inset, the activation stresses τ_{act} are normalized with $\mu b/l_0$ and reported on a logarithmic scale.

3.4.2.3 Dynamic case: Frank-Read source activation

To assess the validity of the approach in dynamic cases, the activation stress of a Frank Read source is investigated. As mentioned earlier, when using the DCM approach, all dislocations present in the simulation volume must be introduced as loops via the Volterra-like process. One simple way to create a realistic initial microstructure in which dislocation sources are present consists in letting the initial loops relax. During this process, the formation of junctions creates physical pinning points for dislocations, thereby circumventing the necessity of introducing artificial pinning points as usually employed to initiate terminating dislocation sources.

In this section, for the sake of simplicity, the Frank Read source is artificially created by immobilizing three of the four edge segments of the prismatic loop depicted in figure 3.4(a). With this, the only remaining mobile segment will act as an edge source pinned at its end points. Here, the activation stresses of prismatic loops of size ranging from $l_0 = 1000b$ to $l_0 = 10000b$ are examined. The Burgers vector $b = [010]$ is chosen such that $\|\vec{\boldsymbol{b}}\| = b = 0.3nm$ and the loops are introduced into

cubic simulation volumes of sides $a = 5l_0$ discretized into 32^3, 64^3 and 128^3 grid points. Only the bottom edge segment is set mobile such that the Frank Read source is gliding on the (001) plane. The material is assumed to be elastically isotropic with shear modulus $\mu = 130GPa$ and Poisson's ratio $\nu = 0.309$. In this setting, the dislocation source is solely driven by the σ_{23} stress component and produces a ϵ^p_{23} shear. The values obtained for the activation stresses τ_{act} when using the regular DDD code and the DDD-FFT approach are reported in figure 3.9. The results show that the Frank Read source activates at similar stresses in all cases, thereby assessing the ability of the present method to deal with dynamics cases. Essentially, obtaining similar activation stresses as when using the regular DDD method in which internal stresses are analytically calculated demonstrates the capacity of the present approach to efficiently capture the evolution of internal stresses directly from the motion of dislocation segments.

3.5 Heterogeneous FFT-based formulation for DDD

In this section, the spectral method for homogeneous DDD simulations presented in Section 3.4 is extended to heterogeneous elasticity. First, FFT-based formulations for composite materials are reviewed, and the DDD-FFT approach for heterogeneous elasticity is then described in details.

3.5.1 Spectral methods for composite materials

As mentioned earlier, the first use of a FFT-based formulation in continuum mechanics was introduced by Moulinec and Suquet as an efficient novel computational approach to determine the overall and local responses of composites [178, 179]. Essentially, in heterogeneous materials (the notion of *heterogeneity* is here associated to the contrast in the elastic properties of a multi-phase material), the fundamental boundary value problem initially expressed by relations (3.10)–(3.13) becomes:

$$\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = \boldsymbol{C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) \tag{3.64}$$

$$\mathbf{div}\ \boldsymbol{\sigma}(\vec{\boldsymbol{x}}) + \vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}) = \vec{\mathbf{0}} \tag{3.65}$$

$$\vec{\boldsymbol{u}} = \vec{\boldsymbol{u}}^{\,*} \text{ on } \partial V_u \tag{3.66}$$

$$\boldsymbol{\sigma} \cdot \vec{\boldsymbol{n}} = \vec{\boldsymbol{t}}^{*} \text{ on } \partial V_t \tag{3.67}$$

where the constitutive equation (3.64) and the mechanical equilibrium (3.67) hold at each material point $\vec{\boldsymbol{x}}$ in the volume. The main difference with the homogeneous formulation lies in the fact that the value of the stiffness tensor $\boldsymbol{C}(\vec{\boldsymbol{x}})$ in the constitutive law (3.64) is now a function of the spatial position $\vec{\boldsymbol{x}}$. Consequently, the strategy presented in Section 3.3.1 for homogeneous elasticity cannot be directly applied since the expansion of the constitutive law (3.16) in Fourier series will involve a multiplication of Fourier integrals, such that no expression in the Fourier space similar to equation (3.19) can be obtained. To circumvent this issue, Moulinec and Suquet proposed to introduce a reference linear elastic medium with stiffness tensor $\boldsymbol{C}^0$ so as to express the constitutive law (3.64) as:

$$\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = \boldsymbol{C}^0 : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) + (\boldsymbol{C}(\vec{\boldsymbol{x}}) - \boldsymbol{C}^0) : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) \tag{3.68}$$

To retrieve the formalism introduced in Section 3.3.1, constitutive relation (3.68) can be conveniently written as:

$$\begin{aligned}
\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) &= \boldsymbol{C}^0 : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) + \boldsymbol{\tau}(\vec{\boldsymbol{x}}) \\
\boldsymbol{\tau}(\vec{\boldsymbol{x}}) &= \boldsymbol{\delta C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) \\
\boldsymbol{\delta C}(\vec{\boldsymbol{x}}) &= \boldsymbol{C}(\vec{\boldsymbol{x}}) - \boldsymbol{C}^0
\end{aligned} \tag{3.69}$$

where $\boldsymbol{\tau}(\vec{\boldsymbol{x}})$ denotes the inhomogeneous polarization tensor and $\boldsymbol{\delta C}(\vec{\boldsymbol{x}})$ is a fourth-order tensor quantifying the elastic deviation from the reference medium at each

material point $\vec{\boldsymbol{x}}$. Following the same procedure as that detailed in Sections 3.3.1 and 3.3.2 (and neglecting the body force $\vec{\boldsymbol{f}}$), the solution for the strain field is given by (see equation (3.35)):

$$\begin{aligned}
&\text{Real space:} \quad \boldsymbol{\epsilon}^e(\vec{\boldsymbol{x}}) = -\boldsymbol{\Gamma}^0 * \boldsymbol{\tau}(\vec{\boldsymbol{x}}) + \boldsymbol{E} = \mathcal{F}^{-1}\left(\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}})\right) \quad \forall \vec{\boldsymbol{x}} \in D \\
&\text{Fourier space:} \quad \widehat{\boldsymbol{\epsilon}^e}(\vec{\boldsymbol{\xi}}) = -\widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}) : \widehat{\boldsymbol{\tau}}(\vec{\boldsymbol{\xi}}) = -\widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}) : \widehat{\boldsymbol{\delta C} : \boldsymbol{\epsilon}^e}(\vec{\boldsymbol{\xi}}) \quad \forall \vec{\boldsymbol{\xi}} \neq \vec{\boldsymbol{0}}, \quad \widehat{\boldsymbol{\epsilon}^e}(\vec{\boldsymbol{0}}) = \boldsymbol{E}
\end{aligned} \tag{3.70}$$

where $\boldsymbol{\Gamma}^0$ is the modified Green's function associated with the reference medium and for which an analytical expression in the Fourier space for isotropic elasticity is given in equation (3.42), and $\boldsymbol{E}$ is the average imposed strain. Relations (3.70) are analogous to the Lippmann-Schwinger equation in quantum mechanics [130]. Specifically, they provide an implicit expression of the strain $\boldsymbol{\epsilon}^e$, requiring the use of iterative techniques to numerically determine the latter. As a matter of fact, the convergence associated with the determination of problem (3.70) has been the subject of several studies, and different iterative schemes have been proposed [179, 81, 168, 177]. In the general case, it is to be noticed that, provided an appropriate choice for the reference medium $\boldsymbol{C}^0$, the convergence of (3.70) can always be achieved for finite contrasts. Further details and comparisons between the different schemes are provided in reference [177]. In this thesis, the comparison between the schemes and the different strategies that have been employed to accelerate the numerical convergence will be given in Section 3.5.3 for the case of heterogeneous DDD-FFT simulations.

3.5.2 FFT-based formulation for heterogeneous DDD

In Section 3.5.1, the spectral formulation for heterogeneous linear elasticity is presented. However, as mentioned in Section 3.4.1, the stress-strain behavior in DDD-DCM simulations is governed by an elasto-plastic constitutive law. As a result, the constitutive mechanical equilibrium is expressed as:

$$\begin{aligned} \boldsymbol{\sigma}(\vec{\boldsymbol{x}}) &= \boldsymbol{C}(\vec{\boldsymbol{x}}) : (\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}})) \\ \sigma_{ij,j}(\vec{\boldsymbol{x}}) &= 0 \end{aligned} \qquad \forall \vec{\boldsymbol{x}} \in V_s \tag{3.71}$$

where the constitutive law is linearized by regarding the plastic strain $\boldsymbol{\epsilon}^p$, computed using the DCM approach, as a constant input of the FFT-based solver at each simulation time step. Following the procedure presented in the previous section and the introduction of a reference medium defined by $\boldsymbol{C}^0$, the stress can be expressed as:

$$\begin{aligned} \boldsymbol{\sigma}(\vec{\boldsymbol{x}}) &= \boldsymbol{C}^0 : \boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) + \boldsymbol{\tau}(\vec{\boldsymbol{x}}) \\ \boldsymbol{\tau}(\vec{\boldsymbol{x}}) &= \boldsymbol{\delta C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) - \boldsymbol{C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}) \end{aligned} \tag{3.72}$$

where $\boldsymbol{\delta C}(\vec{\boldsymbol{x}})$ has the same definition as in (3.69). In addition to its expression in the linear elastic case, the polarization tensor $\boldsymbol{\tau}(\vec{\boldsymbol{x}})$ accounts for the constant product of the stiffness tensor and the plastic strain. By comparison with its expression (3.45) in the homogeneous case, the polarization tensor – previously denoted $\boldsymbol{\varphi}(\vec{\boldsymbol{x}})$ – is augmented by the product $\boldsymbol{\delta C}(\vec{\boldsymbol{x}}) : \boldsymbol{\epsilon}(\vec{\boldsymbol{x}})$ accounting for the inhomogeneous elasticity distribution. Therefore, following the approach developed by Moulinec and Suquet, the total strain field resulting from the presence of dislocation lines within an elastically heterogeneous DDD simulation volume can be obtained as:

$$\begin{aligned} &\text{Real space:} \quad \boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) = -\boldsymbol{\Gamma}^0 * \boldsymbol{\tau}(\vec{\boldsymbol{x}}) + \boldsymbol{E} = \mathcal{F}^{-1}\left(\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}})\right) \quad \forall \vec{\boldsymbol{x}} \in D \\ &\text{Fourier space:} \quad \widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}) = -\widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}) : \widehat{\boldsymbol{\delta C} : \boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}) + \widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}) : \boldsymbol{C}^0 : \widehat{\boldsymbol{\epsilon}}^p(\vec{\boldsymbol{\xi}}) \quad \forall \vec{\boldsymbol{\xi}} \neq \vec{\boldsymbol{0}}, \quad \widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{0}}) = \boldsymbol{E} \end{aligned} \tag{3.73}$$

Naturally, given that the plastic strain distribution is constant at each time increment, the same iterative schemes as those developed in the linear case can be applied to numerically solve for the total strain implicitly expressed in relations (3.73). However, it is to be noted that the convergence associated with iterative methods to solve for

(3.73) is now expected to depend on both the elastic contrast between the phases – which is expected to be constant for a given simulation – and the distribution of plastic strain – which is directly associated with dislocation motion and therefore evolves during straining.

3.5.3 Iterative and discrete gradient schemes

The development of numerical schemes to improve the convergence of iterative methods dedicated to the implicit formulation in (3.70) (and consequently in (3.73)) has been the subject of many investigations. In this section, the different schemes investigated in this work are presented, while their accuracy and efficiency will be compared in Appendix G.2. Further, the associated algorithms and their numerical implementation will be presented in Section 3.6.3.

3.5.3.1 Heterogeneous iterative schemes

When initially introduced by Moulinec and Suquet, a simple scheme, usually referred to as the *basic* scheme in the literature, was proposed to solve for the heterogeneous linear elastic spectral formulation (3.70) [179]. When applied to the elasto-plastic DDD-FFT formulation (3.73), the *basic* scheme can be written as:

$$\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}) = -\boldsymbol{\Gamma}^0 * \left[\boldsymbol{\delta C} : \boldsymbol{\epsilon}_i - \boldsymbol{C} : \boldsymbol{\epsilon}^p\right](\vec{\boldsymbol{x}}) + \boldsymbol{E} \tag{3.74}$$

where $\boldsymbol{\epsilon}_{i+1}$ denotes the value of the total strain at iteration $i+1$, and where the first term is initialized as $\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}) = \boldsymbol{E}$. Essentially, the *basic* scheme follows from a Neumann series expansion and is numerically analogous to a fixed-point method. As will be detailed in Section 3.6.3 when discussing the numerical implementation of the heterogeneous DDD-FFT method, at each iteration of the *basic* scheme, the convolution in (3.74) is directly calculated as a multiplication in the Fourier space using the FFT algorithm, after which the polarization tensor is calculated back in the

real space using an inverse FFT so as to iterate on real values of $\boldsymbol{\epsilon}_{i+1}$. Although not apparent provided its simplicity, the convergence of the *basic* scheme was established for finite contrasts, and it has further been established that the optimality of the convergence rate is tied to the choice of the reference medium [168]. Thus, for an isotropic medium, the optimal choice is mathematically given for:

$$\begin{aligned} \kappa^0 &= \frac{1}{2}\left(\kappa^{min} + \kappa^{max}\right) \\ \mu^0 &= \frac{1}{2}\left(\mu^{min} + \mu^{max}\right) \end{aligned} \tag{3.75}$$

where $\kappa = \lambda + 2/3\mu$ is the elastic bulk modulus, and where superscripts min and max denote the minimum and maximum values of the constants among the different phases. This choice comes down to select the reference medium as the arithmetic mean of the extremum elastic constants associated with the phases. To the author's knowledge, no study has allowed to determine the optimal reference medium choice in the case of general anisotropy, and it would be assumed that conditions (3.75) can be extended to all elastic constants. Further, in elastic isotropy, a conservative measure of the contrast K between the phases can be given by:

$$K = \max\left(\frac{\kappa^{max}}{\kappa^{min}}, \frac{\mu^{max}}{\mu^{min}}\right) \tag{3.76}$$

The simplicity of the *basic* scheme comes at the cost of a rather slow convergence rate. Thus, when K is sufficiently large, the number of iterations N to reach convergence scales with K. Furthermore, convergence is not ensured in the case of infinite contrasts.

To improve the convergence rate, Eyre and Milton suggested an *accelerated* scheme based on a series expansion of an extended Green's operator [81]. This scheme, originally devised for current conductivity problems, was latter extended to elasticity by Michel et al. [168]. In the case of the elasto-plastic DDD-FFT framework (3.73),

the *accelerated* scheme can be formulated as [168, 258]:

$$\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}) = \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}) + 2\left(\boldsymbol{C} + \boldsymbol{C}^0\right)^{-1} : \boldsymbol{C}^0 : \left(\boldsymbol{E} - \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}) - \boldsymbol{\Gamma}^0 * \left[\boldsymbol{\delta C} : \boldsymbol{\epsilon}_i - \boldsymbol{C} : \boldsymbol{\epsilon}^p\right](\vec{\boldsymbol{x}})\right) \tag{3.77}$$

As a result, a better convergence is obtained than with the *basic* scheme since the number of iterations N required has been demonstrated to be proportional to $\sqrt{K}$ [168]. For this scheme, the optimal choice for the reference medium in elastic isotropy is given by:

$$\begin{aligned} \kappa^0 &= \sqrt{\kappa^{min}\kappa^{max}} \\ \mu^0 &= \sqrt{\mu^{min}\mu^{max}} \end{aligned} \tag{3.78}$$

i.e. the elastic constants should be chosen as geometric means of the extremum properties of different phases. Incidentally, note that along with the introduction of the *accelerated* scheme, Eyre and Milton suggested a grid refinement technique so as to accelerate the convergence rate by progressively seeking the solution on grids with increasing resolutions [81]. However, as for the *basic* scheme, the convergence of the *accelerated* scheme is not ensured for infinite contrasts. To overcome this limitation, other approaches such as the *augmented Lagrangian* scheme [168] have been proposed but are not detailed here. An exhaustive review can be found in reference [177].

As an alternative numerical approach, Zeman and co-workers proposed to solve for the inhomogeneous FFT-based formulation (3.73) using the *conjugate-gradient* method [267]. In the case of an elasto-plastic behavior, the homogenization problem (3.73) can be rewritten as:

$$\begin{aligned} \boldsymbol{\epsilon}(\vec{\boldsymbol{x}}) + \boldsymbol{\Gamma}^0 * \left[\boldsymbol{\delta C} : \boldsymbol{\epsilon}\right](\vec{\boldsymbol{x}}) &= \boldsymbol{\Gamma}^0 * \left[\boldsymbol{C} : \boldsymbol{\epsilon}^p\right](\vec{\boldsymbol{x}}) + \boldsymbol{E} \\ \Leftrightarrow \boldsymbol{A}\left(\boldsymbol{\epsilon}(\vec{\boldsymbol{x}})\right) &= \boldsymbol{B}\left(\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}), \boldsymbol{E}\right) \end{aligned} \tag{3.79}$$

where $\boldsymbol{A}(\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}))$ is a non-linear function of the unknown total strain distribution $\boldsymbol{\epsilon}(\vec{\boldsymbol{x}})$ to solve for, and $\boldsymbol{B}\left(\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}), \boldsymbol{E}\right)$ is a constant function of the plastic strain distribution and the imposed loading. Thus, system (3.79) can be regarded as a system of equation $Ax = b$ that can be solved numerically using the conjugate gradient method. Although form $\boldsymbol{A}$ cannot be expressed explicitly – specifically because it includes a convolution –, the conjugate gradient method can be employed as it solely requires to be provided with a method to compute product Ax for any given x. In that sense, the conjugate gradient method can be used to solve for non-linear systems of equation $A(x) = b$, as long as an explicit formulation of function $A(x)$ is known. In the context of formulation (3.79), function $\boldsymbol{A}(\boldsymbol{X})$ is given for all $\boldsymbol{X}$ as:

$$\boldsymbol{A}(\boldsymbol{X}) = \boldsymbol{X} + \boldsymbol{\Gamma}^0 * [\boldsymbol{\delta C} : \boldsymbol{X}] \tag{3.80}$$

where $\boldsymbol{X}$ represents any vector whose rows are formed by the components of a second-order tensor associated with each voxel of the grid. Mathematically, the convergence of the conjugate gradient method requires a symmetric positive-definite matrix A. Despite the non-trivial functional form of $\boldsymbol{A}(\boldsymbol{X})$ in (3.80), successful applications of the conjugate gradient method have been reported in [267, 27, 28]. For non-symmetric matrices, the biconjugate gradient method making use of the conjugate transpose A^* can be used.

As for the *basic* and *accelerated* schemes, the efficiency of this method stems from the use the FFT algorithm to compute the convolution in equation (3.80) at each iteration of the conjugate gradient method. Note that the faster converging biconjugate gradient stabilized method can also be employed, but requires two evaluations of form $\boldsymbol{A}(\boldsymbol{X})$ per iteration when only one is needed in the regular conjugate gradient. As a result, although a better convergence is generally obtained, the gain in the convergence rate must be balanced with the increase in the numerical cost. In the case where non-linear behaviors are to be considered, such as to consider the

plastic strain $\boldsymbol{\epsilon}^p(\boldsymbol{\epsilon})$ as a function of the strain (or stress), a non-linear extension of the *conjugate-gradient* scheme has been recently proposed [104]. However, as discussed in Section 3.4.1, the plastic strain at each simulation step of DDD-FFT approach is considered as a constant input of the FFT-solver, such that the mechanical behavior in problem (3.73) is linearized. By default, values (3.75) are used for the reference medium. At this point it must also be mentioned that the *conjugate-gradient* method to solve for inhomogeneous FFT-based formulations was simultaneously introduced in the energetic variational framework based on the Hashin and Shtrikman principle developed by Brisard and co-workers [27, 28].

As any iterative methods, the three numerical schemes presented above generate a sequence of improved approximated solutions for the total strain field $\boldsymbol{\epsilon}$. The iterative process is then stopped when a satisfactory approximation of the solution is obtained. To define this, several criteria can be used. In this work, convergence is assumed to be reached when two successive iterations i and $i+1$ produce results such that:

$$\|\boldsymbol{\epsilon}_{i+1} - \boldsymbol{\epsilon}_i\| \leq \epsilon^{tol} \tag{3.81}$$

where ϵ^{tol} is the convergence precision. Other convergence tests such as to compare the deviation from the mechanical equilibrium may be employed [177].

In this work, the two traditional approaches, namely the *basic* and *accelerated* schemes, and the *conjugate-gradient* method – all three methods presented above – will be employed and compared.

3.5.3.2 Discrete gradient schemes

Very recently, the introduction of discrete gradient operators in the Fourier space has permitted significant additional enhancements in the convergence of the iterative schemes. The principle of the discrete derivatives is as follows. If one considers a single Fourier mode $e^{\mathrm{i}\vec{\boldsymbol{\xi}}\cdot\vec{\boldsymbol{x}}}$, its exact derivative with respect to the j-th spatial coordinate

is given by $\mathfrak{i}\xi_j \times e^{\mathfrak{i}\vec{\boldsymbol{\xi}}\cdot\vec{\boldsymbol{x}}}$. However, a discrete approximation can be used, such that the derivative is replaced by $k_j(\vec{\boldsymbol{\xi}}) \times e^{\mathfrak{i}\vec{\boldsymbol{\xi}}\cdot\vec{\boldsymbol{x}}}$ where $\vec{\boldsymbol{k}}(\vec{\boldsymbol{\xi}})$ is called the effective wavenumber associated with the discrete gradient operator. Thus, in a discrete Fourier representation, the compatibility equation $\epsilon_{ij} = 1/2(u_{i,j} + u_{j,i})$ and the mechanical equilibrium $\sigma_{ij,j} = 0$ are expressed as [258]:

$$\begin{aligned} \widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}}) &= \frac{1}{2}\left(k_j(\vec{\boldsymbol{\xi}})\widehat{u}_i(\vec{\boldsymbol{\xi}}) + k_i(\vec{\boldsymbol{\xi}})\widehat{u}_j(\vec{\boldsymbol{\xi}})\right) \\ \widehat{\sigma}_{ij,j}(\vec{\boldsymbol{\xi}}) &= k_j^*(\vec{\boldsymbol{\xi}})\widehat{\sigma}_{ij}(\vec{\boldsymbol{\xi}}) = 0 \end{aligned} \tag{3.82}$$

where $\vec{\boldsymbol{k}}^{\,*}(\vec{\boldsymbol{\xi}})$ is discrete divergence operator corresponding to the complex conjugate of $\vec{\boldsymbol{k}}(\vec{\boldsymbol{\xi}})$. In the case of a centered-scheme (C), one has:

$$k_i^C(\vec{\boldsymbol{\xi}}) = \mathfrak{i}\sin(\xi_i) \tag{3.83}$$

that is obtained from the following spatial finite difference approximation:

$$f_{,i}(\vec{\boldsymbol{x}}) = \frac{f(\vec{\boldsymbol{x}} + \vec{\boldsymbol{\delta}}_i) - f(\vec{\boldsymbol{x}} - \vec{\boldsymbol{\delta}}_i)}{2\delta_i} \tag{3.84}$$

where $\vec{\boldsymbol{\delta}}_i$ is the unit voxel vector in the i-th direction whose magnitude δ_i corresponds to the spacing between subsequent voxels in the i-th direction. For a backward difference scheme (W), the discrete gradient operator is expressed as:

$$k_i^W(\vec{\boldsymbol{\xi}}) = e^{\mathfrak{i}\xi_i} - 1 \tag{3.85}$$

which corresponds to the backward differentiation formula given by:

$$f_{,i}(\vec{\boldsymbol{x}}) = \frac{f(\vec{\boldsymbol{x}}) - f(\vec{\boldsymbol{x}} - \vec{\boldsymbol{\delta}}_i)}{\delta_i} \tag{3.86}$$

The primary valuable consequence following the use of discrete gradient operators is the acceleration of the convergence rates that is observed [258] (see Appendix

G.2). Although its origin results from several factors, the resort to discrete schemes generally produces more accurate local field responses. However, the efficiency of the different methods is specific to the application, and it would be difficult to draw general conclusions on the superiority of one scheme with respect to the others. For instance, while the backward scheme estimates the derivatives more locally than the centered scheme, it also breaks the symmetry and may therefore be inappropriate for certain problems where the symmetry must be respected. Very recently, Willot proposed a new discretization scheme referred to as the rotational scheme (R) [258]. The core idea of this discretization lies in evaluating the displacement fields at the corners of each voxel, while calculating the stress and strain at the center of the voxels. The direction joining each corner of a voxel to its center forms a 45°–rotated basis with respect to the original frame, hence the name of this scheme. In a three-dimensional setting, the resulting discrete gradient operator (R) is obtained as [258]:

$$k_i^R(\vec{\boldsymbol{\xi}}) = \frac{\mathrm{i}}{4}\tan\left(\frac{\xi_i}{2}\right)\left(1+e^{\mathrm{i}\xi_1}\right)\left(1+e^{\mathrm{i}\xi_2}\right)\left(1+e^{\mathrm{i}\xi_3}\right) \tag{3.87}$$

From relations (3.82), the general form of the new discrete modified Green's operator is obtained as:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}) = \left\{ k_j(\vec{\boldsymbol{\xi}}) \left[k_m(\vec{\boldsymbol{\xi}}) C_{kmin} k_n^*(\vec{\boldsymbol{\xi}}) \right]^{-1} k_l^*(\vec{\boldsymbol{\xi}}) \right\}_{\mathrm{sym}}, \quad \forall \vec{\boldsymbol{\xi}} \neq \vec{\mathbf{0}} \tag{3.88}$$

where the only difference from its original continuous counterpart expressed in (3.37) follows from the use of complex conjugates in the effective wavenumbers. This results from the fact that when using discrete differentiations, the gradient operator may lose its pure imaginary character. As a matter of fact, when setting

$$k_i(\vec{\boldsymbol{\xi}}) = \mathrm{i}\xi_i \tag{3.89}$$

in expression (3.88), the continuous modified Green's operator (3.37) is recovered.

In addition to accelerating the heterogeneous iterative schemes, the use of discrete gradients has been reported to further attenuate the spurious Gibbs oscillations associated with FFT-based methods. The effects of the different operators presented in this section on the Gibbs oscillations are fully discussed in Appendix G.1.2.

3.5.4 Validation of the heterogeneous DDD-FFT approach

Assessing the validity of the heterogeneous DDD-FFT framework developed in this work includes two major aspects: (1) the validation of the solution fields obtained for heterogeneous elasto-plastic problems, and (2) the investigation of the accuracy of the discrete and iterative schemes presented in Section 3.5.3.

The first aspect cannot be validated directly since, to the author's knowledge, no analytical solution exists for elasto-plastic problems in heterogeneous elasticity, and current simulation techniques do not offer sufficient accuracy to establish a trustful comparison. As a result, the heterogeneous formulation has been validated for elastic problems, for which analytical solutions are available and comparisons with other numerical works are feasible. For such purpose, and to incorporate the examination of the discrete and iterative schemes independently, an incremental validation approach has been used. Thus, the accuracy of the discrete gradient schemes have been first investigated for the case of a static dislocation loop in an homogeneous medium. Details and results related to this aspect are provided in Appendix G.1.2. Then, the validity of the solution of elastic problems in heterogeneous media has been assessed, and the impact of the discrete and iterative schemes on the convergence rate has been investigated. Full details are provided in Appendix G.2.

Consequently, by virtue of the superposition principle – the plastic strain distribution is considered as a constant input of the elastic Lippmann-Schwinger problem in the heterogeneous DDD-FFT formulation, see equation (3.73) –, independent validations of the homogeneous elasto-plastic (see Section 3.4.2) and heterogeneous elastic

formulations demonstrate the validity of the heterogeneous elasto-plastic DDD-FFT method. Application of the heterogeneous DDD-FFT approach, i.e. when accounting for the presence of dislocations in heterogeneous elastic media, will be presented Section 5.2 when investigating the interactions between dislocations and second-phase particles.

3.6 Numerical implementation

Although the development of the DDD-FFT method may appear as relatively simple from the formulations given in Sections 3.4.1 and 3.5.2, its numerical implementation requires special care and involves several challenges. As already stated, one of the main challenges associated with spectral methods lies in the treatment of the spurious numerical oscillations pertaining to the Gibbs effect, and whose sole presence would inevitably result in the collapse of DDD-FFT simulations. Although such aspect has been one of the principal difficulties to be addressed during the development of the DDD-FFT model, the different solutions that were implemented have already been presented in Sections 3.4.2 and 3.5.3, and are extensively discussed in Appendix G.1. Therefore, they are not further presented in this section.

Consequently, the focus of this section is placed on the implementation of a novel analytical regularization approach to replace the prohibitive original DCM regularization procedure, and on the coupling between the regular DDD framework developed in this work and presented in Chapter 2 and the FFT-based solver for homogeneous and heterogeneous elasticity. Particular attention will be given to the strategy adopted for the parallel computing implementation, while the development of a GPU-accelerated version of the DDD-FFT method is presented in Appendix G.4.

3.6.1 Analytical regularization procedure

While the use of a FFT-based solver has permitted to remove the burden associated with the first aspect, the cost of the numerical regularization procedure as originally

introduced by its authors would have certainly prevented the DDD-FFT method to become competitive with regular DDD simulations. Therefore, a new efficient analytical regularization procedure is developed in this work.

Following the DCM method described in Section 3.2.1, the plastic shear $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ produced by the glide of a dislocation segment ij with Burgers magnitude b can be regularized at each grid point $\vec{\boldsymbol{p}}$ according to equation (3.7) as:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}} = \frac{6b}{\pi h^3} dS_{ij}^{\vec{\boldsymbol{p}}} = \frac{b}{V_e} dS_{ij}^{\vec{\boldsymbol{p}}} \qquad \text{(3.7 repeated)}$$

where $V_e = \pi h^3/6$ is the volume of the elementary spherical sheared volume $d\phi(\vec{\boldsymbol{x}})$ of radius $h/2$. In refrence [234], the computational cost of the regularization procedure stems from the the numerical integration procedure employed to determine quantity $dS_{ij}^{\vec{\boldsymbol{p}}}$. Thus, as indicated by equation (3.6), the value of the incremental surface $dS_{ij}^{\vec{\boldsymbol{p}}}$ is numerically determined by discretization of the total glide area dA_{ij} produced by motion of dislocation segment ij into elementary areas dS, such that:

$$dS_{ij}^{\vec{\boldsymbol{p}}} = \int_{dA_{ij}} \chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}}) d\vec{\boldsymbol{x}} \qquad \text{(3.6 repeated)}$$

where $\chi(\vec{\boldsymbol{p}}, \vec{\boldsymbol{x}})$ is the characteristic function associated with $dS(\vec{\boldsymbol{x}})$ when $\vec{\boldsymbol{x}}$ spans the area swept by dislocation segment ij, and that takes value 1 when integration point $\vec{\boldsymbol{p}}$ lies into $d\phi(\vec{\boldsymbol{x}})$, 0 otherwise (see figure 3.2). In addition to its cost, the original regularization procedure is also limited to the case of trapezoidal glide areas associated with segment-based discretization approaches in which dislocation segments do not rotate in their slip plane, and would therefore not be applicable to nodal DDD representations.

Therefore, in this thesis, an alternative analytical method to perform the integration in equation (3.6) that should be suitable for general dislocation motions in nodal-based discretization schemes – i.e. compatible with rotation of dislocation segments

in their slip plane – is proposed. The main idea behind this alternative regularization procedure results from the observation that computing $dS_{ij}^{\vec{\boldsymbol{p}}}$ is mathematically equivalent to computing the intersection between the sphere $d\phi(\vec{\boldsymbol{p}})$ of radius $h/2$ centered at integration point $\vec{\boldsymbol{p}}$ and the quadrilateral defined by the segment sheared area defined in figure 3.1(a). In the formulation, $\vec{\boldsymbol{p}}$ now refers to the position of any point of the Fourier grid. From a purely geometrical perspective, the intersection between a sphere and a quadrilateral reduces to an in-plane intersection between a circle – resulting from the intersection between the sphere and the plane containing the quadrilateral – and the quadrilateral itself. As result, the contour of such intersection is exclusively composed of a succession of straight segments and arcs, as depicted in figure 3.10. Following that, $dS_{ij}^{\vec{\boldsymbol{p}}}$ can be analytically computed by line integration using Green's theorem:

$$dS_{ij}^{\vec{\boldsymbol{p}}} = \frac{1}{2} \oint_{\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}} (-ydx + xdy) \tag{3.90}$$

where $\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}$ denotes the closed contour defined by the intersection between the quadrilateral defined by the motion of segment ij and the sphere $d\phi(\vec{\boldsymbol{p}})$ centered in grid point $\vec{\boldsymbol{p}}$, and x and y are the coordinates spanning the contour in the two-dimensional frame defined in the dislocation glide plane. Full details on the Green's theorem and on the derivation of equation (3.90) are provided in Appendix G.3. For a closed contour formed of n successive straight segments and arcs, expression (3.90) can be further decomposed as the summation of individual line integrals (see figure 3.10(b)):

$$dS_{ij}^{\vec{\boldsymbol{p}}} = \left| \sum_{k=1}^{n} \frac{1}{2} \oint_{\mathcal{C}_k} (-ydx + xdy) \right| = \left| \sum_{k=1}^{n} I_{\mathcal{C}_k} \right| \tag{3.91}$$

where $\{\mathcal{C}_k\}_{k=1,n}$ denotes the piecewise continuous set of individual curves defining the entire contour $\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}$, and where the absolute value is taken so as to avoid dealing with the difficulty associated with the orientation of the contour. For any straight

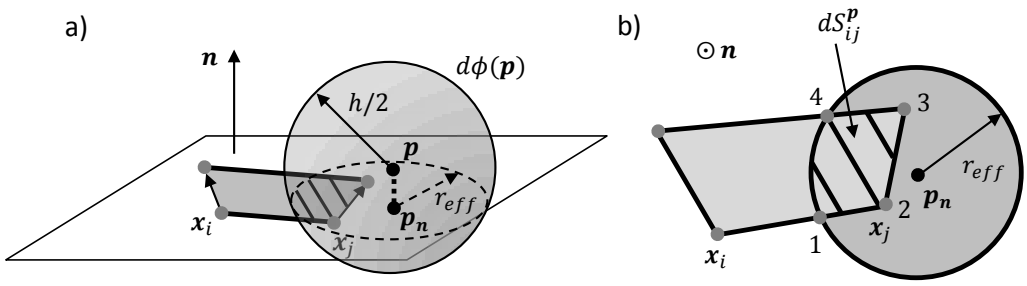


Figure 3.10: (a) Example of the intersection between the area swept by a dislocation segment ij gliding on plane with unit normal $\vec{\boldsymbol{n}}$ and the elementary sphere $d\phi(\vec{\boldsymbol{p}})$ of radius $h/2$ centered in grid point $\vec{\boldsymbol{p}}$. The intersection between the sphere $d\phi(\vec{\boldsymbol{p}})$ and the glide plane of the dislocation results in a circle of radius $r_{eff} = \sqrt{h^2/4 - d^2}$ and of center $\vec{\boldsymbol{p}}_n$ where $\vec{\boldsymbol{p}}_n$ is the orthogonal projection of $\vec{\boldsymbol{p}}$ onto the dislocation plane such that $\vec{\boldsymbol{p}} - \vec{\boldsymbol{p}}_n = d\vec{\boldsymbol{n}}$. (b) The intersection area $dS_{ij}^{\vec{\boldsymbol{p}}}$ (shaded region) can be analytically calculated using Green's theorem by following the oriented contour composed of the straight segments $\overline{12}$, $\overline{23}$ and $\overline{34}$, and the arc $\widehat{41}$.

segment $\mathcal{C}_k$ defined between vertices (x_0, y_0) and (x_1, y_1), the line integral $I_{\mathcal{C}_k}$ along this segment can be analytically derived from equation (3.90) as:

$$I_{\mathcal{C}_k}^{seg} = \frac{1}{2}\left(x_0 y_1 - y_0 x_1\right) \tag{3.92}$$

Similarly, for any arc $\mathcal{C}_k$ defined as a portion of a circle of radius r centered in (x_c, y_c) and delimited by end vertices (x_0, y_0) at angle θ_0 and (x_1, y_1) at angle θ_1 (assuming $\theta_1 > \theta_0$), the line integral $I_{\mathcal{C}_k}$ along this arc can be analytically derived from equation (3.90) as:

$$I_{\mathcal{C}_k}^{arc} = \frac{1}{2}\left[r^2(\theta_1 - \theta_0) + x_c(y_1 - y_0) - y_c(x_1 - x_0)\right] \tag{3.93}$$

Thus, expressions (3.92) and (3.93), for which derivations are given in Appendix G.3.1, provide fully analytical solutions to compute the intersection area $dS_{ij}^{\vec{\boldsymbol{p}}}$ with equation (3.91), provided that the individual pieces $\{\mathcal{C}_k\}_{k=1,n}$ forming the contour $\mathcal{C}$ bounding region $dS_{ij}^{\vec{\boldsymbol{p}}}$ are determined. Such contour is constructed by ordering and joining the quadrilateral vertices and the circle-quadrilateral intersection points

through straight segments and arcs (see figure 3.10(b)). Note that this contour can be directly obtained by using computational geometry libraries such as CGAL [42]. In the present work however, the numerical algorithm presented in Appendix G.3.2 is implemented.

With that, the amount of shear produced by each dislocation segment at every voxel $\vec{\boldsymbol{p}}$ of the Fourier grid can be calculated. However, as discussed in Section 3.2.1, a critical aspect of the regularization procedure in terms of validity and the accuracy is the value of the regularization parameter h. In the novel analytical formulation proposed in this work, this translates by the fact that the value of $dS_{ij}^{\vec{\boldsymbol{p}}}$ is a direct function of h, as h corresponds to the diameter of the elementary sphere $d\phi(\vec{\boldsymbol{p}})$ associated with each grid point $\vec{\boldsymbol{p}}$ such that it defines the effective radius r_{eff} (see figure 3.10). In the current implementation, the value of h must be chosen such that the union of the spheres associated with every grid point at least maps the entire simulation volume. In other words, care must be taken to ensure that the entire sheared area produced by any dislocation segment intersects with elementary spheres, such that the total plastic strain produced is entirely transferred to the mesh. In other words, this issue is related to the concept of partition of unity.

With that, the minimum acceptable value for any regular three-dimensional mesh of size L_{mesh} (i.e. for which the distance between two consecutive grid points is L_{mesh} in each direction) is $h = \sqrt{3}L_{mesh}$ (see figure 3.11). Note that for such value of h, an overlap between the elementary spheres exists, resulting in a smearing out of the plastic strain, i.e. in a loss of accuracy. While in [147], the value $h = 3/2L_{mesh}$ has been determined to be optimal and the same value has been used in [234], the analytical regularization procedure introduced in the present work induces a dependency of $dS_{ij}^{\vec{\boldsymbol{p}}}$ on the position of the dislocation line with respect to the grid (this will be shown later in figure 3.13). If the dislocation core is perfectly aligned with grid points (red dislocation line in figure 3.11), the obtained stress field is similar to the analytical

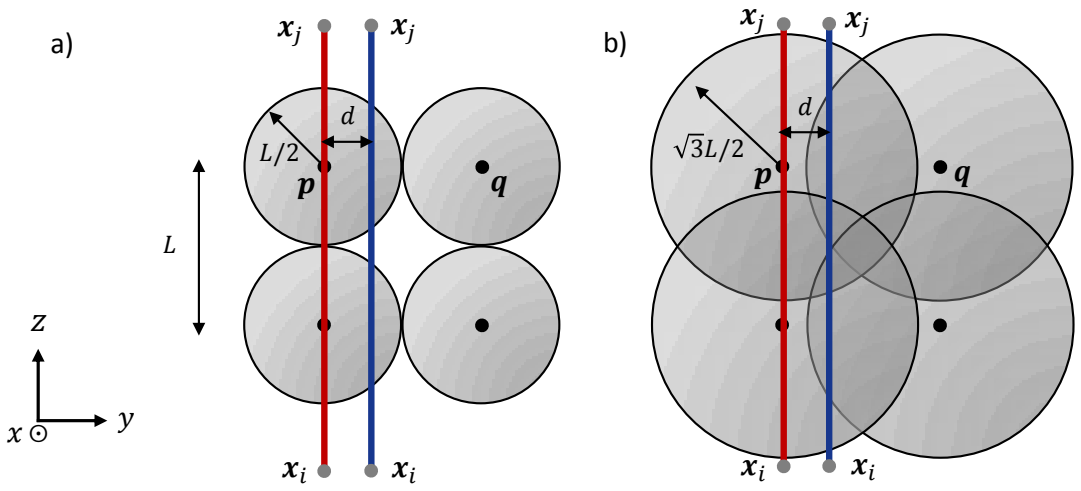


Figure 3.11: Slice in the (yz) plane of a dislocation segment ij shearing the volume in the (xz) plane. The red dislocation segment is aligned with grid point $\vec{\boldsymbol{p}}$ while the blue segment is away from a distance d. (a) For $h = L = L_{mesh}$ the union of all elementary spheres centered in grid points does not map the entire volume. As a result, the plastic shear induced by the blue dislocation segment ij is not entirely transferred to the mesh, leading to inaccurate results. (b) For $h = \sqrt{3}L_{mesh}$ the union of all elementary spheres maps the entire volume, such that the entire plastic strain is transferred to the mesh. However, the overlapping between the elementary spheres results in a smearing out of the plastic strain that needs to be corrected for.

solution; however inaccuracies occur when the dislocation core lies in between two grid points (blue dislocation line in figure 3.11). For this reason, in the following, we propose a method to correct the plastic shear distribution depending on the position of the dislocation with respect to the grid.

According to the description of the regularization procedure, the position dependency of the regularized plastic strain may originate from two sources: (1) the overlap between contiguous elementary spheres, and (2) the evolution of the regularized plastic strain as a function of the distance between the grid points and the dislocation core. Recall from equation (3.7) that if $V_e = \pi h^3/6$ is the volume of the elementary sphere $d\phi(\vec{\boldsymbol{p}})$ of radius $h/2$ around each grid point, the shear $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ induced by the glide of dislocation segment ij at grid point $\vec{\boldsymbol{p}}$ is writes:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}} = \frac{6b}{\pi h^3} dS_{ij}^{\vec{\boldsymbol{p}}} = \frac{b}{V_e} dS_{ij}^{\vec{\boldsymbol{p}}} \qquad \text{(3.7 repeated)}$$

where $dS_{ij}^{\vec{\boldsymbol{p}}}$ is the intersection area defined in figures 3.10 and G.9. To circumvent the first source of position-dependency, let us use an elementary volume such that the union over all grid points exactly maps the simulation volume, without inducing any overlapping. As illustrated on figure 3.12, only an elementary volume chosen as a square box of side $a = L_{mesh}$ would satisfy such a mapping. With a square box, the plastic strain $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ induced by a dislocation segment ij entirely shearing the elementary box (centered at grid point $\vec{\boldsymbol{p}}$) along the (xz) plane would amount to:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}} = \frac{bL_{mesh}^2}{L_{mesh}^3} = \frac{b}{L_{mesh}} = d\gamma_{ij}^{ref} \qquad (3.94)$$

This result is simply obtained from equation (3.7) with $dS_{ij}^{\vec{\boldsymbol{p}}} = L_{mesh}^2$ and $V_e = L_{mesh}^3$ for a dislocation segment entirely shearing a square box of side length L_{mesh}. However, as depicted in figure 3.12(a), with such elementary volume, a red dislocation segment aligned with a grid point $\vec{\boldsymbol{p}}$ will yield the same plastic strain $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ at point $\vec{\boldsymbol{p}}$ as a blue dislocation segment positioned at a distance d from the grid point, since the intersection area $dS_{ij}^{\vec{\boldsymbol{p}}} = L_{mesh}^2$ is the same in both cases. In other words, using elementary square boxes leads to disregard the spatial positioning of the dislocation core for any dislocation lying at distance $\pm d$ from a grid point. This issue is precisely related to second source of position dependency, namely the evolution of $d\gamma_{ij}^{\vec{\boldsymbol{p}}}$ with respect to the distance d from the grid point $\vec{\boldsymbol{p}}$. Note that for the rest of this section, $d\gamma_{ij}^{ref}$ as defined in equation (3.94) will denote the reference amount of regularized plastic shear required to obtain an exact stress field for any red dislocation aligned with the grid, i.e. for which $d = 0$.

However, a linear interpolation of the plastic shear with respect to the distance d between the dislocation plane and the grid point ($0 < d < L_{mesh}$) is seen to

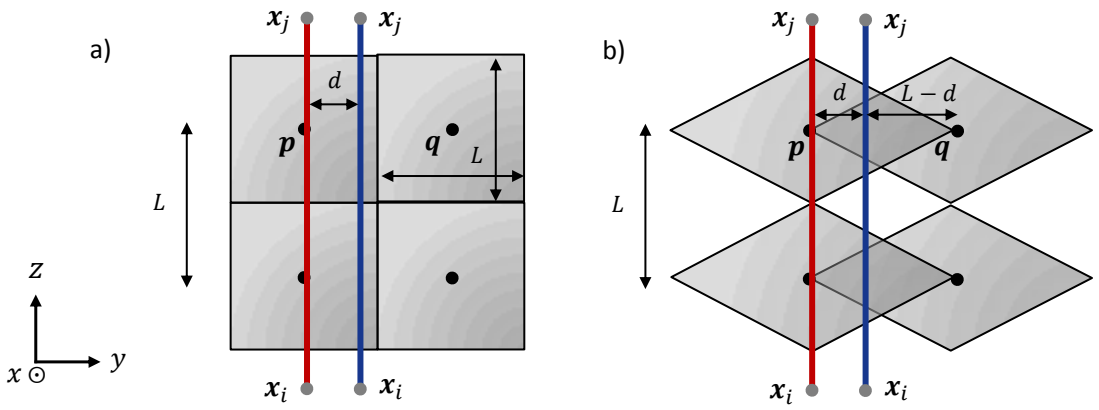


Figure 3.12: Schematic of two dislocation segments shearing the volume along the (xz) plane. The red dislocation segment is aligned with grid point $\vec{\boldsymbol{p}}$ while the blue segment is positioned at a distance d from grid point $\vec{\boldsymbol{p}}$, hence at distance $L_{mesh} - d$ from the subsequent grid point $\vec{\boldsymbol{q}}$. (a) When the elementary volumes associated to each grid point are chosen to be square boxes of side $a = L_{mesh}$, the simulation volume is entirely mapped and no overlapping is present. However both red and blue dislocation segments would induce the same plastic shear $d\gamma_{ij}^{\vec{\boldsymbol{p}}} = b/L_{mesh}$ (see equation (3.94)) at grid point $\vec{\boldsymbol{p}}$, such that the spatial positioning of the dislocation core is not properly accounted for. (b) A linear interpolation of the shear strain distribution with respect tot he core position can be achieved by using diamond-shaped elementary volumes. However such shape is not directly extensible in three dimensions and would produce inaccurate results for tilted dislocations in the (yz) plane.

yield correct stress field values (this will be later shown in figure 3.13). Such linear interpolation between two successive grid points can be illustrated by the use of diamond-shaped elementary volumes as depicted in figure 3.12(b). For the sake of clarity, when considering a linear interpolation, a blue dislocation segment located at a distance d from grid point $\vec{\boldsymbol{p}}$ – and consequently at distance $L_{mesh} - d$ from subsequent grid point $\vec{\boldsymbol{q}}$ – yields a plastic strain $d\gamma_{ij}^{\vec{\boldsymbol{p}}} = (L_{mesh} - d)/L_{mesh} \cdot d\gamma_{ij}^{ref}$ at point $\vec{\boldsymbol{p}}$ and $d\gamma_{ij}^{\vec{\boldsymbol{q}}} = d/L_{mesh} \cdot d\gamma_{ij}^{ref}$ at point $\vec{\boldsymbol{q}}$. Note that, as required, a red dislocation segment located at $d = 0$ from grid point $\vec{\boldsymbol{p}}$ yields $d\gamma_{ij}^{\vec{\boldsymbol{p}}} = d\gamma_{ij}^{ref}$ and $d\gamma_{ij}^{\vec{\boldsymbol{q}}} = 0$. However, the elementary volume illustrated in figure 3.12(b) is only depicted in the (yz) plane and is not straightforwardly extensible to the three-dimensional space. Furthermore, the procedure would collapse for any dislocation segment tilted on the (yz) plane, as

the cross-sectional area of such shape is not solely a function of the distance d from its center, but also of the orientation of the intersecting plane. With that in mind, it seems inevitable that the shape of the elementary volume should be a sphere. This is because partitioning the volume with spheres is the sole choice that would ensure that the method remains valid regardless of orientation of the dislocation glide plane. Using expression (3.7) and considering a sphere of radius r fully sheared, the evolution of $d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d)$ as a function of the distance d to grid point $\vec{\boldsymbol{p}}$ can be expressed as:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d) = \frac{b\pi r_{eff}^2}{(4/3)\pi r^3} = \frac{b(r^2 - d^2)}{(4/3)r^3} \tag{3.95}$$

where $r_{eff} = \sqrt{r^2 - d^2}$ denotes the effective radius of the circle resulting from the intersection between the elementary sphere $d\phi(\vec{\boldsymbol{p}})$ and the glide plane of the dislocation, as depicted in figure 3.10. Clearly, when using a sphere as the elementary volume, $d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d)$ is not a linear function of d. However, it can be corrected such that $d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(d)$ becomes a linear function of d. To obtain a linear interpolation, such corrected function must be bounded by $d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(0) = d\gamma_{ij}^{ref}$ and $d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(L_{mesh}) = 0$. In order to fulfill these requirements, the sphere radius can be conveniently chosen as $r = L_{mesh}$, i.e. $h = 2r = 2L_{mesh}$. With that, expression (3.95) rewrites:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d) = \frac{b(L_{mesh}^2 - d^2)}{(4/3)L_{mesh}^3} \tag{3.96}$$

The linear interpolation with respect to d is obtained for $0 < d < L_{mesh}$ if and only if $d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(d) = (L_{mesh} - d)/L_{mesh} \cdot d\gamma_{ij}^{ref}$. Using equation (3.96) and denoting $c(d)$ the correction function defined such that $d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(d) = c(d) \cdot d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d)$, one can write:

$$d\gamma_{ij}^{\vec{\boldsymbol{p}},corr}(d) = c(d) \cdot d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d) = \frac{L_{mesh} - d}{L_{mesh}} d\gamma_{ij}^{ref} \tag{3.97}$$

Finally, using the definitions of $d\gamma_{ij}^{ref}$ and $d\gamma_{ij}^{\vec{\boldsymbol{p}}}(d)$ from equations (3.94) and (3.96), the correction function $c(d)$ can be determined from equation (3.97) as:

$$c(d) = \frac{4}{3}\frac{(L_{mesh}^2 - L_{mesh}d)}{(L_{mesh}^2 - d^2)} \tag{3.98}$$

Essentially, equation (3.97) states that for $h = 2L_{mesh}$, the plastic strain increment $d\gamma_{ij}^{\vec{p}}$ produced by any dislocation shearing the spherical elementary volume $d\phi(\vec{p})$ needs to be corrected by a position dependent factor $c(d)$ whose expression is given in equation (3.98).

To assess the validity of this correction, the stress field of the prismatic loop depicted in figure 3.4 is compared to the analytical solution when the loop is positioned at three different locations with respect to the grid. The three positions chosen for the dislocation loop lying on the (010) plane are illustrated in figure 3.13(a) and the resulting stress fields obtained with and without the correction are reported in figure 3.13(b). As attested by the dashed lines, it clearly appears that the resulting stress field deviates from the analytical solution depending on the position of dislocation with respect to the grid when the correction introduced in equations (3.97) and (3.98) is not accounted for. Conversely, the resulting stress fields superimpose irrespective of the position in the grid and converge towards the analytical solution when the plastic shear transfer is corrected, as shown by the solid lines.

With such correction in the regularization procedure, the DDD-FFT approach is found to provide an exact match with the analytical solution for the stress field of dislocation segments, irrespectively of their position and orientation on the Fourier grid, as attested in Section 3.4.2. Therefore, the diameter of the elementary spheres associated with every grid point is set to $h = 2L_{mesh}$ in the DDD-FFT approach and the regularized plastic strain computed with expressions (3.7) and (3.91) is systematically corrected with equations (3.97) and (3.98).

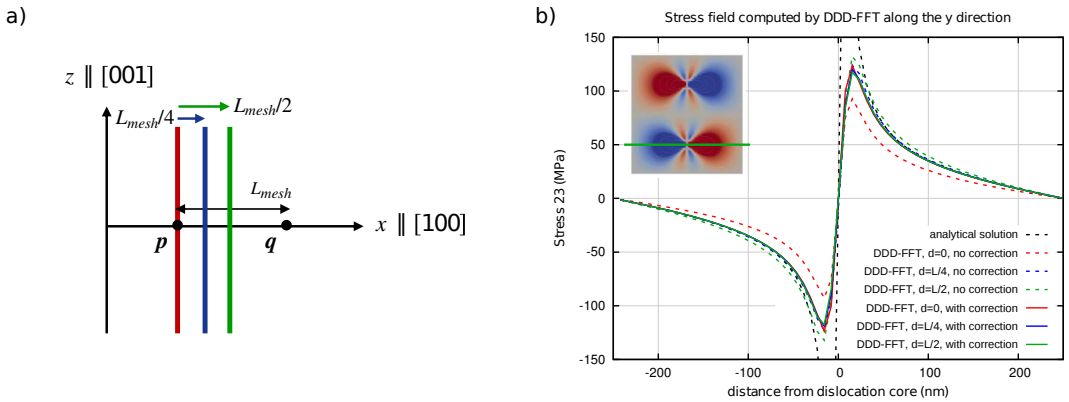


Figure 3.13: (a) Schematic of the different tested positions for the (010) prismatic dislocation loop with respect to the Fourier grid. Two consecutive grid points $\vec{\boldsymbol{p}}$ and $\vec{\boldsymbol{q}}$ are separated by the mesh size distance L_{mesh} and the red, blue and green dislocation segments introduced as depicted in figure 3.4 are positioned at distances $d = \{0, L_{mesh}/4, L_{mesh}/2\}$ from grid point $\vec{\boldsymbol{p}}$, respectively. (b) Effect of the correction on shear component σ_{23} of the stress field of a prismatic loop along the green line showed in the inset as a function of the dislocation position with respect to the Fourier grid made of $64 \times 64 \times 64$ voxels.

3.6.2 Homogeneous FFT-based implementation

With respect to the regular DDD framework developed in Chapter 2, the main changes to be implemented in the DDD-FFT method pertain to the calculation of the stress state associated with the microstructure. Thus, (1) the regularization procedure, (2) the FFT-based solver and (3) the calculation of nodal forces are the principal components that must be implemented or modified. Furthermore, apart from the development and the implementation of the analytical procedure detailed in Section 3.6.1, the numerical implementation of the homogeneous DDD-FFT approach does not involve major difficulties.

The main difficulty associated to the development of the FFT-based solver relates to the removal of the Gibbs oscillations, whose implementation is fully discussed in Appendix G.1. In contrast, the numerical implementation of the FFT-based approach is straightforward and is described in the following. First, the primary simulation volume is discretized into a regular grid of $N_{vox} = N_1 \times N_2 \times N_3$ voxels with coordinates

$\{\vec{\boldsymbol{x}}_d\}_{d=1,N_{vox}}$. Although some discrete Fourier transforms formulations allow for non-regular grids, the discretization must be here chosen such as to ensure that each voxel are cubes, i.e. that the spacing $L_{mesh} = l_i = V_i/N_i$ between the center of subsequent voxels is identical in all directions $i = \{1, 2, 3\}$, where V_i is the size of the volume in the i-th direction. Besides the FFT requirement, this condition is also required when spherical elementary volumes of radius L_{mesh} are used in the regularization procedure. Thus, when l_i are different for $i = \{1, 2, 3\}$, ellipsoids with principal axes of length l_i should be used and a corresponding overlapping correction should be developed, which is out of the scope of this work. Furthermore, to ensure full computational efficiency, the number of voxels in each direction must be chosen as powers of 2. Any choice for N_i that is not a power of 2 will induce a significant drop in the performance of the FFT algorithm. Therefore, it must be noted that the requirements on the numerical discretization of the primary volume induce constraints on its size.

Once an appropriate discretization is chosen for the simulation volume, the plastic strain produced by the glide of dislocations can be calculated via the regularization procedure described in Section 3.6.1. When initializing a simulation, the initial dislocation loops are introduced using a Volterra-like process described in Section 3.4.2.1 and illustrated in figure 3.4(b). In this process, the area swept corresponding to the entire domain enclosed by the dislocation loop is simply transferred to the mesh with the regularization procedure such that the initial plastic strain is computed. Obviously, in general, the swept area may intersect several elementary spheres, and the intersection should be computed independently for each of them. Numerically, the regularization procedure is therefore a $\mathcal{O}(N_{seg}N_{vox})$ procedure, since the intersection between the sheared area produced by each segment and the sphere associated with each voxel must be theoretically calculated. However, many of these intersections will be empty as dislocation segments are localized and their size is usually significantly smaller than the size of the simulation volume. Therefore, to avoid testing for

all possible intersections, solely the intersections between the sheared area and the spheres lying within the bounding box of the sheared area are calculated. With this, the computational complexity of the regularization is reduced to $\mathcal{O}(N_{seg})$, and it is ensured that no intersection will be missed such that the plastic strain will be entirely transferred to the mesh. To ensure good scalability of the DDD-FFT approach, this procedure is further parallelized. To achieve optimal load balance among the CPUs, dislocation swept areas are distributed across the processors based on the number of voxels comprised in their bounding boxes.

Once the regularization procedure has been performed for all dislocation segments, the plastic strain distribution $\{\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ is known at each voxel $\vec{\boldsymbol{x}}_d$. At this stage, the FFT algorithm is used to compute the discrete Fourier coefficients $\{\widehat{\boldsymbol{\epsilon}}^p(\vec{\boldsymbol{\xi}}_d)\}_{d=1,N_{vox}}$ such that the total strain $\{\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)\}_{d=1,N_{vox}}$ can be calculated in the Fourier space from equation (3.50) as:

$$\begin{aligned} &\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d) = \widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d) : \boldsymbol{C} : \widehat{\boldsymbol{\epsilon}}^p(\vec{\boldsymbol{\xi}}_d) \\ &\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d = \vec{\boldsymbol{0}}) = \boldsymbol{0} \end{aligned} \qquad \forall \text{ voxel } \vec{\boldsymbol{\xi}}_d \tag{3.99}$$

where $\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)$, $\widehat{\boldsymbol{\epsilon}}^p(\vec{\boldsymbol{\xi}}_d)$ and $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ denote the total strain, plastic strain and modified Green's function tensors in the Fourier space at voxel $\vec{\boldsymbol{\xi}}_d$. Then, the FFT algorithm is used a second time to compute the resulting total strain distribution $\{\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ in the real space (see equation (3.51)). In turns, the sequence of operations performed in one time step of the DDD-FFT approach is listed in figure 3.14.

In figure 3.14, the $\mathcal{FFT}$ and $\mathcal{FFT}^{-1}$ operators denote the discrete Fourier transforms and inverse discrete Fourier transforms that are performed using the FFT algorithm. Details on the calculation of the Fourier coefficients and on the FFT algorithm are given in Appendix F. Obviously, the performance of the DDD-FFT approach is tied up to its numerical implementation and to the choice of the FFT library. In this work, the FFTW library is used [94]. In steps iv(b) and iv(d) of figure

(i) Compute nodal forces $\vec{\boldsymbol{F}}(\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d))$

(ii) Integrate dislocation motion and determine swept areas

(iii) Regularize plastic strain $\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$

(iv) Compute new stress state $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)$ using the FFT solver

(a) $\boldsymbol{\varphi}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$

(b) $\{\widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)\} = \mathcal{FFT}\left(\{\boldsymbol{\varphi}(\vec{\boldsymbol{x}}_d)\}\right)$

(c) $\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d) = \widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)$

(d) $\{\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d)\} = \mathcal{FFT}^{-1}\left(\{\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)\}\right) + \boldsymbol{E}$

(e) $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : (\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d))$

Figure 3.14: General algorithm describing the main stages composing one time step of the homogeneous DDD-FFT approach.

3.14, the FFT computation of tensors $\boldsymbol{\varphi}$ and $\widehat{\boldsymbol{\epsilon}}$ is performed independently for each component. Thus, given the symmetry, a total of 2×6 three-dimensional FFTs need to be performed at each time step.

The computation of each FFT in steps iv)(b) and iv)(d) has a complexity of $\mathcal{O}(N_{vox} \log N_{vox})$, while that of the calculation of the total strain in iv)(c) scales with $\mathcal{O}(N_{vox})$. However, the prefactor of step iv)(c) associated with the numerical evaluation of the modified Green's operator $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ (81 components) and the double dot product $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)$ exceeds $\mathcal{O}(\log N_{vox})$ in general, so that step iv)(c) amounts to more flops than the FFTs. For isotropic elasticity, step iv)(c) can be alleviated. Recall that in the Fourier space, the expression of the modified Green's operator $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ is given from equation (3.49) as:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}_d) = \frac{1}{2}\left(\xi_l \xi_j \widehat{G}_{ik}(\vec{\boldsymbol{\xi}}_d) + \xi_l \xi_i \widehat{G}_{jk}(\vec{\boldsymbol{\xi}}_d)\right) \qquad \forall \vec{\boldsymbol{\xi}}_d \neq \vec{\boldsymbol{0}} \tag{3.100}$$

where $\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}_d) = [C_{kjil}\xi_l\xi_j]^{-1}$ is the Green's function. When using discrete gradient operators, its expression is given by equation (3.88). In the case of isotropic elasticity,

the polarization tensor in step iv)(a) is expressed as:

$$\widehat{\varphi}_{ij}(\vec{\boldsymbol{\xi}}_d) = \lambda\delta_{ij}\widehat{\epsilon}^p_{qq}(\vec{\boldsymbol{\xi}}_d) + 2\mu\widehat{\epsilon}^p_{ij}(\vec{\boldsymbol{\xi}}_d) \tag{3.101}$$

where λ and μ are the Lamé constants, and an analytical expression of the modified Green's operator is given by equation (3.42) as:

$$\widehat{\Gamma}_{ijkl}(\vec{\boldsymbol{\xi}}_d) = \frac{(\delta_{ik}\xi_l\xi_j + \delta_{il}\xi_k\xi_j + \delta_{jk}\xi_l\xi_i + \delta_{jl}\xi_k\xi_i)}{4\mu\xi^2} - \frac{(\lambda+\mu)\xi_i\xi_j\xi_k\xi_l}{\mu(\lambda+2\mu)\xi^4} \quad \text{(3.42 repeated)}$$

Therefore, the total strain in step iv)(c) can be directly obtained as:

$$\begin{aligned}\widehat{\epsilon}_{ij}(\vec{\boldsymbol{\xi}}_d) = &\left(\frac{\lambda}{\mu\xi^2}\left(1 - \frac{(\lambda+\mu)}{(\lambda+2\mu)}\right)\widehat{\epsilon}^p_{qq}(\vec{\boldsymbol{\xi}}_d) - \frac{2(\lambda+\mu)}{(\lambda+2\mu)\xi^4}\xi_k\xi_l\widehat{\epsilon}^p_{kl}\right)\xi_i\xi_j \\ &+ \frac{1}{\xi^2}\left(\xi_l\xi_j\widehat{\epsilon}^p_{il}(\vec{\boldsymbol{\xi}}_d) + \xi_l\xi_i\widehat{\epsilon}^p_{jl}(\vec{\boldsymbol{\xi}}_d)\right)\end{aligned} \tag{3.102}$$

such that the cost of the full numerical evaluation of $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ and of the product $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)$ can be reduced. However, such analytical result cannot be obtained for general anisotropy. As a result, another strategy to reduce the computational time is used.

The remarkable property of the modified Green's operator is that it solely depends on the elastic stiffness tensor $\boldsymbol{C}$ of the medium and on the voxel to which it is associated. Therefore, it can be precomputed at the beginning of each simulation for each voxel and stored in the memory, or recomputed at each time step for each voxel. Although at first sight a precomputation may appear as the most efficient approach, the benefit that can be obtained strongly depend on the architecture and hardware capacities. Thus, when running simulations on desktop computers, the cost of computing $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ at each time step is usually lower than that of storing the $\widehat{\boldsymbol{\Gamma}}(\vec{\boldsymbol{\xi}}_d)$ tensor (81 components in general) and accessing the memory. In this work, it is seen that an optimal computational efficiency is obtained when storing the Green's function $\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}_d)$ (9 components) for each voxel and recomputing the modified Green's

operator on the fly from equation (3.100).

On top of its efficiency, the FFTW library offers a parallel implementation. In the regular DDD code, the parallel implementation has been primarily devised to distribute segment-segment elastic interactions among CPUs (see Section 2.9.3). Although the computation of segment-segment elastic interactions is reduced to a minimum in the DDD-FFT approach, a fully parallel implementation is desirable, as a non-parallel FFT-solver will dramatically affect the overall performance of the code. In the parallel FFTW library, a slab decomposition technique is used, whereby the 3D primary discretized volume is decomposed into N_{cpu} layers in one of the spatial directions, where N_{cpu} is the number of CPU to be used. With this procedure, each CPU only needs to know the field values at the voxels lying in its slab. Similarly, after the FFT is computed, each CPU can solely access the values of the Fourier coefficients associated with the voxels contained in its slab. However, since the quantities in the Fourier space are independent from one another – the frequencies are only associated with the voxels – each CPU can perform step iv)(c) in figure 3.14 in a fully parallel manner. As a result, the parallel implementation of the DDD-FFT is expected to scale well, provided that the FFTW library offers a good scalability. Since the parallel FFT algorithm involves numerous point-to-point communications, the overall scalability is highly dependent on the hardware capabilities and configuration.

Regarding memory usage, a minimum of two real-valued and one complex-valued arrays of size $6N_{vox}$ are required. Following the history-dependent character of the DCM approach, one real-valued array must be dedicated to contain the plastic strain distribution $\{\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$. The other real-valued array can be used to sequentially store $\{\boldsymbol{\varphi}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$, $\{\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ and $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ values, while the complex-valued array is used to contain $\{\widehat{\boldsymbol{\varphi}}(\vec{\boldsymbol{\xi}}_d)\}_{d=1,N_{vox}}$ and $\{\widehat{\boldsymbol{\epsilon}}(\vec{\boldsymbol{\xi}}_d)\}_{d=1,N_{vox}}$ distributions in the Fourier space.

To further accelerate the DDD-FFT method and fully exploit the benefits associated with the FFT algorithm, a GPU-accelerated version of the FFT-based solver has been developed in this work, for which implementation details are provided in Appendix G.4

3.6.3 Heterogeneous FFT-based implementation

The implementation of the heterogeneous DDD-FFT approach share most of the developments introduced in the homogeneous DDD-FFT framework implementation detailed in Section 3.6.2. The main modification lies in the implementation of the *basic*, *accelerated* and *conjugate-gradient* iterative schemes in place of the direct FFT-based solver used in Section 3.6.2.

When dealing with heterogeneous elasticity, the first step consists in defining the reference medium $\boldsymbol{C}^0$. For isotropic elasticity, the elastic constants can be chosen as those reported in equations (3.75) and (3.78) for the *basic* and the *accelerated* schemes, respectively. Following the results obtained in Appendix G.2, it is seen that the convergence of the *conjugate-gradient* method is not very sensitive to the choice of $\boldsymbol{C}^0$, such that any of the previous choices can be used. In the case of anisotropic elasticity, and to the author's knowledge, no such optimal values for the reference medium have been reported in the literature, and, by default, the results obtained in equations (3.75) and (3.78) are extended to all elastic constants.

Following the iterative formulation of the *basic* scheme given by relation (3.74), the computation of the discrete stress distribution $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)$ in step iv) in figure 3.14 for the homogeneous formulation is replaced by the algorithm provided in figure 3.15. This scheme corresponds to the original implementation proposed by Moulinec and Suquet [179]. In this work, the convergence test is given by equation (3.81) and is performed on the value of the total strain $\boldsymbol{\epsilon}_{i+1}$ such that step A)(a) is optional and step B)(e) only needs to be performed when the convergence criterion is satisfied.

(iv) Compute new stress state $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)$ using the FFT solver

(A) Initialization

(a) $\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{E}$

(b) $\boldsymbol{\sigma}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{C}(\vec{\boldsymbol{x}}_d) : (\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}_0^p(\vec{\boldsymbol{x}}_d))$ (optional)

(B) Iteration i

(a) $\boldsymbol{\tau}_i(\vec{\boldsymbol{x}}_d) = \boldsymbol{\delta C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}_d) - \boldsymbol{C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$

(b) $\{\widehat{\boldsymbol{\tau}}_i(\vec{\boldsymbol{\xi}}_d)\} = \mathcal{FFT}\left(\{\boldsymbol{\tau}_i(\vec{\boldsymbol{x}}_d)\}\right)$

(c) $\widehat{\boldsymbol{\epsilon}}_{i+1}(\vec{\boldsymbol{\xi}}_d) = -\widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{\tau}}_i(\vec{\boldsymbol{\xi}}_d)$

(d) $\{\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d)\} = \mathcal{FFT}^{-1}\left(\{\widehat{\boldsymbol{\epsilon}}_{i+1}(\vec{\boldsymbol{\xi}}_d)\}\right) + \boldsymbol{E}$

(e) $\boldsymbol{\sigma}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : (\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d))$

(f) Convergence test

Figure 3.15: Algorithm to iteratively solve for the stress distribution $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}$ when using the *basic* scheme (3.74) in the heterogeneous DDD-FFT approach. Indices 0, i and $i+1$ refer to the values of the field quantities at iterations 0, i and $i+1$, respectively.

Nevertheless, such criterion implies the usage of a supplementary real-valued array of size $6N_{vox}$ with respect to the homogeneous implementation to store the values of the total strain field at the previous iteration. Note that other convergence criteria can be used such as to test for the departure from the mechanical equilibrium [177].

The algorithm for the *accelerated* scheme is provided in figure 3.16. In contrast with the *basic* scheme, the *accelerated* algorithm involves a supplementary step whose cost is generally largely compensated by the increase in convergence rate that it provides (see Appendix G.2). Furthermore, quantity $2\left(\boldsymbol{C}(\vec{\boldsymbol{x}}_d) + \boldsymbol{C}^0\right)^{-1} : \boldsymbol{C}^0$ can be precomputed for each phase so as to avoid inverting a fourth-order order tensor at each iteration for each voxel. Besides, by appropriately using the arrays containing quantities $\{\boldsymbol{\epsilon}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ and $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$, no supplementary memory usage is required to store the intermediate quantity $\{\boldsymbol{e}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$.

The implementation of the *conjugate-gradient* (CG) method is slightly different

(iv) Compute new stress state $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)$ using the FFT solver

(A) Initialization

(a) $\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{E}$

(b) $\boldsymbol{\sigma}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{C}(\vec{\boldsymbol{x}}_d) : (\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}_0^p(\vec{\boldsymbol{x}}_d))$ (optional)

(B) Iteration i

(a) $\boldsymbol{\tau}_i(\vec{\boldsymbol{x}}_d) = \boldsymbol{\delta C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}_d) - \boldsymbol{C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$

(b) $\{\widehat{\boldsymbol{\tau}}_i(\vec{\boldsymbol{\xi}}_d)\} = \mathcal{FFT}\left(\{\boldsymbol{\tau}_i(\vec{\boldsymbol{x}}_d)\}\right)$

(c) $\widehat{\boldsymbol{e}}_{i+1}(\vec{\boldsymbol{\xi}}_d) = -\widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{\tau}}_i(\vec{\boldsymbol{\xi}}_d)$

(d) $\{\boldsymbol{e}_{i+1}(\vec{\boldsymbol{x}}_d)\} = \mathcal{FFT}^{-1}\left(\{\widehat{\boldsymbol{e}}_{i+1}(\vec{\boldsymbol{\xi}}_d)\}\right) + \boldsymbol{E}$

(e) $\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}_d) + 2\left(\boldsymbol{C}(\vec{\boldsymbol{x}}_d) + \boldsymbol{C}^0\right)^{-1} : \boldsymbol{C}^0 : (\boldsymbol{e}_{i+1}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}_d))$

(f) $\boldsymbol{\sigma}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : (\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d))$

(g) Convergence test

Figure 3.16: Algorithm to iteratively solve for the stress distribution $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}$ when using the *accelerated* scheme (3.77) in the heterogeneous DDD-FFT approach. Indices 0, i and $i+1$ refer to the values of the field quantities at iteration 0, i and $i+1$, respectively.

from both the previous schemes. Essentially, it consists in implementing the classical CG algorithm to solve for system $Ax = b$ to the functional system of equations $\boldsymbol{A}(\boldsymbol{X}_d) = \boldsymbol{B}$ described in equation (3.79), for which an expression of form $\boldsymbol{A}(\boldsymbol{X}_d)$ is given in equation (3.80). The algorithm developed in this work is presented in figure 3.17. In contrast with the previous schemes, the CG algorithm requires the calculation of a convolution in the initialization stage to evaluate the constant right-hand side term $\boldsymbol{B}(\vec{\boldsymbol{x}}_d)$ containing the plastic strain distribution $\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$. To avoid the calculation of $\boldsymbol{A}(\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d))$ involving a second convolution, the initial value of the total strain $\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d)$ is set to $\boldsymbol{0}$ at every voxel $\vec{\boldsymbol{x}}_d$. As the CG method has been observed to be insensitive to the choice of the initial guess in this work, this choice is expected to have no impact on the convergence rate. Then, the classical algorithm of the CG method is applied, in which quantity $\boldsymbol{A}(\boldsymbol{P}_i(\vec{\boldsymbol{x}}_d))$ involving a convolution with the modified Green's operator (see equation (3.80)) is calculated in the Fourier space using FFTs.

(iv) Compute new stress state $\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)$ using the FFT solver

(A) Initialization

(a) $\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{0}$

(b) Compute $\boldsymbol{B}(\vec{\boldsymbol{x}}_d) = \boldsymbol{\Gamma}^0 * [\boldsymbol{C} : \boldsymbol{\epsilon}^p](\vec{\boldsymbol{x}}_d) + \boldsymbol{E}$

(1) $\boldsymbol{B}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)$

(2) $\{\widehat{\boldsymbol{B}}(\vec{\boldsymbol{\xi}}_d)\} = \mathcal{FFT}(\{\boldsymbol{B}(\vec{\boldsymbol{x}}_d)\})$

(3) $\widehat{\boldsymbol{B}}(\vec{\boldsymbol{\xi}}_d) = \widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{B}}(\vec{\boldsymbol{\xi}}_d)$

(4) $\{\boldsymbol{B}(\vec{\boldsymbol{x}}_d)\} = \mathcal{FFT}^{-1}\left(\{\widehat{\boldsymbol{B}}(\vec{\boldsymbol{\xi}}_d)\}\right) + \boldsymbol{E}$

(c) $\boldsymbol{R}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{B}(\vec{\boldsymbol{x}}_d) - \boldsymbol{A}(\boldsymbol{\epsilon}_0(\vec{\boldsymbol{x}}_d)) = \boldsymbol{B}(\vec{\boldsymbol{x}}_d)$

(d) $\boldsymbol{P}_0(\vec{\boldsymbol{x}}_d) = \boldsymbol{R}_0(\vec{\boldsymbol{x}}_d)$

(e) $\boldsymbol{\sigma}_0(\vec{\boldsymbol{x}}_d) = -\boldsymbol{C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{\epsilon}_0^p(\vec{\boldsymbol{x}}_d)$ (optional)

(B) Iteration i

(a) Compute $\boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d) = \boldsymbol{A}(\boldsymbol{P}_i(\vec{\boldsymbol{x}}_d)) = \boldsymbol{P}_i(\vec{\boldsymbol{x}}_d) + \boldsymbol{\Gamma}^0 * [\boldsymbol{\delta C} : \boldsymbol{P}_i](\vec{\boldsymbol{x}}_d)$

(1) $\boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d) = \boldsymbol{\delta C}(\vec{\boldsymbol{x}}_d) : \boldsymbol{P}_i(\vec{\boldsymbol{x}}_d)$

(2) $\{\widehat{\boldsymbol{Q}}_i(\vec{\boldsymbol{\xi}}_d)\} = \mathcal{FFT}(\{\boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d)\})$

(3) $\widehat{\boldsymbol{Q}}_i(\vec{\boldsymbol{\xi}}_d) = \widehat{\boldsymbol{\Gamma}}^0(\vec{\boldsymbol{\xi}}_d) : \widehat{\boldsymbol{Q}}_i(\vec{\boldsymbol{\xi}}_d)$

(4) $\{\boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d)\} = \mathcal{FFT}^{-1}\left(\{\widehat{\boldsymbol{Q}}_i(\vec{\boldsymbol{\xi}}_d)\}\right) + \{\boldsymbol{P}_i(\vec{\boldsymbol{x}}_d)\}$

(b) $\alpha_i = \frac{\boldsymbol{R}_i^T(\vec{\boldsymbol{x}}_d)\boldsymbol{R}_i(\vec{\boldsymbol{x}}_d)}{\boldsymbol{P}_i^T(\vec{\boldsymbol{x}}_d)\boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d)}$

(c) $\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{\epsilon}_i(\vec{\boldsymbol{x}}_d) + \alpha_i \boldsymbol{P}_i(\vec{\boldsymbol{x}}_d)$

(d) $\boldsymbol{\sigma}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{C} : (\boldsymbol{\epsilon}_{i+1}(\vec{\boldsymbol{x}}_d) - \boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d))$

(e) Convergence test

(f) $\boldsymbol{R}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{R}_i(\vec{\boldsymbol{x}}_d) - \alpha_i \boldsymbol{Q}_i(\vec{\boldsymbol{x}}_d)$

(g) $\beta_i = \frac{\boldsymbol{R}_{i+1}^T(\vec{\boldsymbol{x}}_d)\boldsymbol{R}_{i+1}(\vec{\boldsymbol{x}}_d)}{\boldsymbol{R}_i^T(\vec{\boldsymbol{x}}_d)\boldsymbol{R}_i(\vec{\boldsymbol{x}}_d)}$

(h) $\boldsymbol{P}_{i+1}(\vec{\boldsymbol{x}}_d) = \boldsymbol{R}_{i+1}(\vec{\boldsymbol{x}}_d) + \beta_i \boldsymbol{P}_i(\vec{\boldsymbol{x}}_d)$

Figure 3.17: Algorithm to iteratively solve for the stress distribution $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}$ when using the *conjugate-gradient* (*CG*) method (3.79) in the heterogeneous DDD-FFT approach. Indices 0, i and $i+1$ refer to the values of the field quantities at iteration 0, i and $i+1$, respectively.

As attested by the algorithms provided in figures 3.15, 3.16, and 3.17, the *conjugate-gradient* scheme requires more calculations per iteration than the *basic* and *accelerated* schemes. However, these calculations are not very demanding as they only involve

dot-products and vector additions. Furthermore, the gain in the number of iterations to reach convergence that it allows largely balances the slightly higher cost associated with individual iterations. Nonetheless, the main drawback of the *CG* method lies in its memory requirements. Thus, compared to the *basic* and *accelerated* schemes, three supplementary real-valued array of size $6N_{vox}$ are required to store $\{\boldsymbol{P}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$, $\{\boldsymbol{R}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ and $\{\boldsymbol{Q}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ quantities.

Details on the GPU-accelerated implementation of the DDD-FFT heterogeneous schemes are provided in Appendix G.4

3.7 Performance of the DDD-FFT approach

In this section, the computational performance of the DDD-FFT approach is compared with that of the regular DDD simulation developed in Chapter 2. First, to overcome the complications associated with the comparison between parallel implementations, the comparison between both approaches is made on the basis of a single CPU utilization for homogeneous simulations. Then, the performance of the homogeneous and heterogeneous GPU implementations is assessed.

3.7.1 Homogeneous serial implementation

The overall efficiency of the DDD-FFT code primarily results from the use of the very efficient FFT algorithm to compute the mechanical fields (see Appendix F). For a DDD simulation volume discretized according to a regular Fourier grid composed of N voxels in each spatial direction, i.e. for $N_{tot} = N^3$ total number of grid points, the complexity associated with the computation of the discrete Fourier transform and its inverse scales with $\mathcal{O}(N_{tot} \log N_{tot})$ while the computation of the strain field in the Fourier space as defined in equation (3.50) is a $O(N_{tot})$ process. Comparatively, a complexity of $O(N_{tot}^2)$ can be achieved in the most favorable case with the standard 3D FEM (assuming simple linear elements), where N_{tot} is the total number of elements, i.e. for comparable resolutions. Obviously, for large values of N_{tot},

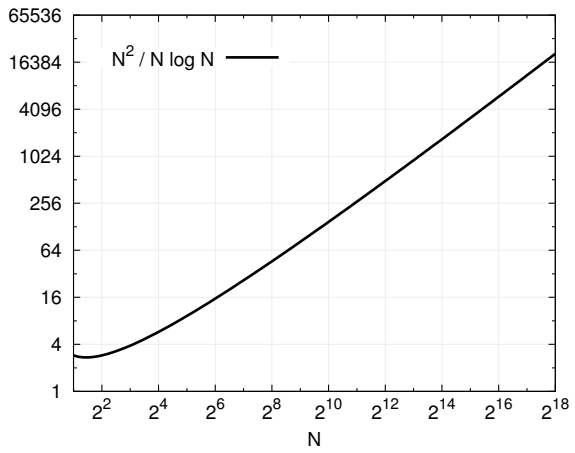


Figure 3.18: Evolution of the ratio $N^2/(N \log N)$ as a function of N in logarithmic scales. Assuming both complexity prefactors are of the same order of magnitude, this ratio provides an insight on the difference of the computational cost between FFT-based and FEM solvers as a function of the mesh resolution.

the difference between $N_{tot} \log N_{tot}$ and N_{tot}^2 becomes considerable, hence the gain in the computation time up to several orders of magnitudes generally attributed to FFT-based approaches.

Although a direct comparison between the recent implementation of the DCM-FEM model proposed in [234] and the DDD-FFT approach cannot be made – the DCM-FEM code is currently not available to the author –, insights on the computational gain offered by the FFT-based solver with respect to the FEM framework are provided in figure 3.18. In this figure, the ratio $N^2/(N \log N)$ is plotted as a function of N. Considering that the pre-factors are roughly identical in both methods, this ratio provides a basis of comparison for the gain in computational time that can be obtained with FFT-based over FEM solvers. For instance, for a resolution of $N_{tot} = 16^3$, the gain amounts to a factor 500 and for $N_{tot} = 32^3$ it exceeds 3000. For a resolution of $N_{tot} = 2^{18} = 64^3 = 128^2$ pixels/voxels generally used in DDD-FFT, an equivalent FEM calculation would require about 21000 times the number of flops.

In parallel, by coupling the FFT-solver with the DCM procedure, the DDD-FFT approach benefits from the significant reduction in the number of dislocation-dislocation elastic interaction calculations compared to most of the current DDD approaches. Thus, in the case where sufficiently fine grids are used, the DDD complexity is significantly less dependent on the number of dislocation segments N_{seg} present in the volume, but more on the number of grid points N_{tot}, i.e on the resolution. The numerical complexity associated with the DCM procedure alone is a $\mathcal{O}(N_{seg})$ process requiring considerably less numerical operations than computing dislocation-dislocation interactions (even with an analytical formulation), since it solely consists in analytically determining the intersection area between the dislocations sheared area and the elementary spheres of their neighboring grid points, as detailed in Section 3.6.1. In the current implementation, the algorithm to compute the intersection areas is able to process a million intersections in 0.2 seconds. Furthermore, as mentioned earlier, the computation of supplementary local contributions requires the determination of pairs of interacting portions of dislocation segments whose distance are closer than $h/2 = L_{mesh}$. The analytical method presented in Section 3.4.1.2 to determine such portions of segments is implemented so as to process a million segment pairs in 0.5 seconds.

Figure 3.19 shows a comparison of the execution run times per time step as a function of the number of segments N_{seg} and of the DDD approach used for a periodic DDD simulation on an elastically isotropic material. Here, the regular DDD refers to the version of the DDD code developed in Chapter 2 in which the nodal forces on neighbor dislocation segments are analytically computed from the non-singular formulation developed in [35], while the Box Method technique is employed to compute far-field elastic interactions [240] (see Section 2.9.2.1). The computation of the far-field elastic interactions via the use of the Box Method is performed every 10 simulation steps. The averaged execution times per simulation step obtained using the

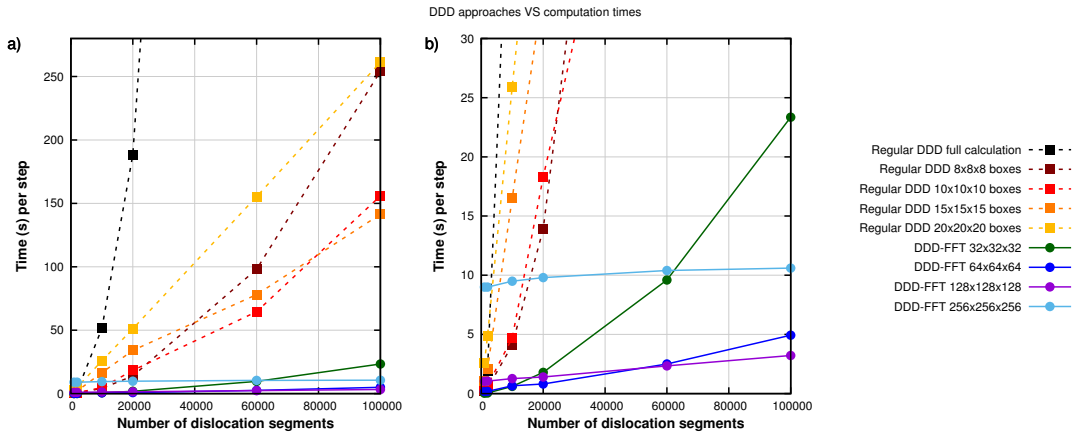


Figure 3.19: (a) Comparison of the computational cost between regular DDD simulations and the DDD-FFT approach as a function of the number of segments in the simulation volume for elastically isotropic materials. The times, averaged over several simulation steps, are given for the utilization of a single CPU and measured according to the current implementation of the DDD code. The times of the regular DDD simulations are given based on the full $\mathcal{O}(N_{seg}^2)$ calculation and on the utilization of the box method for different number of boxes. The times for the DDD-FFT approach are given for different grid sizes. (b) Close-up on the computation times obtained with the DDD-FFT approach for different grid sizes.

regular DDD approach with different number of boxes N_{box} are compared to those obtained with the DDD-FFT approach for different grid sizes N_{vox}. For the sake of comparison, the run time corresponding to a full $\mathcal{O}(N_{seg}^2)$ calculation is also reported. For a fair comparison, i.e. not accounting for the complications associated with parallel computation, the execution times have been measured in running the code on a single CPU.

Examination of figure 3.19 clearly shows that the DDD-FFT becomes very efficient when large numbers of segments are to be treated, hence highlighting the potential of the approach. Note as well that in order to obtain the same precision with the regular DDD approach than with the DDD-FFT approach, the Box Method would have to be performed at every simulation step. In practice, as discussed in Section 2.9.2.1, the calculation of far-field interactions with the Box Method is generally performed at a specified frequency f_{box} in order to reduce the computation time, taking

advantage of the fact that the $1/R$ elastic field of far dislocations is not expected to drastically change from one time step to another. In the computation times reported in figure 3.19, $f_{box} = 1/10$, i.e. the Box Method is performed every 10 steps. If it were to be performed at every step, i.e. working with the same precision as with the DDD-FFT approach, the computation times of the regular DDD method reported in figure 3.19 would roughly increase by a factor 10. A close-up on the DDD-FFT computation times as a function of the grid size is given in figure 3.19(b). Interestingly, the computation time associated to the grid made of $32 \times 32 \times 32$ voxels becomes significantly more important than that of finer meshes as the number of segments N_{seg} increase. This is because the number of local interactions becomes all the more prevalent than the mesh is coarse when N_{seg} increases. For the same reason, the computation cost for a grid of $64 \times 64 \times 64$ voxels becomes more expensive than that for $128 \times 128 \times 128$ voxels for $N_{seg} > 50000$ in the current implementation. Therefore, for optimal performances, the size of the grid should be chosen as a function of the maximum number of segments to be treated, if such quantity is known a priori. Alternatively, considering the multiplication of dislocation segments during DDD simulations, a grid refinement technique may be implemented to dynamically select the optimal resolution [81]. However, implementing such technique would require the development of grid interpolation operators so as to accurately transfer the accumulation of plastic strain between grids of different sizes. For $N_{seg} = 100000$, the computation times for grids of $64 \times 64 \times 64$ and $128 \times 128 \times 128$ voxels remain under 5 seconds per simulation step, thereby allowing for a gain in time of a factor up to 30 compared to the regular DDD method when using the Box Method. Finally, note that for a grid of $256 \times 256 \times 256$ voxels, the computation time ranges between 9 and 11 seconds according to the number of segments. For that resolution, the FFT solver alone (including the computation of both Fourier transforms and that of the strain in the Fourier space) requires about 8 seconds. While the actual computation

of local interactions is insignificant for such mesh size, the sole search for the portions of segments closer than $h/2$ amounts for the rest of the time.

As underlined in Section 2.9.2.1, the more efficient Fast Multipole Method (FMM) has progressively replaced the Box Method approximation for the computation of long-range interactions in DDD simulations [148, 5, 270, 263]. Since this technique has not been implemented in the regular DDD code in this thesis, a direct comparison is difficult. However, through implementation of their new version of the FMM, Zhao and co-workers reported a gain of a factor 21 in the isotropic stress calculation when using the FMM compared to the full calculation for $N_{seg} = 25000$ [270]. In the present work, the gain in the total step time compared to the full calculation amounts to 230 when using the DDD-FFT with $64 \times 64 \times 64$ voxels and to 140 for $128 \times 128 \times 128$ voxels for $N_{seg} = 20000$.

Further, the results reported in figure 3.19 are given for simulations using homogeneous isotropic elasticity. In the case of anisotropic elasticity, the computation gain is expected to amount to a supplementary order of magnitude, since the overall cost is expected to be comparable to that of isotropic elasticity for fine meshes. In contrast, anisotropic calculations are rarely used in regular DDD simulations due to their prohibitive cost. For instance, in previous works [202, 40, 264], the relative cost of anisotropic calculations was reported to amount between 200 and 500 times that of isotropic calculations when numerically integrating the anisotropic Green's function given (2.23). However, the recent development of approximations based on expansions in spherical harmonics have permitted significant gains in computational time, but their efficiency remains largely conditioned by the amount of anisotropy and the desired level of accuracy, and their relative cost still remains above one order of magnitude [12, 13]. In the case of heterogeneous elasticity, a direct comparison with the DDD-FFT method and current approaches is more difficult. For bicrystal materials, line integral solutions for the stress field of dislocation segments have been developed

[107, 2, 48] and can be used in replacement of the analytical formulation in the regular DDD framework. However, the cost of their numerical integrations is expected to be similar to that of line integral anisotropic formulations, and is therefore prohibitive. In the case of more complex heterogeneous geometries, current techniques solely include the DCM-FEM approach, whose relative cost with the DDD-FFT approach can be estimated with figure 3.18.

Finally, note that the times reported here are strongly dependent on the current numerical implementation and optimizations of the DDD code, as well as on the software and hardware configurations, and are therefore meant to provide a basis for comparison. As such, the results reported in figure 3.19 were extracted using a single Intel Xeon CPU at 3.40 GHz and the Intel Fortran *mpiifort* compiler.

3.7.2 GPU-accelerated version

In this section, the performance and acceleration provided by the GPU-FFT-solver developed in Appendix G.4 are assessed for the elastically homogeneous and heterogeneous DDD-FFT formulations.

3.7.2.1 Homogeneous FFT-solver

As discussed in Appendix G.4, the GPU-accelerated version of the present DDD-FFT code is, by lack of time, limited to the development of a GPU-accelerated FFT-based solver, while the other components of the DDD cycle remain performed on the CPU(s). Therefore, to first assess the overall acceleration that would be allowed by the GPU implementation, the relative CPU times associated with three most intensive stages of the DDD cycle – namely the FFT-based solver, the regularization procedure and the forces calculation – are plotted in figure 3.20(a) for different numbers of dislocation segments and different resolutions. In this graph, the relative times correspond, for each of the three stages, to the ratio of the execution time of each stage to the total execution time of a single DDD-FFT step executed on a single

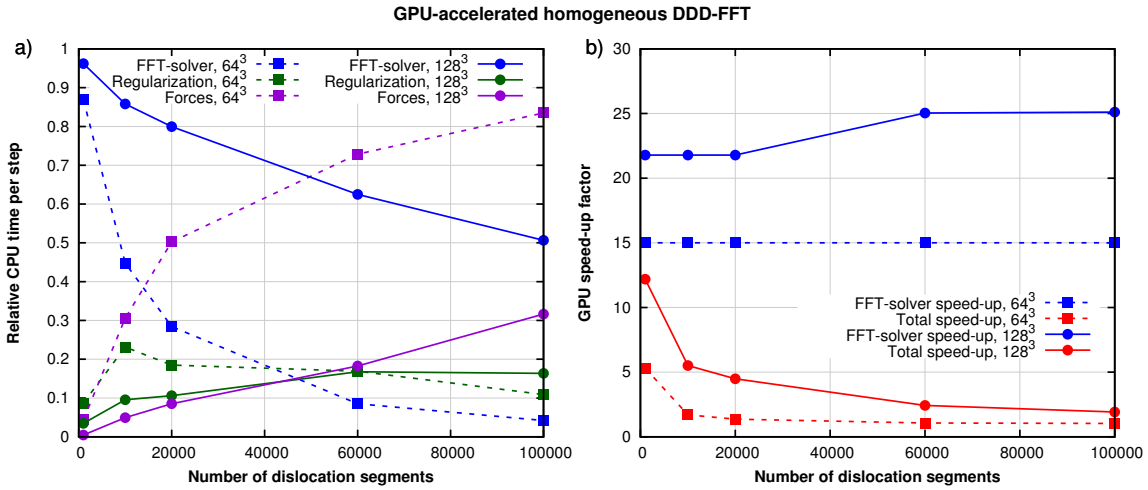


Figure 3.20: (a) Relative CPU execution times of the three main stages of the DDD-FFT cycle, namely the FFT-solver, the regularization procedure and the force calculations, as a function of the number of dislocation segments and for different resolutions. For each stage, the relative time is calculated as the ratio of the execution time of the stage to the total time per simulation step when executed on a single CPU. (b) Speed-up factors obtained when running the host DDD-FFT program on a single CPU and using the GPU-FFT-solver on a GeForce GTS 450 device. The blue lines correspond to the speed-up factor associated with the FFT-solver alone and the red lines to the speed-up achieved for the total simulation step.

CPU.

These three main stages, forces calculation, regularization procedure and FFT-based solver, correspond to stages (i), (iii) and (iv) in the general homogeneous DDD-FFT algorithm presented in figure 3.14, respectively. When summing the relative times associated with these three main stages, the value of 1 is nearly reached, attesting that these three stages together account for most of the computational work per simulation step. Interestingly, it is observed that when the number of segments is small ($N_{seg} < 10000$), the FFT-solver accounts for most of the computational time per simulation step. Conversely, as the number of segments increases, the part of the FFT-solver to the total computational time diminishes, especially due to the increase in the computational cost of force calculations – here force calculations involve the interpolation of the stress field from the FFT grid and the computations of the supplementary local contributions, see equation (3.54) –. This is particularly apparent

for the resolution of 64^3 voxels for which the computation time associated with the FFT-solver drops to 5% of the total time, while that of the force calculations rises to nearly 85% for $N_{seg} = 100000$. For a resolution of 128^3 voxels, the contrast is less pronounced as the FFT-solver accounts for half of the execution time while the force calculations require approximately 30% of the total time per step. As a result, and since the GPU implementation solely pertains to the FFT-solver, the acceleration allowed by the GPU-accelerated version is expected to be maximal for high resolutions and small numbers of segments.

The speed-up factors obtained with the GPU-accelerated implementation are reported in figure 3.20(b). As briefly discussed in Appendix G.4, these results and the associated performance are intrinsically linked to the hardware configuration that is used. Here, the homogeneous DDD-FFT code is primarily executed on a single Intel Xeon E31270 CPU at 3.40 GHz using the GNU Fortran 90 *gfortran* compiler, while the GPU-FFT-solver is executed on a entry-level GeForce GTS 450 GPU device. This GPU device features 192 CUDA cores, a graphics clock at 783 MHz, a processor clock at 1566 MHz, and a memory bandwidth of 57.7 GB/s. In figure 3.20(b), two speed-up factors are reported. The first, denoted S_{fft} and plotted with blue lines, corresponds to the speed-up associated with the FFT-solver only, i.e. the ratio of the execution times when the FFT-solver is executed on the CPU to that when it is performed on the GPU, and is given as:

$$S_{fft} = \frac{T_{fft}^{CPU}}{T_{fft}^{GPU}} \tag{3.103}$$

where T_{fft}^{CPU} denotes the execution time of the FFT-solver on the CPU and T_{fft}^{GPU} that when executed on the GPU. From figure 3.20(b), $S_{fft} = 15$ for a resolution of 64^3 voxels, and ranges from 22 to 25 when using 128^3 voxels. The quasi-constant value of the speed-up factor S_{fft} as a function of the number of segments highlights the independence of the FFT-solver cost to the microstructure complexity, as attested

for the resolution of 64^3 voxels. In the case $N_{vox} = 128^3$, the increase in the speed-up factor with the number of segments is linked to the increase in ram page time due to higher memory usage when executed on the CPU. In the current implementation and using a GeForce GTS 450, the homogeneous GPU-FFT-solver requires approximately 0.0366 second per step for 64^3 voxels and 0.2751 second for 128^3 voxels. Note that for the execution with 64^3 voxels, 0.0157 second is dedicated to memory copy, while it amounts to 0.1175 second for 128^3 voxels. Therefore, in both cases, 43% of the homogeneous GPU-FFT-solver execution time is dedicated to memory transfers. Therefore, both better hardware capabilities in terms of number of CUDA cores and memory bandwidth will induce higher values of the S_{fft} speed-up factor reported in this work.

The second speed-up factor S_{tot} reported with red lines in figure 3.20(b) corresponds to the overall acceleration allowed by the GPU-accelerated FFT-solver for the DDD-FFT homogeneous formulation. If T_{tot}^{CPU} and T_{tot}^{GPU} denote the total execution time per step when performing the FFT-solver computation on the CPU or on the GPU, respectively, S_{tot} is expressed as the ratio:

$$S_{tot} = \frac{T_{tot}^{CPU}}{T_{tot}^{GPU}} \equiv \frac{T_{fft}^{CPU} + T_{rem}}{T_{fft}^{GPU} + T_{rem}} \tag{3.104}$$

where T_{rem} is the remaining time per step executed on the CPU and associated with every other stages than the FFT-solver (e.g. including the regularization procedure, force calculations, etc...). It is seen that the total acceleration S_{tot} decreases with increasing number of segments, as the FFT-solver accounts for less relative CPU time. Thus, the overall speed-up factor of 5 for $N_{vox} = 64^3$ when $N_{seg} = 1000$ segments drops to 1.03 when $N_{seg} = 100000$. Similarly for a resolution of $N_{vox} = 128^3$, it falls from 12 for $N_{seg} = 1000$ to 1.91 for $N_{seg} = 100000$. However, these results were to be expected. From equation (3.104), the maximum theoretical overall speed-up S_{tot}^{max} that can be achieved is estimated as:

$$S_{tot}^{max} = \frac{T_{fft}^{CPU} + T_{rem}}{T_{rem}} \tag{3.105}$$

As a result, for $N_{vox} = 64^3$ and $N_{seg} = 100000$, for which the relative CPU time of the FFT-solver accounts for 5% (see figure 3.20(a)), i.e. T_{rem} accounts for 95%, the maximum speed-up that can be achieved is $S_{tot}^{max} = 1.05$. For a resolution of 128^3 voxels, the FFT-solver accounts for 50% of the CPU time, such that a maximum speed-up of $S_{tot}^{max} = 2$ can be achieved. These nearly correspond to the factors reported in figure 3.20(b). Consequently, for large numbers of segments, a GPU device with better performance would not provide any further acceleration, and supplementary gains necessarily involve running the other stages of the code in parallel or developing a GPU-implementation of the remaining components. For local force calculations, the GPU accelerated DDD approach recently developed in [83] can be adopted.

3.7.2.2 Heterogeneous FFT-solver

As detailed in Section 3.5.3 and 3.6.3, the heterogeneous DDD-FFT formulation involves an iterative loop over the FFT calculations (e.g. see figures 3.15 and 3.16). As a result, the heterogeneous GPU-FFT-solver is expected to yield larger speed-up factors than for the homogeneous formulation since (1) the relative CPU execution times of the FFT-solver accounts for a larger fraction of the total execution time, and (2) the constant cost of the memory transfers to the GPU accounts for a smaller fraction of the total heterogeneous GPU-FFT-solver execution time.

The speed-up factor S_{fft} defined in equation (3.103) associated with the heterogeneous FFT-solver alone is reported for different numbers of FFT iterations in figure 3.21(a). Thus, when using a GeForce GTS 450 device, the acceleration allowed by the GPU implementation increases towards an asymptotic value for approximately

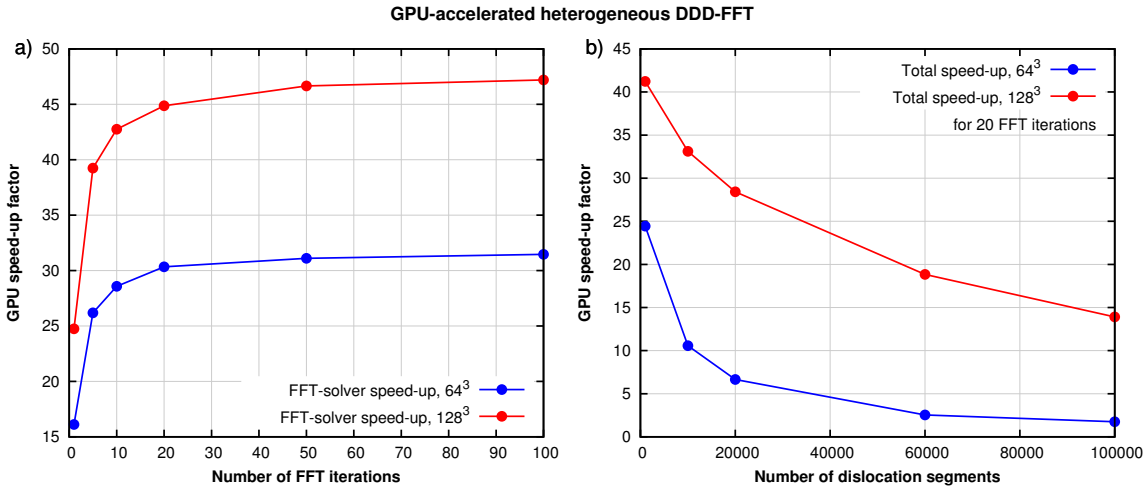


Figure 3.21: (a) GPU speed-up factor associated with the heterogeneous FFT-solver alone as a function of the number of FFT iterations to reach convergence. (b) Overall speed-up that can be achieved with the heterogeneous DDD-FFT code in the current configuration as a function of the number of dislocation segments when 20 iterations are needed to reach convergence.

$N_{iter} = 100$ iterations. As pointed out earlier, this increase can be principally attributed to the decreasing fraction of the GPU execution time associated with memory copy as the number of iteration increases. As such, the overall memory transfers drop from 43% for $N_{iter} = 1$ (i.e. homogeneous formation) to 1% of the total FFT-solver time for $N_{iter} = 100$, thereby yielding the S_{fft} speed-up factor to increase by nearly a factor 2 between $N_{iter} = 1$ and $N_{iter} = 100$.

Consequently, the overall speed-up factor S_{tot} defined in equation (3.104) is expected to exceed that of the homogeneous formulation in all cases for heterogeneous elasticity. Thus, as reported in figure 3.21(b) for $N_{iter} = 20$, the overall acceleration provided by the heterogeneous GPU-FFT-solver is more important than that reported in figure 3.20(b), independently of the resolution and the number of dislocation segments. Specifically, for small numbers of segments ($N_{seg} < 10000$), the overall speed-up factor S_{tot} is very similar to the FFT-solver speed-up S_{fft} alone. This is because in this case, the heterogeneous FFT-solver accounts for nearly 100% of the total CPU time per step. Therefore, in this case, better performance of the GPU

device is expected to translate into a direct scaling in the overall acceleration that can be achieved. Nonetheless, as in the homogeneous case – although at a slower rate –, the overall acceleration drops when the number of segments becomes large. In this case, other components of the code would benefit from (1) being run in parallel on several CPUs or (2) receiving a GPU-implementation.

3.8 Conclusion

In this chapter, a new full-field approach to compute mechanical fields in periodic DDD simulations is presented. The main idea of the DDD-FFT approach relies on the computation of strains and stresses induced by the presence of dislocation segments in a periodic simulation volume directly in the Fourier space, while the transformation between the real Cartesian space and the Fourier space is performed thanks to the FFT algorithm. To this end, the DDD-FFT approach is coupled with the DCM model originally developed by Lemarchand and co-workers [147]. As a result, the stress state calculated in the Fourier grid inherently incorporates long-range elastic interactions with no distinction between isotropic and anisotropic elastic media. In addition, to accurately evaluate the stress state associated with the microstructure, the DCM approach requires the computation of supplementary local dislocation-dislocation interactions than are not accounted for by the grid. The local contributions have to be added for portions of segment pairs closer than the mesh size $L_{mesh} = h/2$ such that the number of interactions inversely scales with the resolution. Consequently, in the case where the Fourier grid is chosen to be sufficiently fine such that the computation of local contributions pertaining to neighbor segments seldom occurs, the gain in computation time with respect to the regular DDD approach is significant when a large number of segments are to be treated, and the computational cost become insensitive to the shape of the elasticity tensor.

From a general perspective, given that the mean spacing of dislocations typically

scales as $\lambda = 1/\sqrt{\rho}$ where ρ is the dislocation density, and assuming that all dislocation segments are equally spaced within the volume, the number of local contributions to compute is not expected to be significant up to dislocation densities of $\rho = 1/L^2$. For instance, for a simulation volume of $V_s = (5\mu m)^3$ discretized into $64 \times 64 \times 64$ voxels, the DDD-FFT approach is theoretically expected to perform well up to densities of $\rho = 10^{14} m^{-2}$. Thus, simulations on realistic microstructure can be performed on desktop computers, without requiring a large number of CPUs. Conversely, when the Fourier grid is coarse, the DDD-FFT approach becomes analogous to a DDD simulation in which elastic far fields are computed via the FMM, and in which stress fields are inherently computed from infinite replications of the primary volume. In addition, it is interesting to notice that FFT-based approaches are easily parallelizable, especially thanks to efficient parallel 3D FFT implementations that have been developed, and are particularly suitable for GPU-accelerated computing, for which a dedicated implementation have been developed, thereby enabling further acceleration.

Furthermore, the ability of DDD-FFT approach to handle elastic heterogeneities offers broad new possibilities to DDD simulations. Practically, as originally done in FFT-based approaches for studying composite materials [178, 179] and polycrystalline aggregates [143, 144], the primary DDD simulation volume can easily be partitioned into grains with different elastic properties. Note that the FEM-based implementation of the DCM approach proposed in the work of Vattré et al. [234] already provides such possibility. However, the application to realistic simulations is not comparable to the DDD-FFT approach in terms of achievable level of strains and accuracy. Therefore, the method presented in this work paves the way towards achieving practical scale transition between DDD scale plasticity and crystal plasticity. Especially, DDD can be regarded as a tool to test and validate constitutive laws. Furthermore, the present approach establishes a direct connection between DDD simulations and Field Dislocation Mechanics models for which spectral methods were recently introduced for static

cases [26, 20]. In that sense, the DDD-FFT can be regarded as a three-dimensional dynamic extension of these works.

Note however that the principal limitation of the present formulation lies in its prerequisite to deal with fully periodic simulations. Nonetheless, an heterogeneous simulation including an outer layer composed of a porous phase may be used to simulate the presence of free-surfaces. Alternatively, the recent development of spectral methods to account for the effects of free-surfaces [252] may be adapted to the DDD-FFT approach as a way to overcome the limitation associated with the periodicity. Implementing these is beyond the scope of the present work.

CHAPTER IV

DISLOCATION-DISLOCATION INTERACTIONS

4.1 Motivation

The strain hardening behavior in single crystals deformed by slip is primarily governed by the interactions between dislocations. For instance, these can take the form of glissile junctions, Lomer-Cottrell locks, Hirth locks or annihilations in face-centered cubics depending on the crystallography of intersecting dislocations. Following the difficulties associated with the experimental evaluation of the strength of the different types of intersections, small-scale studies of paired interactions between two discrete dislocations have been first carried out analytically using dislocation theory [207, 93, 209, 111], or numerically using atomistic and dislocation dynamics simulations [30, 203, 205, 256]. Later, Madec and co-workers proposed a numerical approach using DDD simulations to quantify the collective effect of specific dislocation interactions on larger scale simulations, and extracted representative dislocation-dislocation interaction coefficients [164, 165]. Following this approach, a hierarchy between slip system interactions was established and latent-hardening coefficients were obtained for FCC [65] and BCC [198] crystals. However, due to the additional complexities associated with HCP structures, such interaction coefficients have never been quantified for hexagonal magnesium.

Consequently, in this chapter, the regular DDD tool developed in Chapter 2 is used to extract latent-hardening parameters for pure hexagonal magnesium to be used in constitutive models at higher scales. This chapter is organized as follows. After a brief introduction on the role of latent-hardening parameters is given, the incorporation of dislocation-dislocation interactions at higher scales is presented in

Section 4.1.2. Then, the crystallography of HCP Mg is investigated and the different types of interactions that can occur during deformation are detailed in Section 4.2. In Section 4.3, the methodology for computing the hardening coefficients for single crystal Mg is presented while the validity of the additive rule used in constitutive models is examined in Section 4.4. The coefficients obtained and the effects of friction and mobilities are discussed in Section 4.5. Finally, the influence of the hardening coefficients on the macroscopic mechanical response of polycrystals is studied in Section 4.6 using the VPSC formalism.

4.1.1 Role of dislocation-dislocation interactions on strain hardening

In crystalline media, strain hardening primarily results from the interactions between lattice defects. Focusing on slip-driven plasticity in single crystals, it has been observed that, in addition to the lattice friction stress, the flow stress is proportional to the square root of the dislocation density (Taylor law) during plastic deformation of metals [127]. This law can be readily related to the forest hardening model, whereby the mean spacing between forest dislocations acting as obstacles for moving dislocations define the mean free path of dislocations [18, 52, 207, 136, 88].

While early models relied solely on the total dislocation density, Franciosi and Zaoui [90] and Lavrentev and Pokhil [141] proposed different extensions of the Taylor-law accounting for individual slip systems dislocation densities in which latent-hardening interaction coefficients were introduced. In these models, the latent-hardening coefficients account for the collective effect of the different dislocation-dislocation interactions between slip systems, and their importance in predicting the mechanical response of metals has been highlighted [65, 132]. For example, Hoc et al. [112] suggested that the orientation dependence of stage I deformation of face-centered cubic (FCC) crystals is strongly linked to the values of the hardening coefficients. In addition, the large values associated with interactions between slip systems with identical

Burgers vectors – referred to as collinear interactions – have provided an explanation for the unfavorable simultaneous activation of collinear systems [165, 64]. Also, latent hardening is expected to play an important role in the mechanical response for complex loadings paths [200, 95]. However, current constitutive models developed for HCP materials incorporate latent hardening models that are not directly related to the microstructure, but whose parameters are obtained by a fitting procedure. For example, slip induced hardening is described through a Voce law [195, 243] or using a single hardening coefficient in a dislocation density law [25, 189, 122].

Several efforts have been dedicated to evaluate these coefficients. Analytical approaches were used by Lavrentev and Pokhil [141] and Lavrentev [140], whereby values of the coefficients were assumed to be related to the energy gain due to pairwise interactions of dislocations. In addition, coefficients for different types of interactions were experimentally determined at different deformation stages for Mg [142, 141]. However, difficulties in measuring dislocation densities are likely to induce large uncertainty in experimental estimates for these coefficients.

Interestingly, DDD simulations have allowed for a numerical quantification of the hardening coefficients. Madec [164], Madec et al. [165] and Devincre et al. [65] have computed the coefficients for face-centered cubic metals, and were further able to construct junction formation maps depicting the nature of the interactions as a function of the relative orientation of the intersecting dislocations. In doing so, they also revealed the strong effect of collinear interactions on hardening [165]. Using the same approach, Queyreau et al. [198] computed the coefficients for BCC α-iron. In references [175] and [198], the effect of lattice friction on junction formation was also investigated.

However, compared to cubic crystals, considerably less DDD studies have been dedicated to the more complex case of HCP metals. As mentioned earlier, the primary reason lies in the increasing complexity associated with HCP crystals and the nature

of their dislocations. For instance, atomistic simulations have revealed 3D spread of prismatic dislocations cores [16] as well as strong anisotropies in dislocation mobilities and friction stress of slip systems [102]. Such anisotropies in core properties were supported by in situ TEM experiments in which the presence of long screw segments – suggesting significant lattice friction and low mobility of such dislocations – was observed for prismatic dislocations [37]. In a first attempt to mimic such anisotropic properties, Monnet et al. [176] used the DDD method in which dislocation mobilities were modelled in a phenomenological manner to study prismatic slip in Zirconium. As a result, it was found that junctions could not be formed between screw dislocations. Using DDD as well, Capolungo [39] investigated the strength of junctions in Mg relying on elastic interactions only. Similar work was further extended to Mg and Be [260]. In these works, the formation and destruction of a single binary junction was studied, leaving collective effects and other possible interactions pertaining to latent hardening aside. DDD was also used to show the weak effect of anisotropic calculations in the case of Mg, suggesting that an isotropic calculation could be safely used as an approximation [40]. Nevertheless, to the author's knowledge, except for ice single crystal [62], hardening coefficients have not been calculated for hexagonal materials.

In addition, although the flow stress dependence on dislocation density has been verified against experimental data, the validity of multi-slip extensions, such as the ones proposed by Franciosi and Zaoui (FZ) [90], and Lavrentev and Pokhil (LP) [141] has rarely been tested. To the author's knowledge, Queyreau et al. [198] were the only ones to verify the validity of the Franciosi and Zaoui formulation using DDD simulations on multi-slip conditions for BCC α-iron crystals, but the validity of the superposition principle expressed through these laws has never been discussed for HCP materials and the two multi-slip formulations have never been directly compared.

Consequently, the goals of this chapter are to (1) quantify latent hardening coefficients based on both the FZ and LP models, (2) identify which model is more appropriate for HCP metals, and (3) assess the importance of finely accounting for latent hardening on the mechanical response of pure magnesium [22].

4.1.2 Modelling of latent hardening in constitutive approaches

In current constitutive laws, the effects of dislocation interactions on the critical resolved shear stress (CRSS) are usually accounted for via an extended Taylor's law, which relates the CRSS on a given system to the total dislocation densities on all slip systems. In order to account for different strengths in dislocation–dislocation interactions, Franciosi and Zaoui [90] proposed the following expression for defining the CRSS τ_c^s on slip system s:

$$\tau_c^s = \tau_0^s + \mu b \sqrt{\sum_{s'} a_{FZ}^{ss'} \rho^{s'}} \tag{4.1}$$

where τ_c^s and τ_0^s denote the CRSS and the lattice friction stress on system s, respectively, μ and b are the shear modulus and magnitude of the Burgers vector, and $a_{FZ}^{ss'}$ denotes the interaction coefficient pertaining to the strength of the interaction between dislocations of systems s and s', associated with the FZ formulation. In this formulation, system s, on which the CRSS is evaluated, is generally referred to as the primary system, while the other slip systems s' are referred to as forest systems of system s.

Note that the FZ formulation was originally proposed for FCC materials for which the magnitude of the Burgers vector is identical for each slip system [90]. As a result, formulation (4.1) must be modified in order to be applicable for HCP materials in which the magnitude of the Burgers depends on the slip mode (e.g. the magnitude of Burgers vectors of $\langle a \rangle$ and $\langle c+a \rangle$ dislocations are different, see Section 4.2 for further details on the crystallography). In this work, the FZ formulation is extrapolated by

considering the magnitude of the primary Burgers vector b^s such that (4.1) rewrites:

$$\tau_c^s = \tau_0^s + \mu^s b^s \sqrt{\sum_{s'} a_{FZ}^{ss'} \rho^{s'}} \tag{4.2}$$

The validity of generalization (4.2) of the FZ formulation will be investigated in Section 4.4. Further, for the sake of consistency, μ is similarly replaced by the elastic shear modulus μ^s in the direction of slip system s. However, as HCP Mg is almost elastically isotropic [40], any differences in μ^s in equation (4.2) can be neglected [34].

Almost simultaneously with the introduction of the FZ formulation, Lavrentev and Pokhil [141] suggested a different expression for evaluating the CRSS:

$$\tau_c^s = \tau_0^s + \sum_{s'} \mu^{s'} b^{s'} a_{LP}^{ss'} \sqrt{\rho^{s'}} \tag{4.3}$$

In contrast with the FZ formulation, the LP formulation (4.3) was developed for a broader variety of crystal structures and has been particularly employed for HCP materials [141, 140, 34]. Therefore, further extension of (4.3) is not required.

Examining equations (4.2) and (4.3), FZ and LP expressions differ by two aspects: (1) the square root encompasses the summation of the dislocation densities in the FZ formulation whereas the summation is performed outside of the square root in the LP formulation and, (2) the magnitude of the Burgers vector is that of the primary slip system in the FZ expression whereas the Burgers vector is the one associated with the forest slip systems in the LP expression. As a consequence, the values of the FZ and LP coefficients are not expected to coincide.

As extensively discussed in reference [132], the CRSS can be interpreted as a measure of the average force required to bow dislocation segments across a forest of dislocations [111]. In line with this interpretation, a more complex form of expression (4.2) was proposed to account for contributions from line tension [132]. This extension was shown to be of significant importance at large dislocation densities.

Here, for the sake of simplicity, expressions (4.2) and (4.3) will be used to extract a latent hardening matrix from DDD simulations. Overall, the following features are expected to contribute to the strength of interactions between different systems: (1) Frank's rule, which identifies reactions that lead to a reduction in energy [111], (2) the distribution of line orientations upon intersection, and (3) the friction stresses on different systems. Note that in contrast to FCC materials – where all slip systems have identical properties –, differences in the Burgers vector, dislocation mobilities and friction stress associated with HCP slip systems are expected to yield a non-symmetrical latent hardening matrix. Details on the different components of this matrix for HCP Mg are presented in the following section.

At this point, it may also be noticed that, as discussed in references [21] and [132], slip system interactions are not limited to latent hardening effects. Indeed, as dislocation density based constitutive models account for stored dislocation densities, another feature that needs to be accounted for is the latent storage of dislocations associated with interactions between dislocations on different systems. This effect pertains to the transformation with strain of glissile to stored dislocation density, particularly through junction formation. Different approaches have been proposed in the literature to account for such effects. The present work is limited to the case of latent hardening, i.e., the effect of the evolution of dislocation densities on the CRSS.

4.2 Dislocation-dislocation interactions in pure Magnesium

4.2.1 Slip systems in hcp Magnesium

Pure magnesium has an hexagonal closed-packed (HCP) crystalline structure. The hexagonal structure is composed of compact layers of atoms stacked on top of each other and where the atoms of each layer are positioned in the lattice gaps of the preceding layer. In each layer, the packing consists of a pattern composed of six atoms arranged in an hexagonal shape and a seventh atom lying at the middle, as

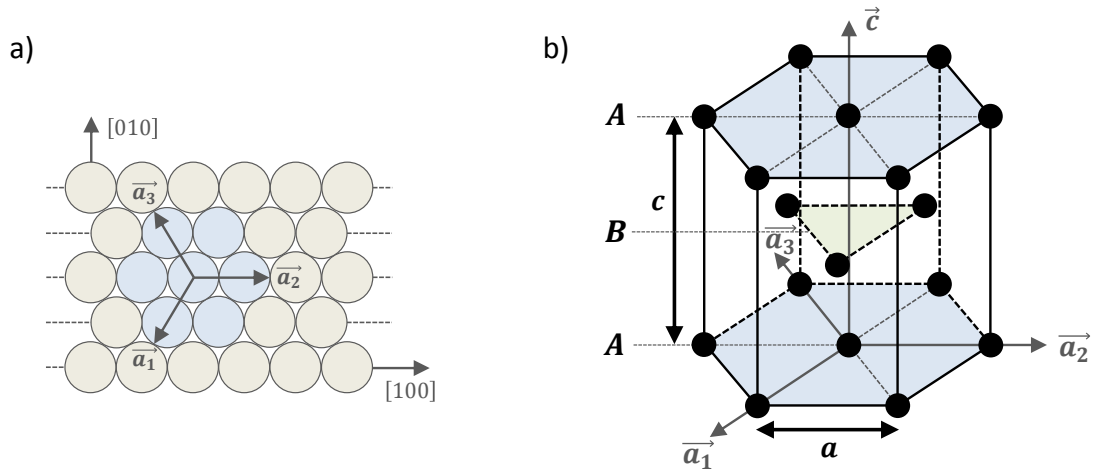


Figure 4.1: Schematic of the hexagonal closed-packed (HCP) crystalline structure. (a) Packing of atoms in the (0001) basal plane and its coordinate system $(\vec{\boldsymbol{a}}_1, \vec{\boldsymbol{a}}_2, \vec{\boldsymbol{a}}_3)$. (b) The unit HCP cell is composed of the stacking of (0001) basal layers in a $ABA...$ sequence, where atoms of the B layer are positioned in the gap of the A layers. For the sake of simplicity, the crystallographic elements (planes, Burgers vectors) of the HCP structure are given in the 4-axis coordinate system $(\vec{\boldsymbol{a}}_1, \vec{\boldsymbol{a}}_2, \vec{\boldsymbol{a}}_3, \vec{\boldsymbol{c}})$, where axes $\vec{\boldsymbol{a}}_i$ are given in (a), and $\vec{\boldsymbol{c}} = (0001)$ is perpendicular to the (0001) basal planes.

shown in figure 4.1(a). Using the 4-axis coordinate system $(\vec{\boldsymbol{a}}_1, \vec{\boldsymbol{a}}_2, \vec{\boldsymbol{a}}_3, \vec{\boldsymbol{c}})$ depicted in figure 4.1(b), each layer corresponds to a (0001) plane, referred to as the basal plane. As a result, the unit cell represented in figure 4.1(b) is composed of three layers of atoms ABA, for which atoms in layers A are arranged in the hexagonal shape, and where layer B is a triangle of atoms whose positions are aligned with the center of the triangles formed by the adjacent layers A. Since the 12 atoms at the corners of the top and bottom A layers are common to six unit cells, and those at the middle of the same layers are common to two unit cells, the HCP unit cell is composed of 6 atoms.

In the HCP structure, the lattice parameter a is given as the distance between the centers of subsequent atoms along the $\vec{\boldsymbol{a}}_i$ axes, while c denotes the distance between the centers of atoms in two subsequent A layers, as indicated in figure 4.1(b). In the ideal case where all atoms are considered as rigid spheres of identical radius, it can be shown that the ratio $c/a = \sqrt{8/3} \approx 1.633$. Interestingly, of all HCP crystals,

Table 4.1: List of the different possible slip modes and systems for the HCP structure.

Burgers vector	Slip direction	Slip plane	Number of systems
$\langle a \rangle$	$\langle 11\bar{2}0 \rangle$	basal (0001)	3
$\langle a \rangle$	$\langle 11\bar{2}0 \rangle$	first-order prismatic $\{10\bar{1}0\}$	3
$\langle a \rangle$	$\langle 11\bar{2}0 \rangle$	first-order pyramidal $\{10\bar{1}1\}$	6
$\langle c+a \rangle$	$\langle 11\bar{2}3 \rangle$	second-order pyramidal $\{11\bar{2}2\}$	6
$\langle c \rangle$	$\langle 0001 \rangle$	first-order prismatic $\{10\bar{1}0\}$	3
$\langle c \rangle$	$\langle 0001 \rangle$	second-order prismatic $\{11\bar{2}0\}$	3

magnesium has the nearest to the ideal ratio, with $c/a = 1.623$ [193].

Despite the apparent simplicity of the HCP crystalline structure, HCP metals activate a large variety of deformation modes and systems, including slip and twinning, generally resulting in a greater complexity in deformation mechanisms than in their body-centered cubic (BCC) and face-centered cubic (FCC) counterparts. In this chapter, the focus is placed on slip-mediated deformation only, i.e. deformation resulting from the motion of dislocations. Following the crystallographic arrangements of atoms in HCP structures, a list of all independent slip systems is reported in table 4.1. From a general perspective, there exist three families of Burgers vector and five families of planes, leading to a combination of six slip modes. This profusion of modes highlights the complexity of slip deformation compared to FCC materials in which the plastic slip activity is mediated by a unique mode.

In practice however, all slip systems listed in table 4.1 are typically not activated simultaneously, and their selection strongly depends on material properties such as lattice friction stresses and dislocation mobilities. In general, the basal systems are easily activated in all HCP metals. However, non-basal slip is commonly required to satisfy the von Mises criterion (five independent slip systems must activate to generate uniform plasticity [169]) and accommodate the deformation in the $\vec{\boldsymbol{c}}$ direction. In magnesium, the observed slip systems are depicted in figure 4.2. Besides the glide of $\langle a \rangle$ dislocations in (0001) basal planes, slip can occur via glide of $\langle a \rangle$ dislocations in

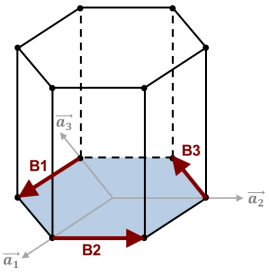
Table 4.2: Slip systems used to model hexagonal pure Mg in this work.

Mode	Slip system	Slip plane	Burgers vector
Basal $\langle a \rangle$	B1	(0001)	$\frac{1}{3}[2\bar{1}\bar{1}0]$
	B2	(0001)	$\frac{1}{3}[\bar{1}2\bar{1}0]$
	B3	(0001)	$\frac{1}{3}[\bar{1}\bar{1}20]$
Prismatic $\langle a \rangle$	P1	$(0\bar{1}10)$	$\frac{1}{3}[2\bar{1}\bar{1}0]$
	P2	$(10\bar{1}0)$	$\frac{1}{3}[\bar{1}2\bar{1}0]$
	P3	$(\bar{1}100)$	$\frac{1}{3}[\bar{1}\bar{1}20]$
Pyramidal $\langle c+a \rangle$	Pca1	$(\bar{2}112)$	$\frac{1}{3}[2\bar{1}\bar{1}3]$
	Pca2	$(1\bar{2}12)$	$\frac{1}{3}[\bar{1}2\bar{1}3]$
	Pca3	$(11\bar{2}2)$	$\frac{1}{3}[\bar{1}\bar{1}23]$
	Pca4	$(2\bar{1}\bar{1}2)$	$\frac{1}{3}[\bar{2}113]$
	Pca5	$(\bar{1}2\bar{1}2)$	$\frac{1}{3}[1\bar{2}13]$
	Pca6	$(\bar{1}\bar{1}22)$	$\frac{1}{3}[11\bar{2}3]$

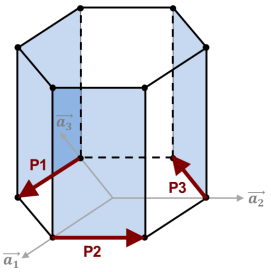
$\{10\bar{1}0\}$ first-order prismatic systems [86] and $\langle c+a \rangle$ dislocations in $\{11\bar{2}2\}$ second-order pyramidal systems [225, 188, 1]. Although glide of $\langle a \rangle$ dislocations on $\{10\bar{1}1\}$ first-order pyramidal systems have been reported [201] and have been considered in numerical investigations [102, 196], these are not considered in this work as it is assumed that basal and prismatic modes – that are easier to activate – provide sufficient $\langle a \rangle$ activity. As a result, the slip systems in HCP Mg considered in this work and their notation are reported in table 4.2. Note that throughout the rest of this chapter, for the sake of simplicity, prismatic systems will refer to first-order prismatic $\langle a \rangle\{10\bar{1}0\}$ systems, and reference to $\langle c+a \rangle$ slip systems will exclusively designate second-order pyramidal systems.

4.2.2 Geometry of interactions

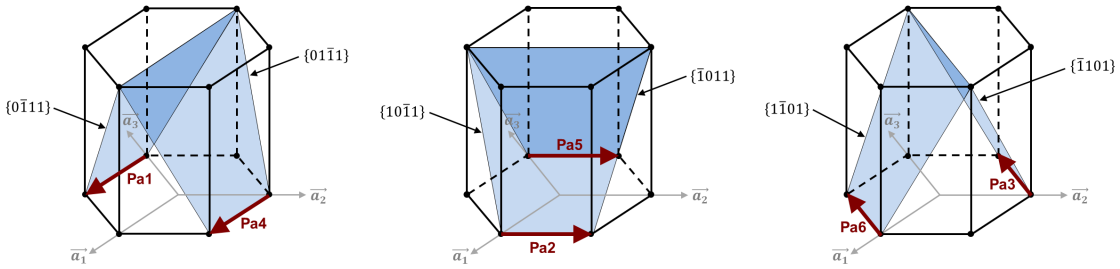
For the sake of simplicity, all dislocations are considered perfect in this work. In other words, it is assumed that the dislocation cores are constricted when interactions occur. Interactions between two dislocations can lead to different configurations depending



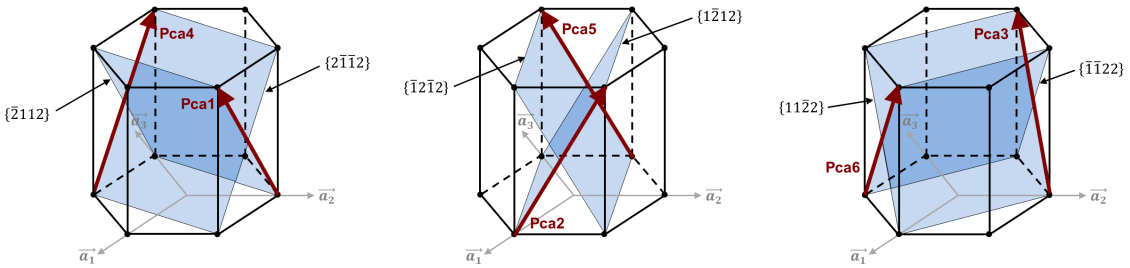
(a) $\langle a \rangle (0001)$ basal systems



(b) $\langle a \rangle \{10\bar{1}0\}$ prismatic systems



(c) $\langle a \rangle \{10\bar{1}1\}$ first-order pyramidal systems



(d) $\langle c+a \rangle \{11\bar{2}2\}$ second-order pyramidal systems

Figure 4.2: Schematic of the different slip systems observed in HCP Mg. The blue planes illustrate the different slip planes belonging to the same family. The red arrows delineate the Burgers vector associated with each slip plane.

on the slip systems and the orientation of the dislocation lines that interact. Four outcomes are possible as the result of line interactions: a repulsive interaction, a so called crossed state, junction formation or annihilation. Junctions will form if they result in a reduction in strain energy. Generally, Frank's criterion is used to determine the likelihood of junction formation between two dislocations with Burgers vectors $\vec{b}_1$ and $\vec{b}_2$. Given that the strain energy of a dislocation line is proportional

Table 4.3: Frank's criteria and corresponding reduction in strain energy for Burgers possible combinations in hexagonal Mg ($c/a = 1.623$).

Interaction $\vec{b}_1$	$\vec{b}_2$	Junction Burgers $\vec{b}_j = \vec{b}_1 + \vec{b}_2$	Initial energy $\|\vec{b}_1\|^2 + \|\vec{b}_2\|^2$	Junction energy $\|\vec{b}_1 + \vec{b}_2\|^2$	Frank's criterion	Reduction in strain energy
Interactions $\langle a\rangle/\langle a\rangle$						
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle a_i\rangle}$	$2\vec{b}_{\langle a_i\rangle}$	$2a^2$	$4a^2$	No	
$\vec{b}_{\langle a_i\rangle}$	$-\vec{b}_{\langle a_i\rangle}$	0	$2a^2$	0	Annihilation	$2a^2$
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle a_j\rangle}$	$-\vec{b}_{\langle a_k\rangle}$	$2a^2$	a^2	Yes	a^2
$\vec{b}_{\langle a_i\rangle}$	$-\vec{b}_{\langle a_j\rangle}$	$\vec{b}_{\langle a_i\rangle} - \vec{b}_{\langle a_j\rangle}$	$2a^2$	$3a^2$	No	
Interactions $\langle c+a\rangle/\langle c+a\rangle$						
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle c+a_i\rangle}$	$2\vec{b}_{\langle c+a_i\rangle}$	$2(c^2+a^2)$	$4(c^2+a^2)$	No	
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle c-a_i\rangle}$	$2\vec{b}_{\langle c\rangle}$	$2(c^2+a^2)$	$4c^2$	No	
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle -c+a_i\rangle}$	$2\vec{b}_{\langle a_i\rangle}$	$2(c^2+a^2)$	$4a^2$	Yes	$3.268a^2$
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle -c-a_i\rangle}$	0	$2(c^2+a^2)$	0	Annihilation	$7.268a^2$
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle c+a_j\rangle}$	$2\vec{b}_{\langle c\rangle} - \vec{b}_{\langle a_k\rangle}$	$2(c^2+a^2)$	$4c^2+a^2$	No	
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle c-a_j\rangle}$	$2\vec{b}_{\langle c\rangle} + \vec{b}_{\langle a_i\rangle} - \vec{b}_{\langle a_j\rangle}$	$2(c^2+a^2)$	$4c^2+3a^2$	No	
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle -c+a_j\rangle}$	$-\vec{b}_{\langle a_k\rangle}$	$2(c^2+a^2)$	a^2	Yes	$6.268a^2$
$\vec{b}_{\langle c+a_i\rangle}$	$\vec{b}_{\langle -c-a_j\rangle}$	$\vec{b}_{\langle a_i\rangle} - \vec{b}_{\langle a_j\rangle}$	$2(c^2+a^2)$	$3a^2$	Yes	$4.268a^2$
Interactions $\langle a\rangle/\langle c+a\rangle$						
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle \pm c+a_i\rangle}$	$\pm\vec{b}_{\langle c\rangle} + 2\vec{b}_{\langle a_i\rangle}$	c^2+2a^2	c^2+4a^2	No	
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle \pm c-a_i\rangle}$	$\pm\vec{b}_{\langle c\rangle}$	c^2+2a^2	c^2	Yes	$2a^2$
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle \pm c+a_j\rangle}$	$\pm\vec{b}_{\langle c\rangle} - \vec{b}_{\langle a_k\rangle}$	c^2+2a^2	c^2+a^2	Yes	a^2
$\vec{b}_{\langle a_i\rangle}$	$\vec{b}_{\langle \pm c-a_j\rangle}$	$\pm\vec{b}_{\langle c\rangle} + \vec{b}_{\langle a_i\rangle} - \vec{b}_{\langle a_j\rangle}$	c^2+2a^2	c^2+3a^2	No	

to the square of the magnitude of its Burgers vector, a reduction in energy upon dislocation intersection is expected to occur when $\|\vec{b}_1 + \vec{b}_2\|^2 < \|\vec{b}_1\|^2 + \|\vec{b}_2\|^2$ [111], where $\vec{b}_j = \vec{b}_1 + \vec{b}_2$ denotes the Burgers vector of the resulting junction. Conversely, junction formation is expected not to be energetically favorable when this criterion is not satisfied. In the case of Mg dislocation-dislocation interactions, Frank's criterion for the possible combinations of Burgers vectors is listed in table 4.3. Note that annihilation can only occur between collinear slip systems, i.e. when the two reacting dislocations have opposite Burgers vectors. In this case, the resulting Burgers vector vanishes, leading to a total reduction in strain energy.

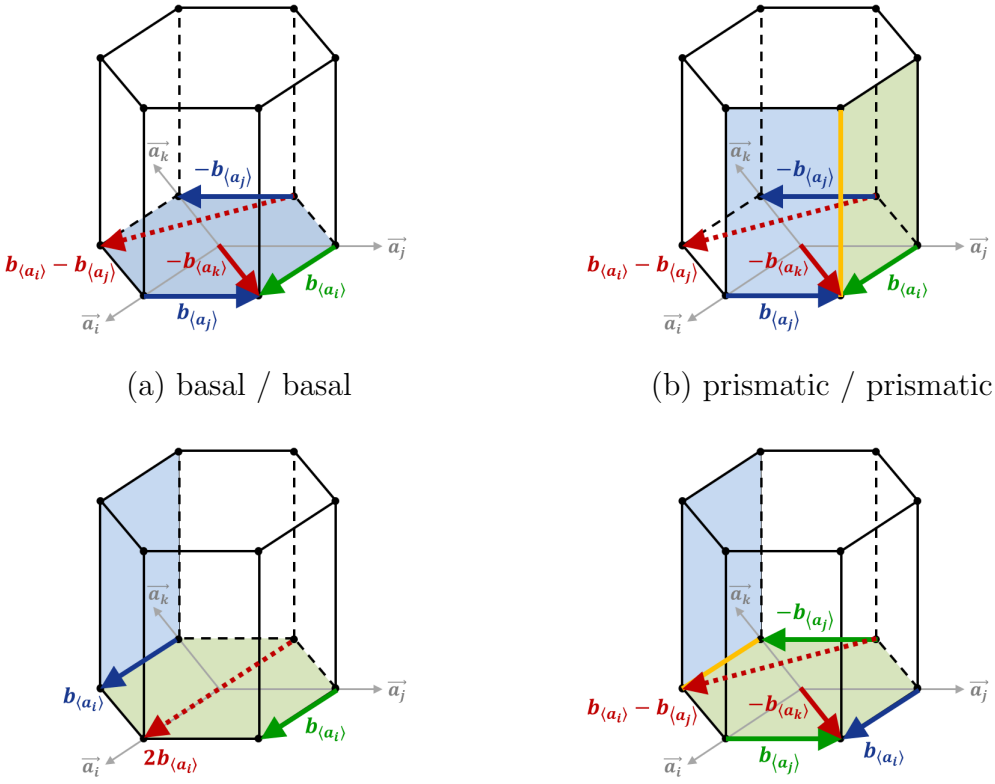
Numerically, as detailed in Section 2.6, the formation of junctions in DDD simulations is governed by the zipping/unzipping mechanism [30] whereby the most energetically favorable dislocation-dislocation configuration is determined via the use

of a dissipation criterion. When a junction does not form, the intersecting dislocations do not combine, but only interact elastically. This elastic interaction can be classified either as a crossed state – where attractive interactions are not sufficient to overcome the line tension such that a junction cannot form [166] – or as a repulsive configuration.

Considering the slip systems listed in table 4.2, apart from self-interactions, 11 possible different crystallographic configurations of interactions can be distinguished. These can be grouped into three classes of interactions: (1) $\langle a \rangle / \langle a \rangle$ interactions, (2) $\langle c+a \rangle / \langle c+a \rangle$ interactions and (3) $\langle a \rangle / \langle c+a \rangle$ interactions. These are illustrated in figures 4.3 to 4.5 and the following convention is used: blue and green vectors indicate the Burgers vector of interacting segments. For each of these, the two possible directions of the Burgers vector are considered (blue vectors). Red vectors indicate the possible Burgers vector resulting from the interaction. Red solid and dashed lines correspond to junction Burgers vector that do and do not lead to a minimization of energy, respectively. Finally, when applicable, a yellow line is used to denote the orientation of the junction.

Within the set of interactions between $\langle a \rangle$ dislocations, six different cases can occur, namely basal/basal, prismatic/prismatic, collinear basal/prismatic and prismatic/basal, and non-collinear basal/prismatic and prismatic/basal intersections. These are depicted in figure 4.3. In the case of collinear interactions, Frank's rule suggests that only annihilation can occur, as the formation of a junction with $2\vec{\boldsymbol{b}}_{\langle a_i \rangle}$ Burgers vector does not lead to a reduction in strain energy. In all the other cases, only combinations between dislocations of Burgers $+\vec{\boldsymbol{b}}_{\langle a_i \rangle}$ and $+\vec{\boldsymbol{b}}_{\langle a_j \rangle}$ can lead to junction formation with a resulting $-\vec{\boldsymbol{b}}_{\langle a_k \rangle}$ Burgers vector. Note that for all $\langle a \rangle / \langle a \rangle$ junctions, the resulting Burgers vector lies in the basal plane, such that all of them are glissile.

In the case of $\langle c+a \rangle / \langle c+a \rangle$ interactions, two types are considered. First are



(a) basal / basal

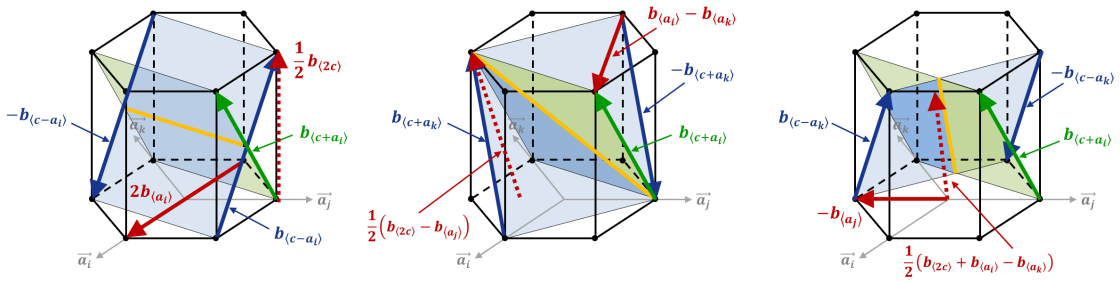
(b) prismatic / prismatic

(c) non-collinear basal / prismatic

(d) collinear basal / prismatic

Figure 4.3: Schematic of $\langle a \rangle / \langle a \rangle$ interactions.

interactions between slip systems whose Burgers vectors share the same $\langle a \rangle$ components, such as reactions between $\vec{\boldsymbol{b}}_{\langle c+a_i \rangle}$ and $\vec{\boldsymbol{b}}_{\langle -c+a_i \rangle}$ Burgers vectors. Here, we refer to these reactions as *semi-collinear* interactions, as they share only one component of their Burgers vectors, as opposed to collinear interactions which are typically related to interactions between slip systems having the same Burgers vectors. As depicted in figure 4.4, semi-collinear interactions involve intersections between semi-collinear slip systems, i.e. between systems Pca1 and Pca4, Pca2 and Pca5, and Pca3 and Pca6, following the convention adopted in table 4.2, respectively. For these semi-collinear systems, Frank's rule suggests that only reactions between Burgers vectors that do not lead to the cancelling of the $\langle a \rangle$ component are energetically favorable, leading to a resulting $\pm 2\vec{\boldsymbol{b}}_{\langle a_i \rangle}$ Burgers vector where the $\langle c \rangle$ component has cancelled out. Second,



(a) semi-collinear pyramidal / pyramidal

(b) $\langle c + a_i\rangle/\langle c + a_k\rangle$ pyramidal / pyramidal

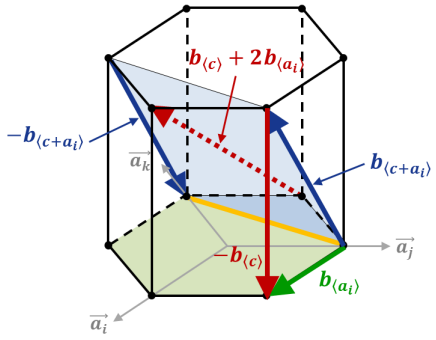
(c) $\langle c + a_i\rangle/\langle c - a_k\rangle$ pyramidal / pyramidal

Figure 4.4: Schematic of $\langle c + a\rangle/\langle c + a\rangle$ interactions.

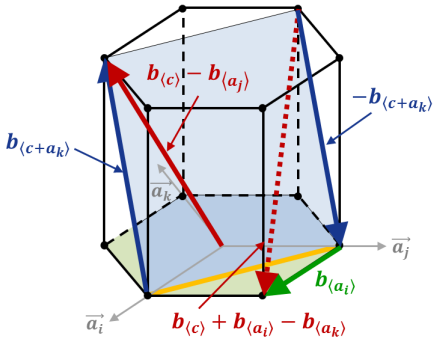
non-semi-collinear pyramidal/pyramidal interactions involve both $\langle c + a_i\rangle/\langle c + a_k\rangle$ and $\langle c + a_i\rangle/\langle c - a_k\rangle$ Burgers combinations. In both cases, favorable junctions are those pertaining to a cancellation of the $\langle c\rangle$ component.

The $\langle a\rangle/\langle c + a\rangle$ interactions include combinations of dislocations from basal slip systems with pyramidal systems and dislocations from prismatic slip systems with pyramidal systems. In the case of semi-collinear reactions, a reduction in strain energy is expected only for reactions leading to pure $\langle c\rangle$ junctions. For other reactions, junctions are expected to form when the combination of $\langle a\rangle$ components leads to another $\langle a\rangle$ component, as in the case of $\langle a\rangle/\langle a\rangle$ reactions described earlier. Schematics of these interactions are given in figure 4.5.

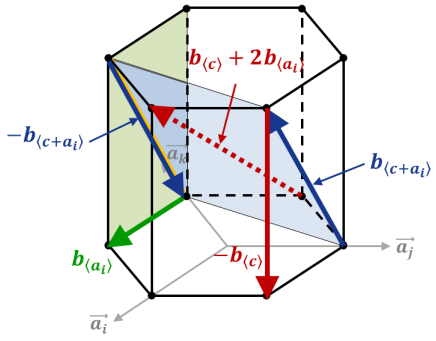
As a result, the full latent hardening matrix that needs to be quantified contains 19 distinct coefficients for hexagonal Mg, each pertaining to one specific interaction. These interactions are listed in table 4.4 and the corresponding components of the hardening matrix are reported in table 4.5. As mentioned earlier, the hardening matrix is not expected to be symmetric when interactions occur between different slip modes. By comparison, in the case of FCC, the hardening matrix is symmetric and populated with only six independent coefficients since interactions solely occurs between slip systems belonging to the same $\frac{1}{2}\langle\bar{1}10\rangle\{111\}$ slip mode [90].



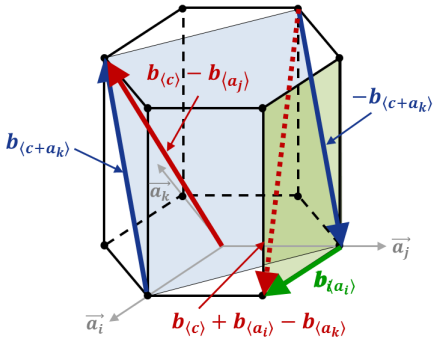
(a) semi-collinear basal / pyramidal



(b) semi-collinear prismatic / pyramidal



(c) basal / pyramidal



(d) prismatic / pyramidal

Figure 4.5: Schematic of $\langle a\rangle/\langle c+a\rangle$ interactions.

4.3 Latent-hardening coefficients quantification

4.3.1 Methodology and simulations setting

In this work, the parallel three-dimensional regular DDD nodal code developed in Chapter 2 was further modified such that latent-hardening coefficients can be extracted. The use of the regular DDD simulation is allowed here since it has been demonstrated that an elastic approximation could be safely used to model HCP Mg [40] – the anisotropy ratio $A \approx 0.979$ (see equation (3.63)) of HCP Mg is very close to unity. Therefore, the non-singular isotropic analytical formulation developed by Cai and co-workers [35] will be used to evaluate dislocation-dislocation elastic interactions.

The methodology used here to extract latent-hardening coefficients for Mg is similar to that used in the work of Devincre and co-workers to compute coefficients

Table 4.4: Different types of interactions to be quantified in HCP Mg.

Interaction #	Designation
S1	Basal self-interaction (not computed)
S2	Prismatic self-interaction (not computed)
S3	Pyramidal $\langle c+a \rangle$ self-interaction (not computed)
1	Coplanar basal/basal (not computed)
2	Prismatic/prismatic
3	Collinear basal/prismatic
4	Non-collinear basal/prismatic
5	Collinear prismatic/basal
6	Non-collinear prismatic/basal
7	Semi-collinear basal/pyramidal $\langle c+a \rangle$
8	Non-collinear basal/pyramidal $\langle c+a \rangle$
9	Semi-collinear prismatic/pyramidal $\langle c+a \rangle$
10	Non-collinear prismatic/pyramidal $\langle c+a \rangle$
11	Semi-collinear pyramidal $\langle c+a \rangle$/basal
12	Non-collinear pyramidal $\langle c+a \rangle$/basal
13	Semi-collinear pyramidal $\langle c+a \rangle$/prismatic
14	Non-collinear pyramidal $\langle c+a \rangle$/prismatic
15	Semi-collinear pyramidal $\langle c+a \rangle$/pyramidal $\langle c+a \rangle$
16	Non-collinear pyramidal $\langle c+a \rangle$/pyramidal $\langle c+a \rangle$

for FCC materials [65]. To obtain the coefficient value for one specific interaction, the method consists in letting a mobile dislocation lying on a primary system glide through a prescribed forest of dislocations. The primary and the forest systems are chosen such that the possible intersections between primary and forest dislocations will lead to one type of reaction only. Consequently, in the case of the FZ formulation, the value of the coefficient corresponding to the specific type of interaction p/f can be directly obtained from (4.2):

$$a_{FZ}^{pf} = \left(\frac{\tau^p - \tau_0^p}{\mu b^p \sqrt{\rho^f}} \right)^2 \tag{4.4}$$

while in the case of the LP formulation, the coefficient value is extracted from equation

Table 4.5: Components of the latent hardening matrix for HCP Mg. Definitions of the coefficient numbers are given in table 4.4.

	Forest systems	B1	B2	B3	P1	P2	P3	Pca1	Pca2	Pca3	Pca4	Pca5	Pca6
Primary systems	B1	S1	1	1	3	4	4	7	8	8	7	8	8
	B2	1	S1	1	4	3	4	8	7	8	8	7	8
	B3	1	1	S1	4	4	3	8	8	7	8	8	7
	P1	5	6	6	S2	2	2	9	10	10	9	10	10
	P2	6	5	6	2	S2	2	10	9	10	10	9	10
	P3	6	6	5	2	2	S2	10	10	9	10	10	9
	Pca1	11	12	12	13	14	14	S3	16	16	15	16	16
	Pca2	12	11	12	14	13	14	16	S3	16	16	15	16
	Pca3	12	12	11	14	14	13	16	16	S3	16	16	15
	Pca4	11	12	12	13	14	14	15	16	16	S3	16	16
	Pca5	12	11	12	14	13	14	16	15	16	16	S3	16
	Pca6	12	12	11	14	14	13	16	16	15	16	16	S3

(4.3) as:

$$a_{LP}^{pf} = \frac{\tau^p - \tau_0^p}{\mu b^p \sqrt{\rho^f}} = \sqrt{a_{FZ}^{pf}} \cdot \frac{b^p}{b^f} \tag{4.5}$$

In expressions (4.4) and (4.5), a_{FZ}^{pf} and a_{LP}^{pf} denote the coefficients characterizing the strength of interactions occurring when a moving dislocation belonging to primary system p glide through a forest of dislocations lying on system(s) f, for the FZ and LP formulations, respectively. The total density of the dislocations composing the forest is denoted with ρ^f. According to equations (4.4) and (4.5), the value of the coefficient computed depends on the forest density. As discussed in references [65, 132], the value of the coefficient actually presents a logarithmic dependence on forest density. As a result, the coefficient is computed for a reference forest density and its value is then be extrapolated for different densities through a logarithmic corrective term. In the simulations performed in this work, the reference total forest density is set to $\rho_{ref} = \rho^f = 10^{12} m^{-2}$, for a simulation volume of $4.82 \times 4.14 \times 5.68 \mu m^3$.

The forest is represented by sets of initially straight dislocation segments, each of length $1.6 \mu m$, randomly generated such that the mean free path of the moving

dislocations remains close to $1\sqrt{\rho^f}$. The dislocation orientation – defined by the angle between dislocation lines direction and their Burgers vectors – is randomly generated, leading to a random distribution of edge, screw and mixed characters. Such an initial forest configuration allows us to have enough dislocation lines for the simulations to be statistically representative of an actual microstructure. As a matter of fact, care must be taken in choosing large enough simulation volumes allowing for sufficiently large number of intersection events to occur. In other words, too few forest dislocations may bias the desired result on collective interaction effects, as all intersection orientations may not be represented.

The primary density is initially composed of six mobile dislocations of length $7.2\mu m$, artificially pinned at their ends. In order to avoid dipoles interactions leading to self-hardening, apart from self-forces, primary dislocations are only subjected to the elastic stress fields of the forest dislocations and to the macroscopic loading. Conversely, in order for the forest density to remain roughly constant during the simulations, forest dislocations are not subjected to the external stress arising from the applied strain rate, but only to internal stresses induced by elastic stress fields of other dislocations present in the volume.

In order to evaluate the different interaction coefficients presented in tables 4.4 and 4.5, combinations of primary and forest slips systems for each type of interaction are reported in table 4.6. Here, it is first interesting to notice that self-hardening and basal/basal interactions are special cases as they involve coplanar interactions. In both these cases, two mechanisms are expected to govern the interactions: (1) annihilations when dislocations are lying on the same slip plane and (2) dipolar interactions when dislocations are lying on parallel but distant slip planes. However, it is not clear how such interactions can be accurately quantified since the distinction between primary and forest systems in the current formalism may be delicate, while it remains unclear how to ensure that the occurrence of mechanisms (1) and (2) is

Table 4.6: Primary and forest slip systems used to compute the hardening matrix components. The notation are those introduced in table 4.2

Coefficient #	Interaction designation	Primary system	Forest system(s)
S1	Basal self-interaction	Not computed	
S2	Prismatic self-interaction	Not computed	
S3	Pyramidal $\langle c+a \rangle$ self-interaction	Not computed	
1	Coplanar basal/basal	Not computed	
2	Prismatic/prismatic	P1	P2, P3
3	Collinear basal/prismatic	B1	P1
4	Non-collinear basal/prismatic	B1	P2, P3
5	Collinear prismatic/basal	P1	B1
6	Non-collinear prismatic/basal	P1	B2, B3
7	Semi-collinear basal/pyramidal $\langle c+a \rangle$	B1	Pca1, Pca4
8	Non-collinear basal/pyramidal $\langle c+a \rangle$	B1	Pca2, Pca3, Pca5, Pca6
9	Semi-collinear prismatic/pyramidal $\langle c+a \rangle$	P1	Pca1, Pca4
10	Non-collinear prismatic/pyramidal $\langle c+a \rangle$	P1	Pca2, Pca3, Pca5, Pca6
11	Semi-collinear pyramidal $\langle c+a \rangle$/basal	Pca1	B1
12	Non-collinear pyramidal $\langle c+a \rangle$/basal	Pca1	B2, B3
13	Semi-collinear pyramidal $\langle c+a \rangle$/prismatic	Pca1	P1
14	Non-collinear pyramidal $\langle c+a \rangle$/prismatic	Pca1	P2, P3
15	Semi-collinear pyramidal $\langle c+a \rangle$/pyramidal $\langle c+a \rangle$	Pca1	Pca4
16	Non-collinear pyramidal $\langle c+a \rangle$/pyramidal $\langle c+a \rangle$	Pca1	Pca2, Pca3, Pca5, Pca6

representative of real microstructural events. Therefore, and because our interest primarily lies in computing latent-hardening coefficients, self-hardening and basal/basal coefficients are not considered in the present work, and their evaluation should be undertaken in a dedicated study, as proposed in reference [66] for FCC crystals.

As a result, 15 distinct dislocation interaction coefficients need to be determined: these correspond to coefficient numbers 2 to 16 in tables 4.5 and 4.6. As already mentioned, the hardening matrix for HCP Mg is not expected to be symmetric. For instance, as reported in table 4.6, two different simulations will be used to determine the coefficients associated with collinear basal/prismatic interactions and collinear prismatic/basal interactions. Differences in the values of these coefficients are expected to result from the different dislocation properties associated with each slip mode. Specifically, the values of the friction stresses and mobilities used in this work are reported in table 4.7. Peierls stresses values have been extracted from experimental results obtained by Conrad and Robertson [50] for basal slip and by Reed-Hill and

Table 4.7: Parameters used for each slip mode in HCP Mg. Friction stresses are extracted from Conrad and Robertson [50], Reed-Hill and Robertson [201], Staroselsky and Anand [224], while mobilities were computed by Groh et al. [102].

Property	Basal systems	Prismatic systems	Pyramidal $\langle c+a \rangle$ systems
Friction stress (MPa)	0.52	39.2	105
Edge drag coefficient (Pa.s)	4.7×10^{-6}	7.7×10^{-6}	8.0×10^{-5}
Screw drag coefficient (Pa.s)	1.3×10^{-5}	3.7×10^{-5}	5.0×10^{-4}

Robertson [201] for prismatic systems, while numerical values obtained in matching experimental data for second-order pyramidal systems in [224] are used. Dislocation mobilities values are taken as those computed in the work of Groh and co-workers [102] using both molecular statics and molecular dynamics, in which the strong anisotropies between slip modes and dislocations characters was revealed.

Further, to perform a complete investigation, the effects of dislocation mobilities and lattice friction stress on latent-hardening are investigated. To this end, the coefficients are computed for three different cases:

(1) Different dislocation mobilities with friction stress

(2) Different dislocation mobilities without friction stress

(3) Same dislocation mobilities without friction stress

In cases (1) and (2), different mobilities are assigned to edge and screw dislocations, as reported in table 4.7. Note that sophisticated phenomenological approaches, such as those based on Arrhenius law [176], are required to reproduce the kink-pair mechanism controlling the screw dislocation mobility in HCP. However, for the sake of simplicity here, a linear interpolation between these two ideal characters is performed for mixed dislocations.

In order for the different simulations in table 4.6 to be comparable, the crystal is rotated for each coefficient calculation to ensure that the Burgers vectors and the normal of the slip plane of the primary gliding dislocations are aligned along the

Table 4.8: Material parameters used to model pure hexagonal Mg.

Material parameter	Value
Shear modulus	$\mu = 16.39$ GPa
Poisson's ratio	$\nu = 0.34$
Elastic constants	$C_{11} = C_{22} = 59.74$ GPa
	$C_{33} = 61.70$ GPa
	$C_{44} = C_{55} = C_{66} = 16.39$ GPa
	$C_{12} = 26.24$ GPa
	$C_{13} = C_{23} = 21.70$ GPa
Lattice parameters	$a = 3.21 \times 10^{-10}$ m
	$c/a = 1.623$

same direction for each simulation, such that primary dislocations are subjected to the same resolved shear stress in each case. Further, to account for statistical distribution effects, each coefficient is averaged over five different simulation configurations. For each batch of computed coefficients (here a batch refers to the computation of the 15 coefficients for the 3 cases, i.e. 45 simulations), forest dislocations were generated using the same random seed positions for all simulations, allowing for fair comparison between different coefficients values. Overall five different dislocation seeds were used in this work, for a total of $5 \times 45 = 225$ simulations. In all cases, the initial primary dislocations were pure edge dislocations and were allowed to reach a relaxed configuration under zero applied strain before the strain rate $\dot{\epsilon} = 500s^{-1}$ was applied. An example of the initial and final dislocation configurations is given in figure 4.6. Single crystal Mg material parameters used in this work are reported in table 4.8.

4.3.2 Results

4.3.2.1 Validation of the current approach

As a way to validate the approach and the regular DDD code developed in Chapter 2, the interaction coefficients are first computed for FCC materials and compared with the most recent values obtained by Devincre and co-workers [65, 132]. To this end, simulations were performed on FCC Copper and the coefficients were extracted

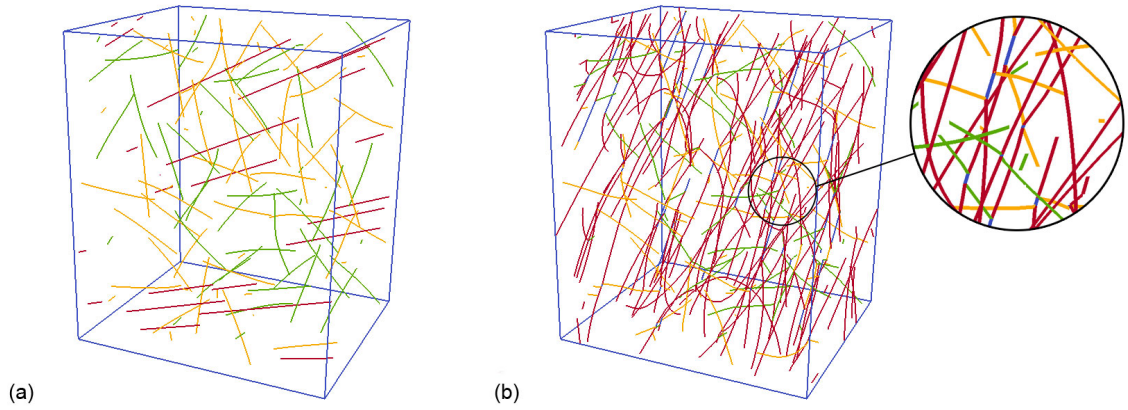


Figure 4.6: Example of (a) an initial dislocation configuration after relaxation. (b) Same dislocation configuration after 0.2% plastic strain. Primary dislocations are shown in red, forest dislocations in yellow and green, and junctions in blue.

Table 4.9: Interaction coefficients computed for FCC using DDD simulations and obtained with the FZ formulation.

	Hirth interaction	Lomer interaction	Collinear interaction
Devincre et al.	0.070 [132]	0.122 ± 0.012 [65]	0.625 ± 0.044 [65]
This work	0.084 ± 0.007	0.118 ± 0.005	0.657 ± 0.105

for Hirth, Lomer and collinear interactions. As reported in table 4.9, the present code gives a similar hierarchy and very close values for the interaction coefficients compared to those computed in [65, 132] in the case of the FZ formulation.

This preliminary result is important for two reasons. First, it ensures that the current approach and simulation setting to compute coefficients for Mg are consistent with the seminal work of Madec [165], Devincre [65], Kubin [132] and co-workers to numerically evaluate dislocation-dislocation interaction coefficients. Second, it constitutes an excellent validation procedure for the DDD code developed in Chapter 2. As many complex numerical techniques, an important sensitivity to numerical parameters and approaches is sometimes observed in DDD simulations. Specifically, the numerical approaches used in the work of Devincre and co-workers [65] and this work differ on two remarkable aspects: a edge / screw model with a fine discretization

length ($l_{avg} \approx 10b$) is used in [65] while a nodal scheme with a coarser discretization ($l_{avg} \approx 500b$) is used here, and a criterion on static equilibrium is used for junction zipping/unzipping in [65] while a dissipative approach is used in the current work (see Section 2.6). Consequently, the convergence that is observed here between the different regular approaches reinforces the overall confidence on the validity and relevance of DDD simulations in general.

4.3.2.2 Mg interaction coefficients

The evolution of all FZ interactions coefficients for Mg computed for case (1) (i.e. accounting for the lattice friction stresses and the system mobilities given in table 4.7) as a function of strain are plotted in figure 4.7.

As shown in figure 4.7, the values of the coefficients increase before oscillating around a quasi-stable value. This is because during the simulated deformation, the flow stress τ^p on the primary system increases until it reaches a quasi-saturation value. The saturation in the flow stress corresponds to the regime where the imposed strain is entirely accommodated by primary dislocation glide through the forest. However, when this regime is established, fluctuations remain. These are due to interactions between dislocations. For example, when one primary dislocation gets pinned by two forest dislocations, an increase in the flow stress is required for the moving dislocation to overcome this obstacle. At some point, the resolved shear stress becomes high enough such that the dislocation can bow-out or break away from the forest, leading to a decrease in the flow stress as glide is suddenly able to relax the stress that has built up. During this process, once the quasi-stationary regime is established, the value of the primary flow stress is seen to oscillate around an average value. As expected from equation (4.2) (respectively (4.3)), similar oscillations are reflected in the coefficient values, given that the forest density is constant during deformation. Therefore a measure of the coefficients can be given as their average value once the

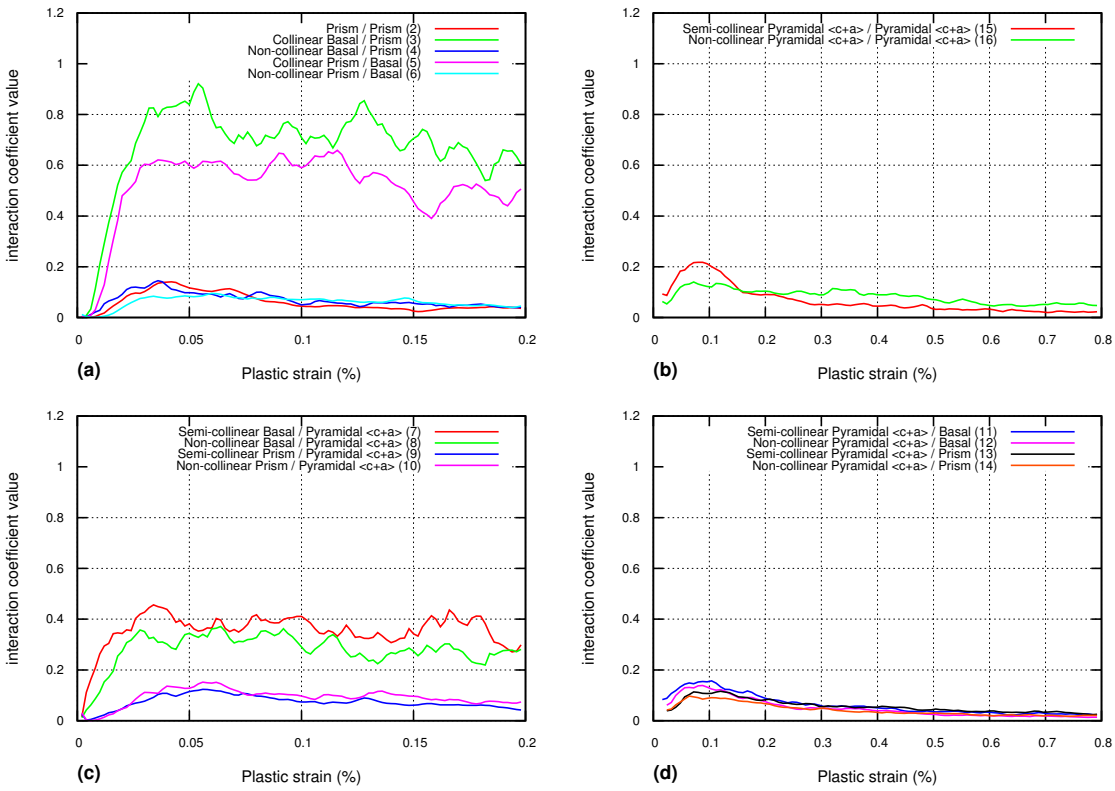


Figure 4.7: Evolution of the averaged values (over 5 simulations) of the interaction coefficients versus plastic strain computed by DDD simulations using the FZ formulation. (a) $\langle a \rangle / \langle a \rangle$ interactions, (b) $\langle c+a \rangle / \langle c+a \rangle$ interactions, (c) basal/pyramidal and prismatic/pyramidal interactions and (d) pyramidal/basal and pyramidal/prismatic interactions.

quasi-static regime is established, i.e., past a certain amount of strain.

Further, as reported in references [65, 198], such type of simulations are expected to be rather insensitive to strain rate when the waiting time of moving dislocations in the quasi-static regime is larger than their gliding time. The results reported in this work are obtained for an imposed strain rate $\dot{\epsilon} = 500 s^{-1}$. Following the relatively large value of the drag coefficient associated with the pyramidal systems, care was taken to ensure that the simulation results involving these systems were falling in similar ranges for higher strain rates. Consequently, simulations for these coefficients were performed up to 0.8% plastic strain to ensure that the quasi-static regime is attained, as reported in figures 4.7(b) and (d), while 0.2% plastic strain is found to

Table 4.10: Averaged values of the interaction coefficients computed for hexagonal Mg for a reference density $\rho_{ref} = 10^{12}m^{-2}$.

Coef.	Interaction	FZ Rank	FZ value ($a_{FZ}^{ss'}$)	LP Rank	LP Value ($a_{LP}^{ss'}$)
2	Prismatic/prismatic	10	**0.038** $\pm$ 0.008	12	**0.193** $\pm$ 0.020
3	Collinear basal/prismatic	1	**0.707** $\pm$ 0.029	1	**0.841** $\pm$ 0.017
4	Non-collinear basal/prismatic	8	**0.054** $\pm$ 0.006	9	**0.232** $\pm$ 0.014
5	Collinear prismatic/basal	2	**0.535** $\pm$ 0.096	2	**0.729** $\pm$ 0.067
6	Non-collinear prismatic/basal	7	**0.060** $\pm$ 0.005	7	**0.245** $\pm$ 0.009
7	Semi-collinear basal/pyramidal	3	**0.367** $\pm$ 0.067	3	**0.317** $\pm$ 0.029
8	Non-collinear basal/pyramidal	4	**0.293** $\pm$ 0.031	5	**0.284** $\pm$ 0.015
9	Semi-collinear prismatic/pyramidal	6	**0.068** $\pm$ 0.023	14	**0.135** $\pm$ 0.023
10	Non-collinear prismatic/pyramidal	5	**0.088** $\pm$ 0.011	13	**0.155** $\pm$ 0.010
11	Semi-collinear pyramidal/basal	13	**0.017** $\pm$ 0.003	6	**0.246** $\pm$ 0.018
12	Non-collinear pyramidal/basal	15	**0.011** $\pm$ 0.004	11	**0.197** $\pm$ 0.033
13	Semi-collinear pyramidal/prismatic	11	**0.025** $\pm$ 0.005	4	**0.301** $\pm$ 0.031
14	Non-collinear pyramidal/prismatic	14	**0.015** $\pm$ 0.002	8	**0.231** $\pm$ 0.019
15	Semi-collinear pyramidal/pyramidal	12	**0.018** $\pm$ 0.004	15	**0.132** $\pm$ 0.017
16	Non-collinear pyramidal/pyramidal	9	**0.042** $\pm$ 0.005	10	**0.205** $\pm$ 0.012

be sufficient to evaluate the interaction coefficients for basal and prismatic primary systems.

The values for both FZ and LP formulations averaged over the quasi-static regime are reported in table 4.10. As already mentioned, given that both FZ and LP approaches differ in their formulation, the interaction coefficients associated with each of them are not expected to coincide. This clearly appears in the coefficients values reported in table 4.10. Also, except for the collinear interactions, i.e. these pertaining to annihilation events, the interaction strength hierarchy established using the FZ formulation differs from that using the LP formulation. As expected from equation (4.3), the coefficients that show the greatest sensitivity to the formulation used are those related to interactions for which the magnitude of the primary Burgers vector differ from that of the forest, namely $\langle a \rangle / \langle c+a \rangle$ and $\langle c+a \rangle / \langle a \rangle$ interactions ($b_{\langle a \rangle} = 0.321nm$ and $b_{\langle c+a \rangle} = 0.612nm$ for HCP Mg).

Therefore, at this stage, it appears necessary to test for the validity and accuracy of both formulations before drawing conclusions on the strength of the interactions.

4.4 Examination of the validity of current hardening laws

As reported in Section 4.3.2.2, the use of the Franciosi and Zaoui (FZ) [90] law or Lavrentev and Pokhil (LP) [141] formulation leads to distinct results for the interaction coefficients. More importantly, the hierarchy established for the interaction strengths is significantly different between both formulations (see table 4.10). Therefore, in order to draw physical interpretations of the interaction coefficients results, both laws need to be compared in order to obtain insights on their accuracy and domain of applicability.

Interestingly, although the flow stress dependence on dislocation density has been verified against experimental data, the validity of both multi-slip extensions of the Taylor law has rarely been tested in the literature. To the author's knowledge, Queyreau and co-workers [198] were the only ones to test for the validity of the FZ formulation using DDD simulations on multi-slip conditions for BCC α-iron crystals. However, the validity of the superposition principle expressed through these laws has never been discussed for HCP materials – the proposed extension (4.2) of the original FZ law (4.1) has never been discussed prior this work –, and the two FZ and LP multi-slip formulations have never been directly compared.

Therefore, the goal of this section is to assess the accuracy of the superposition principle expressed through the two common multi-slip extensions of the Taylor law, particularly when applied to the case of HCP materials. To this end, DDD simulations are performed on HCP Mg in the presence of multiple types of dislocation-dislocation interactions to compare flow stress estimates with hardening law predictions.

4.4.1 Methodology

In order to examine the validity of the superposition principle expressed through the two commonly used FZ and LP hardening laws in equations (4.2) and (4.3), respectively, latent hardening simulations with forest populations leading to multiple

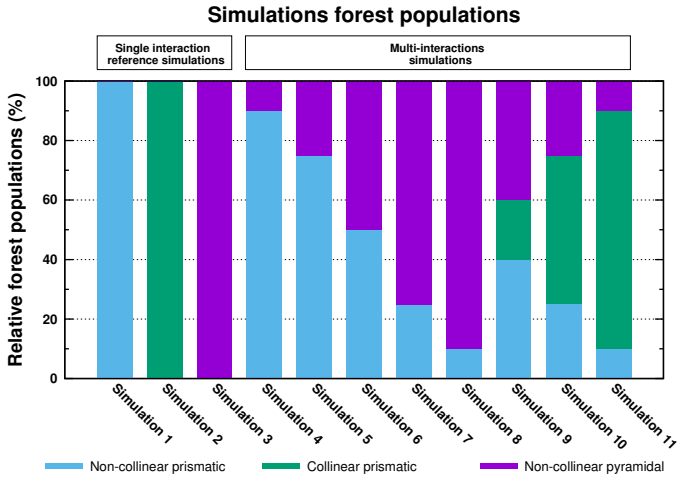


Figure 4.8: Forest populations' distribution interacting with primary basal dislocations for the different simulations.

types of interactions – as opposed to those performed in Section 4.3 whereby only one type of interaction was prescribed at a time – are performed using DDD. In doing so, the validity of the FZ and LP formulations can be assessed by comparing the primary flow stress predictions given by equations (4.2) and (4.3) using reference individual interaction coefficients computed in Section 4.3.2.2 to the flow stress predicted by DDD when multiple types of interactions are involved.

Here, the case of primary basal dislocations gliding through a forest of either prismatic, pyramidal or a combination of both dislocations types is investigated. Therefore, the coefficients computed in Section 4.3.2.2 for the (1) non-collinear basal / prismatic ($a^4_{FZ} = 0.054$; $a^4_{LP} = 0.232$), (2) collinear basal / prismatic ($a^3_{FZ} = 0.707$; $a^3_{LP} = 0.841$) and (3) non-collinear basal / pyramidal $\langle c + a \rangle$ ($a^8_{FZ} = 0.293$; $a^8_{LP} = 0.284$) interactions are used as reference individual interaction coefficients.

Multi-interactions simulations are carried out with different forest population proportions as shown in figure 4.8. The choice of such populations ensures that the forest dislocations differ not only by their slip planes but also by the magnitude of their Burgers vector.

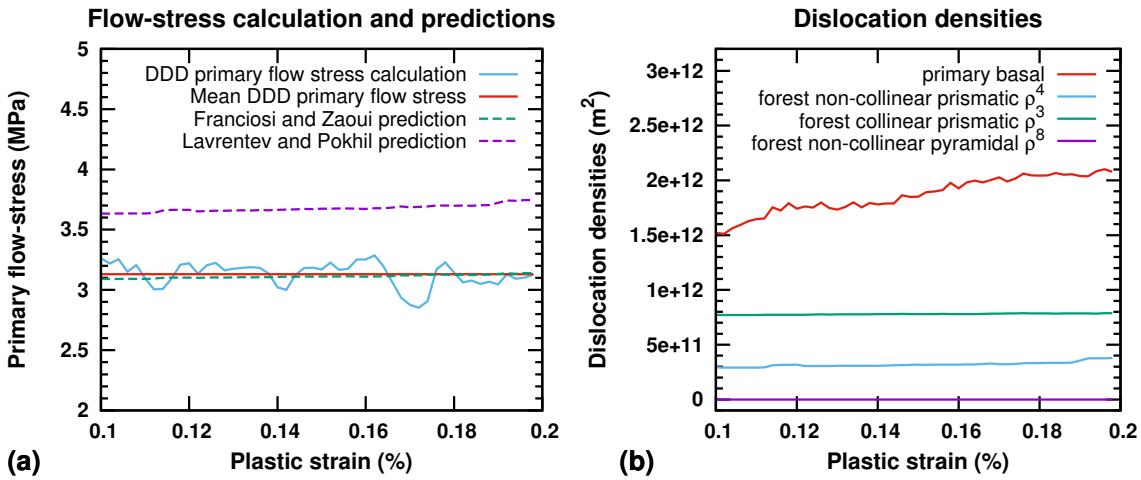


Figure 4.9: (a) Comparison of the primary flow stress and its mean on the basal system calculated by DDD and that predicted by Franciosi and Zaoui (FZ) and Lavrentev and Pokhil (LP) formulations, and (b) dislocation densities evolution computed by DDD and used in the FZ and LP predictions. The results are given for simulation 7 involving non-collinear basal $\langle a \rangle$ /prismatic $\langle a \rangle$ (25%) and basal $\langle a \rangle$ /pyramidal $\langle c+a \rangle$ (75%) interactions.

An example of the FZ and LP predictions for simulation # 7 is plotted in figure 4.9. For the sake of clarity, in that case, the FZ flow stress prediction τ_{FZ}^{basal} is computed from (4.2) as $\tau_{FZ}^{basal} = \tau_0^{basal} + \mu b_{\langle a \rangle} \sqrt{a_{FZ}^4 \rho^4 + a_{FZ}^8 \rho^8}$, where ρ^4 and ρ^8 designate the dislocation densities on non-collinear prismatic and non-collinear pyramidal $\langle c+a \rangle$ systems predicted by the DDD simulation, respectively, and where the dislocation density on the collinear prismatic system is $\rho^3 = 0$. To alleviate the numerical fluctuations in the computed flow stress induced by the discrete approach used in DDD – and hence to allow for a fair comparison –, the mean FZ and LP predictions are also compared to the mean primary DDD flow stress averaged over the plastic strain interval 0.1–0.2%.

4.4.2 Results

Relative errors for both mean FZ and LP predictions with respect to the mean DDD primary flow stress are presented for each case in figure 4.10. As reported, one can

see that the FZ model gives predictions within less than 6% of the mean flow stress value for interactions that do not involve collinear reactions. By comparison, LP formulation systematically overestimates the flow stress between 17% and 30%.

However, for multi-interactions involving collinear interactions, the agreement is poorer in both cases, with relative errors that reach 17% for the FZ model, and rise up to 80% for the LP formulation. In each case, the FZ and LP models predict higher flow stresses than that calculated by DDD. It is also interesting to note that larger error is found when a smaller fraction of collinear interactions is involved. Several factors can be considered for explaining such discrepancies. First, values of collinear interaction coefficients are greater relative to all other coefficients pertaining to non-collinear reactions (e.g. see tables 4.9 and 4.10). In that sense, significant differences in the value of the interaction coefficients are seen to lead to less accurate predictions using FZ and LP formulations. Second, flow stresses computed in the presence of collinear interactions are generally less stable (i.e. more fluctuations are observed) with strain than for other interactions. Third, and complementarily to the second point, discrepancies may be linked to the nature of the collinear interaction itself. As a matter of fact, unlike every other dislocation-dislocation interaction potentially leading to junction formation, collinear reaction leads to annihilation, i.e. junction with zero Burgers vector, inducing therefore a total reduction in the strain energy of the annihilated segments [165, 64]. In that sense, the collinear interaction can be considered as a special type of reaction, and hence may contribute to the flow stress in a different manner than other reactions.

It is nonetheless clear that the FZ formulation, for which suitability is demonstrated in the above for HCP crystals with the extension suggested in equation (4.2), should be preferred in dislocation density based constitutive models to predict CRSS, as it consistently yields closer agreement with simulated flow stress than the LP model. Note that modifying the LP formulation so as to consider the magnitude of

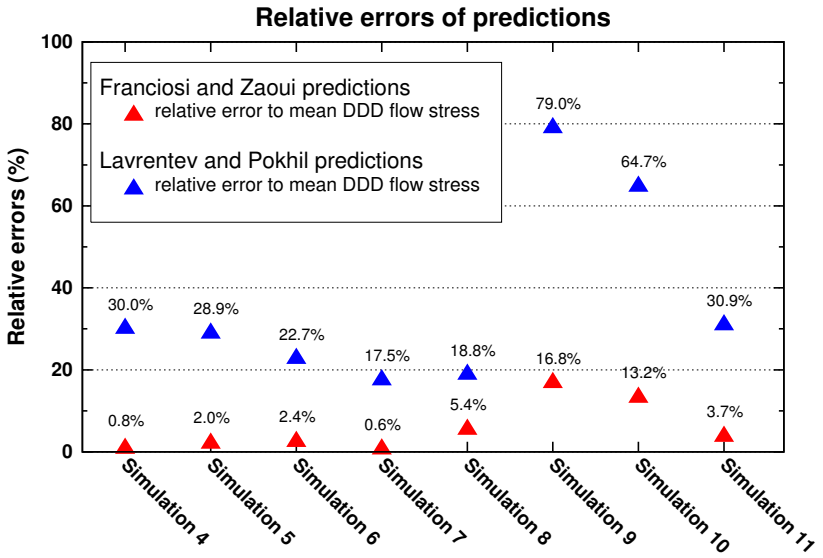


Figure 4.10: Relative errors of the mean predicted flow stresses using Franciosi and Zaoui expression (4.2) and Lavrentev and Pokhil expression (4.3) with respect to mean DDD primary flow stresses.

the primary Burgers vector in (4.3) was tested and found to result in still poorer agreements than those obtained with the FZ formulation. The reason is likely to be a fundamental one. The square root of the dislocation density is a measure of the inverse of the mean free path for mobile dislocations. However, according to equation (4.3), the LP amounts to regard the sum of individual inverses as the effective mean free path, which is physically questionable. Additionally, the fact that the sum of square roots is always larger than the square root of the sum leads to overestimating the critical stress when several interactions are simultaneously present.

Consequently, for the rest of this chapter, the discussions and results presented will be based on the interaction coefficient values obtained with the FZ formulation.

4.5 Discussion on latent hardening for Mg

4.5.1 Strength of interactions

Based on the FZ formulation approach, it is found from the results reported in table 4.10 that the interaction coefficients computed for HCP Mg can be classified into 4

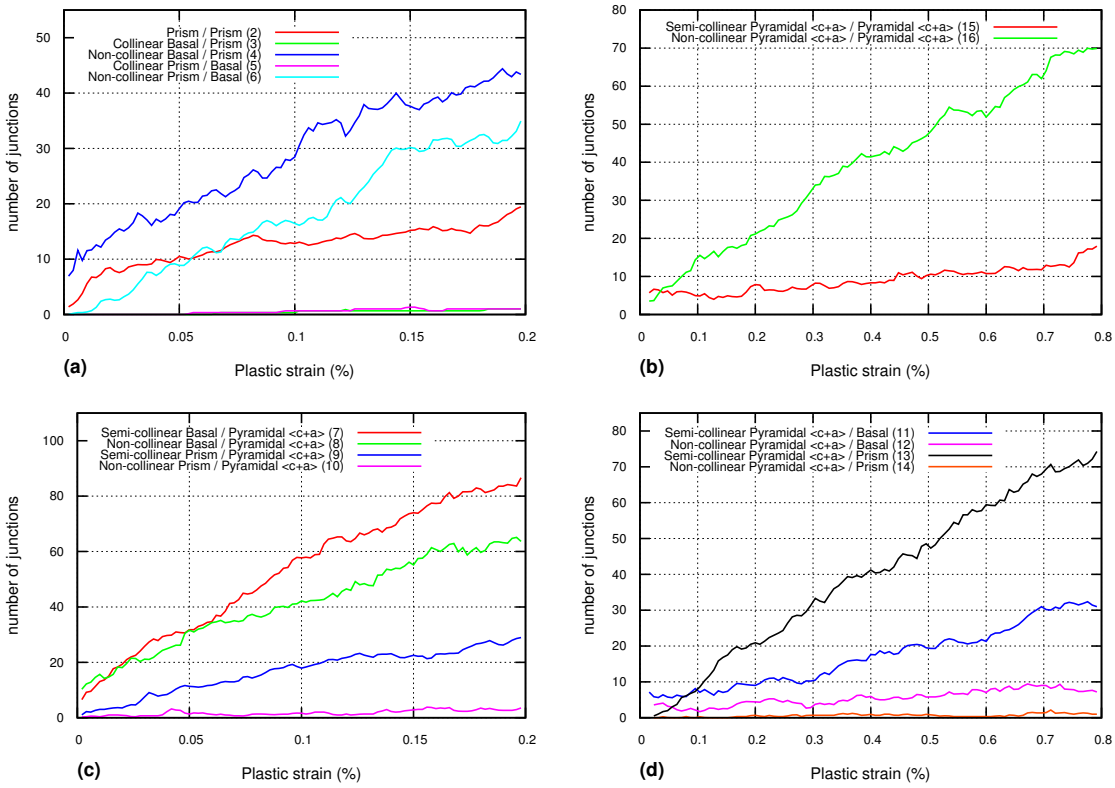


Figure 4.11: Averaged number of instantaneous junctions present during simulations using the FZ formulation for (a) $\langle a\rangle/\langle a\rangle$ interactions, (b) $\langle c+a\rangle/\langle c+a\rangle$ interactions, (c) basal/pyramidal and prismatic/pyramidal interactions and (d) pyramidal/basal and pyramidal/prismatic interactions.

strength groups. The first group is composed of the collinear interactions, i.e. those pertaining to annihilation events. As for the case of FCC materials discussed in [64], these are found to be the strongest interactions. This is because when annihilations occur, parts of the interacting segments length are removed, leaving shorter length primary dislocation lines and reducing the energy. Remobilization of these short segments requires higher stresses, since bowing out of a segment inversely scales with its length [87]. As expected, figure 4.11(a) shows that no junctions are formed in the case of collinear interactions; apart from annihilations, the only possible Burgers combination is not energetically favorable. The second group of interactions comprises the basal/pyramidal $\langle c+a\rangle$ interactions, which are found to be very stable.

Moving down in rank, the third group includes the prismatic/pyramidal $\langle c+a \rangle$ interactions, the non-collinear prismatic/basal and basal/prismatic as well as the prismatic/prismatic interactions. Except for the non-collinear pyramidal $\langle c+a \rangle$ / pyramidal $\langle c+a \rangle$ interaction which is also found in the third group, the fourth group is composed of all interactions involving primary pyramidal $\langle c+a \rangle$ dislocations. Note that interactions in the third and fourth groups are weak ($a_{FZ}^{ss'} < 0.1$) and are thus not expected to significantly contribute to strain hardening.

At this stage, it is interesting to notice that the hierarchy established for the interactions strengths in this work is consistent with that experimentally determined by Lavrentev and Pokhil [142, 141] for some slip systems interactions in single crystal Mg, and with experimental results on single crystal and polycrystalline Mg reviewed by Caceres and Lukac [34]. Although values of the coefficients obtained in these works cannot be directly compared with the ones computed in the present study because the hardening formulations used in both works are different, Lavrentev and Pokhil conclude that second order pyramidal/pyramidal interactions are weaker than the basal/prismatic interaction, and that the semi-collinear basal/pyramidal interaction is very strong, which agrees with the predictions of the present simulations.

Another interesting aspect of slip system interactions is the number of junctions that form in the different cases. Figure 4.11 shows the instantaneous number of junctions present during the course of deformation. As reported in figure 4.11(c), the two basal/pyramidal $\langle c+a \rangle$ interactions seem to lead to the most junction formations among all interactions at 0.2% plastic strain. This suggests that junctions formed in these cases are very stable and hence form strong obstacles, as already noticed by Capolungo [39] using elastic DDD. In the present study, the semi-collinear interaction is found to be stronger than the non-collinear one. Experimentally, a similar observation was reported by Lavrentev and Pokhil [142], who suggest that the smaller reduction in strain energy associated with the junction formation in the non-collinear

basal/pyramidal case is likely to be responsible for its lower strength (see table 4.3).

Interestingly, fewer instantaneous junctions are observed for the pyramidal $\langle c + a \rangle$/basal interactions, suggesting that junctions that are formed are destroyed shortly after they have combined, i.e. are weak obstacles. As a result, lower coefficient values are expected for those interactions, as less stress will be required for mobilizing primary segments across the forest. Thus, this explains the lower values of the pyramidal/basal interactions compared to those of the basal/pyramidal, and evidences the non-symmetrical property of the hardening matrix. Such a result can also be interpreted as considering that it is harder for basal $\langle a \rangle$ dislocations to overcome pyramidal $\langle c + a \rangle$ dislocations than the opposite. Following the same reasoning, figure 4.11(d) shows that almost no junctions are formed in the case of non-collinear pyramidal/prismatic interaction, explaining thereby the low value of the coefficient associated with it.

Finally, the case of prismatic/pyramidal interactions is surprising. Figure 4.11(c) clearly shows that fewer junctions are formed in the non-collinear case compared to the semi-collinear case. Nonetheless, the coefficient associated with the latter is found to be lower than the one pertaining to the non-collinear interaction. In this case, however, it is found that a large amount of 4-connected nodes are present. These correspond to crossed states – i.e. junctions with null length – and to repulsive states. Similarly to junctions, they behave as obstacles by pinning primary dislocations, and are found to induce higher hardening than junctions in the semi-collinear case. This shows that crossed and repulsive states can also significantly contribute to strain hardening when they are numerous and junctions are weak.

4.5.2 Effect of friction stress

The simulation results for cases (1) and (2), i.e. when friction stress is accounted for or not, are reported in figure 4.12. Consistently for all interactions, the friction

stress is found to lead to a decrease in the computed coefficient. This effect is consistent with the work of Monnet and Devincre [175] in which statics and dynamics dislocations simulations were employed to study the effect of lattice friction on forest hardening. From static equilibrium the authors derive that the junction length was inversely proportional to the value of the friction stress, which was verified by DDD simulations. As a result, it is found that increasing friction stress leads to decaying junction stability. In turn, this leads to a larger spacing between pinning points and thus to a lower latent hardening coefficient. A similar result was also reported in reference [198] on BCC α-iron. Correspondingly, in the present work, interactions involving primary pyramidal systems, for which dislocations have the highest lattice friction stress, exhibit the greatest sensitivity to friction. The significant influence of the friction stress also explains the very low coefficients obtained for interactions of primary pyramidal systems in case (1). Conversely, interactions involving primary basal dislocations, such as in the case of the very stable basal/pyramidal interactions, are found to be the least sensitive to lattice friction, as the friction stress of the basal system is almost negligible.

When friction stress is not accounted for, it clearly appears that non-collinear prismatic/basal interactions are inducing more hardening than non-collinear basal/prismatic interactions. This asymmetry was observed for hexagonal ice [62] for which no friction stress was considered, and was attributed to the anisotropy of line tension between basal and prismatic systems. When a junction between basal and prismatic segments is to form, the basal screw character is parallel to the junction line. Therefore, screw segments, that are easier to bow-out, are found to connect successive junction segments, thus lowering the stress required for their remobilization. However, in case (1), such asymmetry is attenuated by the presence of significant prismatic lattice friction stress. Similarly, the prismatic friction stress seems to be responsible for the lower value of the collinear prismatic/basal interaction with respect to the collinear

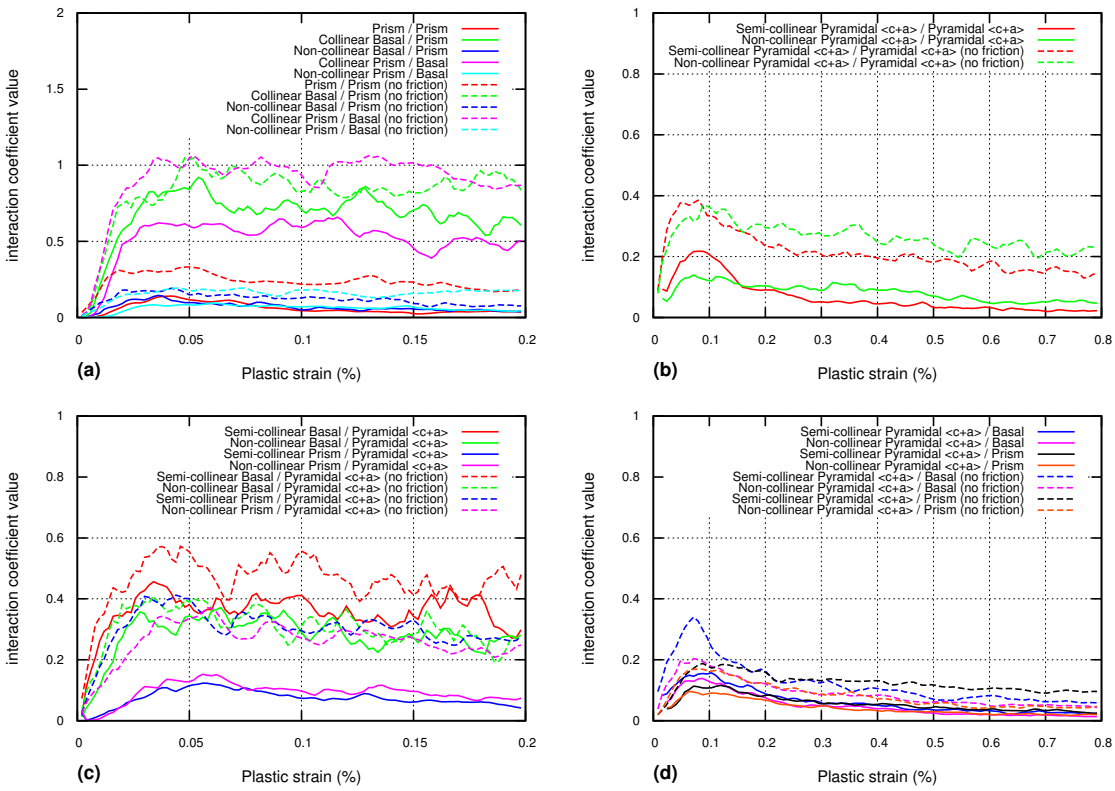


Figure 4.12: Comparison of the averaged values of the coefficients with friction stress (solid lines) and no friction stress (dashed lines) using the FZ formulation.

basal/prismatic interaction in case (1). Finally, it is interesting to notice that with no lattice friction, similarly to what is reported in reference [62], the prismatic/prismatic interaction becomes the strongest non-collinear interaction, omitting the interaction involving second order pyramidal systems, which were not considered in reference [62].

4.5.3 Effect of dislocation mobility

As discussed in reference [102], dislocations in Mg exhibit a strong anisotropy in their mobilities, depending on their slip system and their character. When studying prismatic dislocations with DDD, Monnet et al. [176] used a phenomenological mobility law in which screw dislocations were orders of magnitude slower than edge ones. Such

an approach was adopted to match experimental observations of anisotropic loop expansion favoring long screw segments and short edge segments [37]. To evaluate the effect of mobility on hardening, simulations of interactions with no friction stress and anisotropic mobilities (case (2)) given in table 4.7 are compared with two simulations where (a) edge and (b) screw mobilities are assigned to all dislocations, respectively (case (3)). Results for the non-collinear basal/prismatic interaction are reported in table 4.11.

Examining the results, one can observe that the averaged value of the interaction coefficient, when all dislocations are assigned screw mobilities, is higher than that when all dislocations are assigned edge mobilities. This result can be expected. Assigning the faster edge mobility values to all dislocations will impart more shear to the crystal than if slower screw mobilities were assigned: dislocations with higher mobilities are sweeping larger areas when gliding under a similar stress. As a result, more plastic strain is generated overall, reducing the stress and the value of the coefficient. On the contrary, if all dislocations are slower, as in the case where all mobilities are taken to be those of screw dislocations, then the stress will be higher, hence so will the interaction coefficient. Therefore, increasing the velocity of the dislocations reduces the value of the coefficient. Based on the foregoing, two issues arise. First, a lower value of the coefficient does not necessarily imply lower interaction strength, but mainly reflects a strain-rate scaling effect. Second, it suggests that more meaning needs to be placed on the hierarchy of the coefficients, i.e., relative strengths of the interactions, rather than values themselves. With this in mind, it seems difficult to conclude on the true effect of dislocation mobilities because of the coupling between the effect of (1) the average mobility value and (2) the contrast between edge and screw mobilities. In latent hardening simulations, setting such a contrast would favor the appearance of screw dislocations, assuming we assign the latter the slowest velocity. This would mainly reduce the range of possible intersection orientations

Table 4.11: Averaged values of the coefficients computed for different velocity profiles.

Simulation	Edge mobility (Pa.s)	Screw mobility (Pa.s)	Coefficient value
0: Different mobilities	$B_{edge} = 4.7 \times 10^{-6}$	$B_{screw} = 1.3 \times 10^{-5}$	0.054
a: Edge mobility	$B_{edge} = 4.7 \times 10^{-6}$	$B_{screw} = 4.7 \times 10^{-6}$	0.044
b: Screw mobility	$B_{edge} = 1.3 \times 10^{-5}$	$B_{screw} = 1.3 \times 10^{-5}$	0.059

between primary and forest dislocations as more primary screw segments are likely to interact with the forest. However, for randomly generated forest dislocations, intersection orientations would remain random, such that this effect is not expected to be significant. Similar conclusions were drawn by Wang and Beyerlein [247] when studying BCC materials with DDD, in which a high contrast between edge and screw mobilities was observed to have little influence on material response.

4.6 Effect of the latent hardening coefficients on strain hardening

As attested by the considerable development of dislocation density-based crystal plasticity models, the latent hardening coefficients defining the strength of dislocation-dislocation interactions have rapidly become a key ingredient in constitutive modelling via the use of the FZ (4.2) or LP (4.3) laws [65, 132]. Furthermore, their accurate evaluation was found to play an important role for modelling and predicting the mechanical response of materials, specifically to predict the directionality and dependence of slip activation with respect to crystal orientation [165, 112, 65] and in the case of complex loading paths such as strain path change [200, 95].

However, so far, interactions coefficients were not available for HCP Mg such that default values were used in constitutive models. Consequently, in this section, the effect of hardening coefficient values on the macroscopic response is investigated. To this end, the values of the hardening coefficients computed in this work and reported in table 4.10 are introduced into a visco-plastic-self-consistent (VPSC) polycrystal

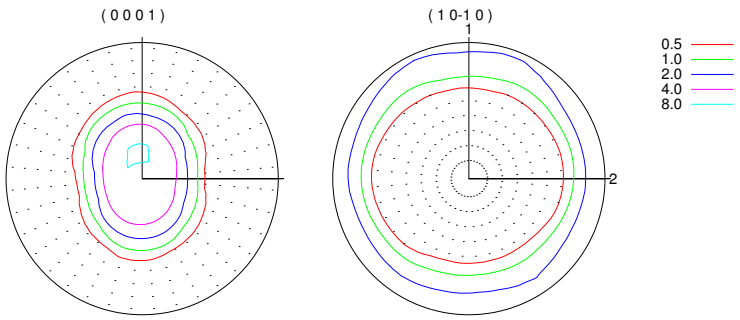


Figure 4.13: Initial rolling texture pole figure used to simulate tension in the rolling direction on polycrystalline pure HCP Mg.

model developed for HCP materials [25, 24], and tension in the rolling direction is simulated for the rolling Mg alloy texture reported by Jain et al. [117]. To represent such texture whose pole figure is plotted in figure 4.13, approximately 2000 grains are used. The loading condition is selected in order to avoid twin activation and to test solely for the effect of the hardening coefficients related to slip so that the latent effects of dislocations-twin boundary interactions need not to be considered here.

The dislocation density model proposed in references [25, 24] was initially developed to account for dislocation densities per slip mode. Here, a modified version of the model which accounts for dislocation densities on each individual slip system is used [122]. Their evolution is given by the following Kocks-Mecking type law:

$$\frac{\partial \rho^{s\in\alpha}}{\partial \gamma^s} = k_1^s \sqrt{\rho^s} - k_2^\alpha(\dot{\epsilon}, T)\rho^s$$
$$k_2^\alpha(\dot{\epsilon}, T) = k_1^\alpha \frac{\chi b^\alpha}{g^\alpha}\left(1 - \frac{kT}{D^\alpha b^3}\ln\left(\frac{\dot{\epsilon}}{\dot{\epsilon}_0}\right)\right) \tag{4.6}$$

where χ is an averaged dislocation interaction parameter and $\rho^{s\in\alpha}$ denotes the dislocation density on system s belonging to slip mode α. The CRSS on each slip system is computed using equation (4.2). The parameters of the dislocation model used here are reported in table 4.12.

In this section, four different cases are tested and compared. Each case corresponds

Table 4.12: Slip modes parameters used in VPSC simulations. For parameters details the reader is referred to reference [25].

Parameter	α = basal	α = prismatic	α = pyramidal $\langle c+a \rangle$
b^α (m)	3.231×10^{-10}	3.231×10^{-10}	6.077×10^{-10}
$\dot{\epsilon}_0$ (s^{-1})	10^7	10^7	10^7
g^α	1.55×10^{-3}	1.55×10^{-3}	1.55×10^{-3}
D^α (MPa)	21.5	21.5	21.5
A^α	0	0	0
q	0	0	0
HP^α	0	0	0

to a different set of coefficients values and is listed below:

(a) Self-hardening, corresponding to a diagonal hardening matrix, and for which the same hardening coefficient is associated to each slip system interaction.

(b) Mode-hardening, in which latent hardening is only considered between slip systems belonging to the same slip mode, and for which the same hardening coefficient is assigned for each slip mode interaction.

(c) Same-hardening, for which the same value is assigned to all hardening coefficients.

(d) Latent-hardening, corresponding to the FZ hardening coefficients computed in this work and reported in table 4.10.

The values of the hardening coefficients associated to each case are given in table 4.13. Note that case (b) corresponds to the dislocation density model proposed by Beyerlein and Tomé [25] and case (c) to the original Taylor law. In case (d), the coefficients are those computed in Section 4.3.2.2 where lattice friction is taken into account and mobilities are derived from atomistic simulations. As stated in the above, self-hardening coefficients were not computed in this work. Here, they are set to 0.150, which corresponds to the average value of the coefficients reported in table 4.10. The same value is used for the coplanar basal/ basal interaction. Also, the averaged interaction parameter χ in equation (4.6) was set to 0.150.

Table 4.13: Set of interaction coefficients values for the different cases.

Coefficient	case (a): self-hardening	case (b): mode-hardening	case (c): same-hardening	case (d): latent-hardening
S1, S2, S3	0.150	0.150	0.150	0.150
1	0	0.150	0.150	0.150
2	0	0.150	0.150	0.038
3	0	0	0.150	0.707
4	0	0	0.150	0.054
5	0	0	0.150	0.535
6	0	0	0.150	0.060
7	0	0	0.150	0.367
8	0	0	0.150	0.293
9	0	0	0.150	0.068
10	0	0	0.150	0.088
11	0	0	0.150	0.017
12	0	0	0.150	0.011
13	0	0	0.150	0.025
14	0	0	0.150	0.015
15	0	0.150	0.150	0.018
16	0	0.150	0.150	0.042

For each case, the experimental tensile stress strain responses obtained by Jain et al. [117] on 20 μm grain-sized AZ31B are fitted with VPSC. Note that the specific stress-strain data used here is not critical, but rather meant to provide a base value for comparison. For instance, one interesting macroscopic metric that can be compared is the Lankford coefficient (r-value) prediction, which provides a sensitive measure of plastic anisotropy. Given that tension is simulated along the rolling direction (RD), the r-value corresponds here to the ratio of strain in the transverse direction (TD) over strain in the normal direction (ND). Here the interest is not specifically to match the experimental data, but rather to give insights on the impact of the choice made for the interaction coefficients on such macroscopic measures. An in-depth study of the impact of dislocation interaction strength on latent hardening and mechanical response, such as in the case of single-crystals, is not within the scope of this study, but could certainly be the subject of future work.

Figure 4.14 shows the simulated stress-strain responses obtained in the four cases (a)–(d) as well as the associated Lankford coefficients (r-values) predictions. The

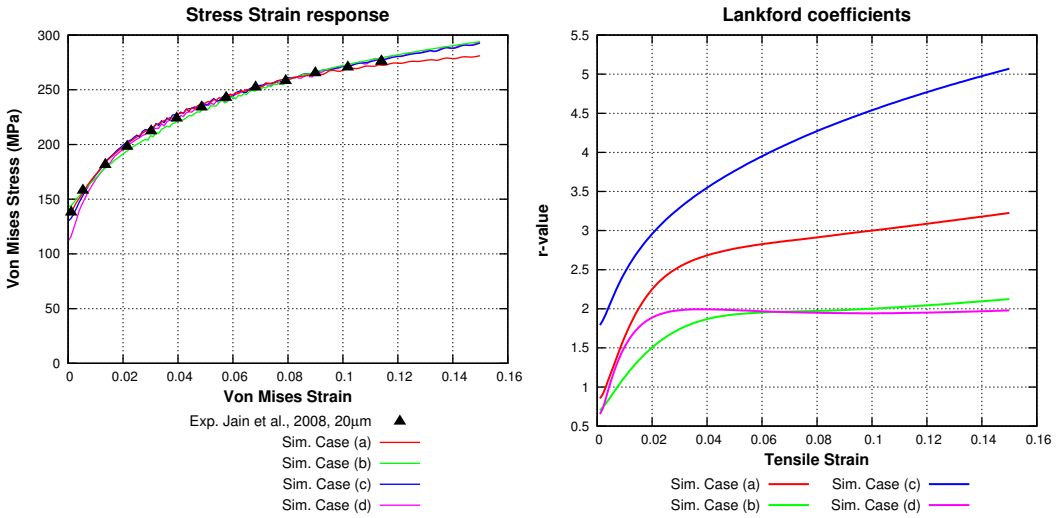


Figure 4.14: Polycrystal stress strain responses and Lankford coefficients (TD/ND strain ratio) predicted by VPSC for RD tension in the case of (a) self-hardening, (b) mode-hardening, (c) same-hardening and (d) latent-hardening.

dislocation density hardening law parameters used to reproduce the experimental stress-strain response in each case are reported in table 4.14.

Several observations are noteworthy. First, notice that appropriate fits of experimental stress–strain data can be achieved using any of the different sets of coefficients. However, internal state variables of the model associated to the different cases do not necessarily evolve in a similar fashion. Interestingly, the r-values predicted using the coefficients computed in this work (case (d)) are close to those predicted in mode-hardening case (b), and are in excellent agreement with the experimental values of around $r = 2$ reported in reference [117]. However, it also appears that the predictions of r-values can be significantly affected by the set of coefficients used, as demonstrated in the same-hardening case (c), for which the Lankford coefficient reaches a value of 5 at 14% strain.

Furthermore, as shown in figure 4.15, the use of different interaction coefficient values induces differences in the predicted slip modes activities. While mode-hardening (b) and latent-hardening (d) cases predict similar r-values, they nonetheless exhibit different dislocations densities evolutions. Such differences are most noticeable in

Table 4.14: Dislocation density based hardening parameters used to fit experimental data for the different cases.

Case	Slip mode α	k_1^α (m^{-1})	τ_0^α (MPa)
case (a): self-hardening	α = basal	3.5×10^9	20
	α = prismatic	1.0×10^9	65
	α = pyramidal $\langle c+a \rangle$	4.0×10^9	105
case (b): mode-hardening	α = basal	8.5×10^8	20
	α = prismatic	7.0×10^8	70
	α = pyramidal $\langle c+a \rangle$	7.0×10^8	110
case (c): same-hardening	α = basal	2.8×10^9	25
	α = prismatic	1.6×10^8	58
	α = pyramidal $\langle c+a \rangle$	1.0×10^8	85
case (d): latent-hardening	α = basal	3.0×10^9	15
	α = prismatic	6.0×10^8	55
	α = pyramidal $\langle c+a \rangle$	6.0×10^8	80

the case of same-hardening (c) – corresponding to the classical Taylor-law –, as attested by the lack of pyramidal activity and the strong predominance of prismatic dislocations compared to the other cases.

4.7 Conclusion

In this chapter, the regular DDD code developed during this thesis and presented in Chapter 2 was employed to compute the strength-interaction matrix for pure hexagonal Mg for the first time. In addition to providing a precise quantification of dislocation-dislocation interaction coefficients in HCP Mg, this work revealed several interesting features.

First, similar to FCC materials, it was found that collinear interactions potentially contribute the most to latent hardening. Via annihilations of portions of segments, interactions between collinear systems leave segments of shorter length that require an increase in the flow stress to be remobilized, hence yielding a strong hardening. Besides, it was shown that basal/pyramidal $\langle c+a \rangle$ interactions are very stable, leading to significant hardening, while interactions involving primary second order pyramidal $\langle c+a \rangle$ dislocations were found to be the weakest ones. Further, it was shown

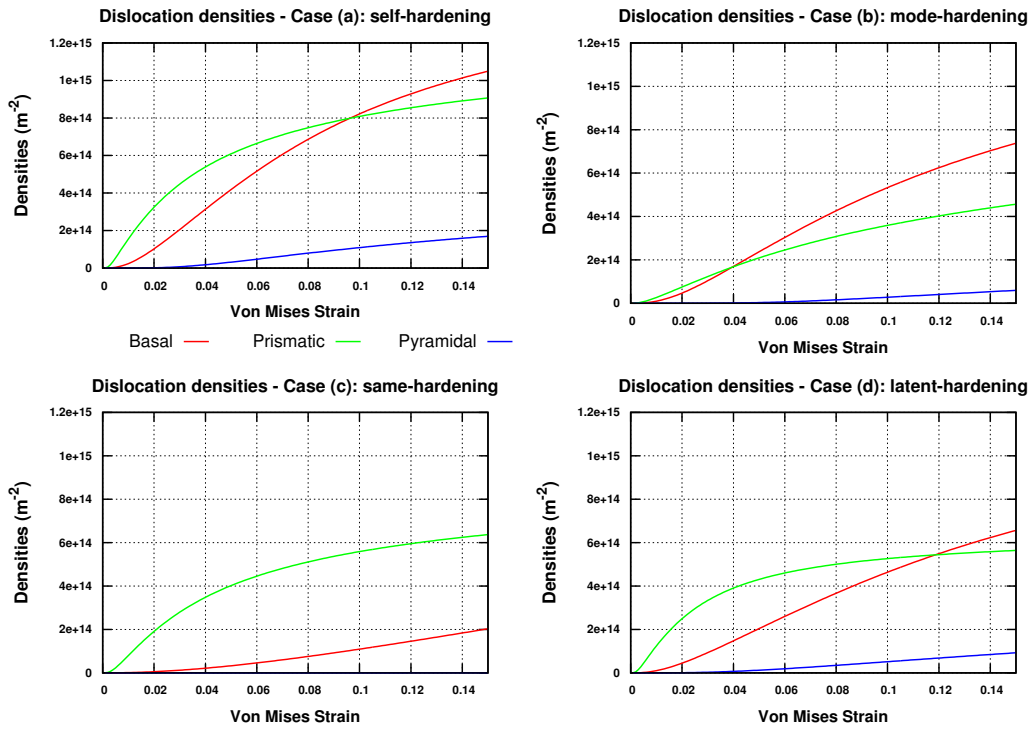


Figure 4.15: Evolution of the dislocation densities for each slip mode as a function of strain for tension in the rolling direction (RD) for the different cases.

that crossed and repulsive states could significantly contribute to strain hardening when junctions are weak as for the prismatic/pyramidal interaction. Also, the large contrast between intrinsic properties of the different HCP slip systems (mobilities, friction stress) was shown to induce non-symmetrical latent hardening. Furthermore, the effect of lattice friction has been studied, and, consistent with previous studies, it was found that lattice friction tends to lower the forest hardening effect. Nonetheless, it was observed that while the dislocation mobility is seen to have an impact on the interaction coefficient value by virtue of strain-rate sensitivity, the contrast between edge and screw dislocation mobilities was found hard to quantify, and an in-depth investigation would require the incorporation of more sophisticated mobility functions.

In addition, this work revealed that Franciosi and Zaoui [90] formulation, whose

extension suggested in equation (4.2) is demonstrated to be applicable to HCP crystals, gives accurate predictions when multiple interaction types are involved in the deformation process, whereas the formulation suggested by Lavrentev and Pokhil [141] consistently overestimates the flow stress predictions. As a result, it is suggested that the FZ law (4.2) should be preferred to the LP formulation (4.3) in dislocation density-based constitutive modelling.

Finally, in studying latent hardening effects on polycrystal response, it was shown that an accurate quantification of the interaction coefficients seemed to have an important impact on predicting the macroscopic plastic anisotropy. Specifically, it was demonstrated that although any experimental stress-strain response could be reproduced with any set of interaction coefficient parameters, differences in their values can lead to significantly different predictions in terms of dislocation densities and slip systems activities.

CHAPTER V

DISLOCATION-PHASES INTERACTIONS

5.1 Dislocation-twin interactions

5.1.1 Motivation

The mechanical behavior of polycrystalline metallic materials is strongly affected by the interaction between moving dislocations and grain boundaries and interfaces [245, 222]. Of particular interest, the presence of twins has been reported to have a significant impact on the mechanical response of nanostructures FCC crystals [273, 275, 274]. Specifically, particular attention has been recently directed towards nano-twinned FCC materials, exhibiting significant improvements in terms of strength and ductility, unveiling promising processes to fabricate materials with enhanced mechanical properties [159, 158, 160].

Experimentally, the substantial work-hardening obtained has been primarily attributed to the accumulation of dislocations at the twin boundaries (TB) [47, 159, 158, 149, 160], and twinning deformation in nano-crystalline materials has been found to generate a greater plastic activity than in their coarse-grained counterparts [138, 150, 233, 261]. Presumably, slip-twin interactions can be regarded as inducing two correlated effects: while dislocation motion is impeded by TBs because of the crystallographic reorientation, twin growth is conversely expected to be hindered by interactions between moving TBs and slip bands. In this section, the focus is placed on the interaction of moving dislocations with fixed TB interfaces.

Overall, the prominent role of dislocation interactions with TB is expected to yield two major correlated effects. First, upon accumulation, the pile-ups of dislocations at the boundary are expected to generate large stress concentrations. Second,

these high stress concentrations are expected to create favorable conditions for dislocation reactions to occur. Depending on the local conditions, the crystallography, and the stacking-fault energies, dislocation-twin reactions can either lead to absorption, formation of locks or transmission of dislocations. In certain circumstances, the transmission of dislocations across the twin boundary via dissociation of their Burgers vectors may lead to the generation of partial twinning dislocations mobile on the interface. In that sense, it is apparent that dislocation-twin boundary interactions are more complex than producing a simple Hall-Petch effect by reducing the effective mean free path of dislocations in the nano-grains.

From a modelling standpoint, atomistic simulations studies on FCC crystals have confirmed the prominent role of dislocation transmission across coherent twin boundaries (CTB) on the strengthening observed experimentally [119, 118, 59]. However, due to limitations in terms of time and length scales, MD simulations solely permit to gain insights on unit processes, and are therefore not able to provide an extensive comprehension of the collective effect of dislocation interactions with CTB. Furthermore, very few DDD studies have been dedicated to investigate such mechanism and those relied on the use of simple line tension models [57, 272]. Therefore, by lack of information, the role of dislocation-twin has rarely been incorporated into constitutive laws [120, 208, 25]. However, note that very recently, Fan et al. studied dislocation-CTB interactions in polycrystalline HCP Mg by incorporating transmission rules in their DDD framework [82]. However, while their work primarily focused on the resulting macroscopic hardening response, dislocation multiplication and interfacial plasticity was not studied in details.

Consequently, dislocation transmission across a coherent twin boundary in FCC materials using DDD is proposed to be investigated in this thesis so as to assess the role of dislocation-CTB transmission and its effects on interfacial and mesoscale plasticity. For such purpose, the DDD code developed in this work is further extended

to incorporate details on individual transmission events based on atomistic simulations and predictive models resulting from experimental observations. It is expected that a fine study at this scale would provide relevant information on the transmission mechanism that could be incorporated into crystal plasticity models to refine current transmutation-induced hardening approaches [78, 189].

The present section is organized as follows. First, current modelling approaches dedicated to the role of dislocation-twin interactions and their results are reviewed in Section 5.1.2. Then, the methodology used to model dislocation-CTB interactions in DDD and the associated numerical implementation are presented in Section 5.1.3. Finally, results of the present study are discussed in Section 5.1.4 while new perspectives on the effects of dislocation-twin interactions are presented in Section 5.1.5.

5.1.2 Earlier studies of dislocation-twin boundary interactions

5.1.2.1 Predictive models

The first studies dedicated to the investigation of individual interactions between dislocations and boundaries – of which coherent twin boundary (CTB) is a special case – focused on developing transmission rules to predict the activation of slip in the neighboring grain due to a dislocation pile-up the grain boundary (GB). Following the illustration given in figure 5.1, Livingston and Chalmers [156] first suggested a geometrical criterion, referred to as the N-criterion, in which the activated slip system i was proposed to be that maximizing the N value defined as

$$N = (\vec{\boldsymbol{n}}_p \cdot \vec{\boldsymbol{m}}_i)(\vec{\boldsymbol{n}}_p \cdot \vec{\boldsymbol{m}}_i) + (\vec{\boldsymbol{n}}_p \cdot \vec{\boldsymbol{m}}_i)(\vec{\boldsymbol{m}}_p \cdot \vec{\boldsymbol{n}}_i) \tag{5.1}$$

where $\vec{\boldsymbol{n}}_p$ and $\vec{\boldsymbol{m}}_p$ denote the unit normal and slip direction associated with the incoming slip system in which the pile-up occurs, and $\vec{\boldsymbol{n}}_i$ and $\vec{\boldsymbol{m}}_i$ that of the potential transmitted slip system i in the neighbor grain. The principal shortcoming of this model is that it does not account for the orientation of the grain boundary plane,

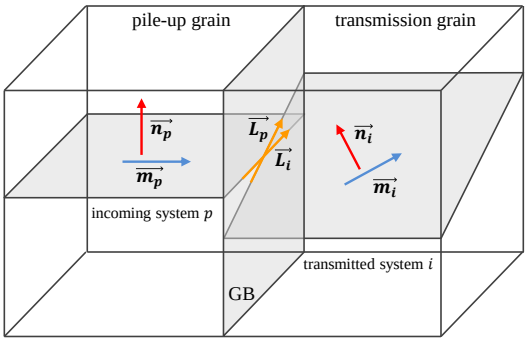


Figure 5.1: Schematic of the notations used for the N and M predictive criteria in equations (5.1) and (5.2), respectively. Reproduced from [222].

but solely accounts for the misorientation between both the incoming and the transmitted system. This was later incorporated in the approach developed by Shen and co-workers [214, 215]. In this approach, a combined geometrical and stress-based criterion was proposed, in which the N criterion in equation (5.1) was replaced by the M criterion to account for the GB orientation as:

$$M = (\vec{\boldsymbol{L}}_p \cdot \vec{\boldsymbol{L}}_i)(\vec{\boldsymbol{b}}_p \cdot \vec{\boldsymbol{b}}_i) \tag{5.2}$$

where $\vec{\boldsymbol{L}}_p$, respectively $\vec{\boldsymbol{L}}_i$, denotes the line intersection direction between the incoming p plane, respectively the transmitted plane i, and the grain boundary plane, as depicted in figure 5.1. In this approach, the geometrical M criterion in equation (5.2) to select the most favorable slip plane $\vec{\boldsymbol{n}}_i$ was further complemented by a stress-based criterion to determine the slip direction $\vec{\boldsymbol{m}}_i$, so as to ensure that the resolved shear stress (RSS) on the transmitted slip system is maximum.

Although devised from experimental observations, in both these seminal approaches, the activation of slip systems in the neighboring grain can be regarded as resulting from stress concentrations generated by the dislocation pile-up at the grain boundary. In other words, the accumulation of dislocation lines on one side of the GB triggers

the activation of distinct dislocations lines on the other side of the GB through elastic effects. As such, the transmission of dislocations across the GB, which has been observed experimentally, was not readily accounted for. In studying Ni bicrystals, Lim and Raj [151] further underlined the important role of residual dislocation arrays in the GB upon absorption and re-emission of dislocations. As a result, and to address the limitations of the model of Shen et al. [214, 215], Lee and co-workers [145] suggested a new predictive approach, in which (1) the activated transmitted slip plane is determined so as to maximize $(\vec{\boldsymbol{L}}_p \cdot \vec{\boldsymbol{L}}_i)$, i.e. minimize the incoming and transmitted planes misorientation, and where (2) the activated slip direction is determined upon maximization of the RSS and minimization of the residual Burgers. This criterion was further refined to ensure that the RSS on the selected slip system is larger than the critical resolved shear stress (CRSS) on that system, thereby allowing for dislocation motion on the transmitted plane [170]. Via in situ TEM studies, the approach of Lee et al. [145] was demonstrated to provide good predictive capabilities for GB transmission due to pile-ups, especially by removing the inconsistencies of earlier models. Further, it is interesting to notice that this criterion can be readily used for dislocation transmission across twin boundaries, as no specific description of the structure of the interface is involved.

5.1.2.2 Atomistic simulations

Several investigations have been performed using Molecular Dynamics (MD) simulations to study the interaction between dislocation and twin-boundaries (TB) for different crystalline structures. Such studies have provided a better understanding of the absorption and transmission mechanisms, and have allowed to speculate about possible factors underlying the occurrence of the different reactions.

Regarding FCC crystals, Jin and co-workers investigated the slip transmission of screw [119] and 60° [118] dislocations across coherent twin-boundaries (CTB) in Al,

Cu and Ni. In the case of screw dislocations, it was found that recombination of Shockley partial dislocations at the CTB was first necessary before interactions to occur, further leading to either absorption and dissociation into partials spreading along the CTB in Al, or transmission into the twin by cutting through the CTB in Cu. For this last case, the constriction of partials at the CTB is required for the leading and trailing partials to exchange order before propagation in the twin [119]. In the case of a 60° dislocation, the resulting interaction processes can lead to several outcomes depending on the sign of the dislocation, the amount of strain, and the material. In all cases, and similarly to the screw dislocation, the two Shockley partials first recombine to form a perfect $\frac{1}{2}[101]$ incoming dislocation at the CTB. Then, the different interactions consist in various dissociations of the 60° $\frac{1}{2}[101]$ dislocation. Thus, in Cu, the dislocation either decomposes into (1) another perfect dislocation transmitted in the twin plus a twinning dislocation lying in the CTB, or into (2) a transmitted Shockley partial and a remaining defect forming a sessile "i-lock", depending on its sign. In Al, the interaction mechanism either proceeds in a similar way than case (1) for Cu at high strains, leads to the decomposition into a partial in the CTB and a defect forming a sessile Frank lock at lower strains, or produces two repulsive twinning partials along the CTB [118]. In addition, following their results, Jin et al. suggested that a novel criterion for slip transmission should preferably rely on energetic considerations rather than on geometric and stress-based approaches, and postulated that the different types of dissociations accompanying transmission are related to the material-dependent energy barriers to form partial dislocations [118]. While these conclusions were later supported by the work of Chassagne et al. [43] in which the same interaction mechanisms were studied for 10 different inter-atomic potentials, it was further highlighted that transmission events require dislocations to be forced into the CTB by ways of pile-ups.

5.1.2.3 Dislocation dynamics simulations

Very few DDD studies have been dedicated to the investigation of dislocation interactions with interfaces. To the author's knowledge, Zhou and co-workers were the first to employ DDD simulations accounting for dislocation-boundary interactions to study the size-dependent plasticity in polycrystalline thin-films [272]. In their simulations, the line tension model developed by de Koning et al. [57, 56] from atomistic simulations results was implemented into the DDD framework to account for transmission across GB. As a result, the authors were able to describe the resulting stress-strain response with a Hall-Petch relation, attesting for the plasticity dependence to the grain size. However, for the sake of simplicity, the line tension model in their work was limited to model transmission events that do not produce residual dislocations at the boundary. Therefore, such approach would not be appropriate to study in details the dislocation-twin interactions reported in the previous sections. Alternatively, an explicit treatment of dislocation-TB interactions should be incorporated in order to account for the different possible dissociations occurring at the interface, thereby allowing for gaining insights on their collective effects.

Nonetheless, it is to be noticed that very recently, the effect of CTB interactions on the macroscopic response have been investigated in HCP Mg using DDD simulations in which a transmission mechanism involving dissociation of dislocations was explicitly incorporated into the DDD framework [82]. The study revealed the complex role of CTB reactions inducing a competition between a hardening effect resulting from the hindered motion of dislocations and a softening effect arising from twin growth. However, while the focus was placed on the overall mechanical response, the precise role of dislocation-CTB reactions and their effects were not investigated in details.

5.1.3 Methodology and numerical implementation

Here, the collective effect of interactions between dislocations and CTB and their role on the multiplication of dislocations at the interface are studied using the DDD simulation tool developed in Chapter 2, in which an explicit transmission mechanism has been implemented. For the sake of simplicity, and to depart from the work of Fan et al. on HCP Mg [82], the present investigation is carried out on FCC Cu crystals.

5.1.3.1 Geometry of dislocation-twin intersection in FCC crystals

The crystallography of slip in FCC materials is conveniently represented using the Thompson's tetrahedron shown in figure 5.2(a), in which perfect $\frac{1}{2}\langle\bar{1}10\rangle(111)$ dislocations and $\frac{1}{6}\langle\bar{2}11\rangle(111)$ Shockley partials are delineated.

In FFC crystals, twinning occur by glide of $\frac{1}{6}\langle\bar{2}11\rangle$ twinning dislocations on successive $\{111\}$ planes. The type I twinned lattice of a FCC crystal can be regarded as a mirror reflection of the parent lattice across the $\{111\}$ twin plane, i.e. as a 180° rotation around the normal to the twin plane. A simple way to illustrate parent-twin orientation relationship is to consider the double Thompson's tetrahedron drawn in figure 5.2(b). In this simple case, the atomic structure of the $\{111\}$ interface is coherent; hence it is referred to as a coherent twin-boundary (CTB).

The different possible dissociation reactions that can occur upon interaction of a dislocation and a $\{111\}$ CTB in FCC crystals have been systematically identified in [275]. As reported in MD simulations [119, 118], leading and trailing partials recombine in each case upon interaction. Therefore, the interaction of a perfect $\frac{1}{2}\langle101\rangle$ dislocation with a $\{111\}$ CTB is investigated in this work.

In the following, the intersection between a (111) twin boundary and dislocations on the $(\bar{1}11)$ plane is considered. As illustrated by the tetrahedron representation in figure 5.2, perfect dislocations on the $(\bar{1}11)$ plane have either screw of 60° characters upon intersection with the (111) plane. The case of the $\frac{1}{2}[0\bar{1}1]$ screw dislocation does

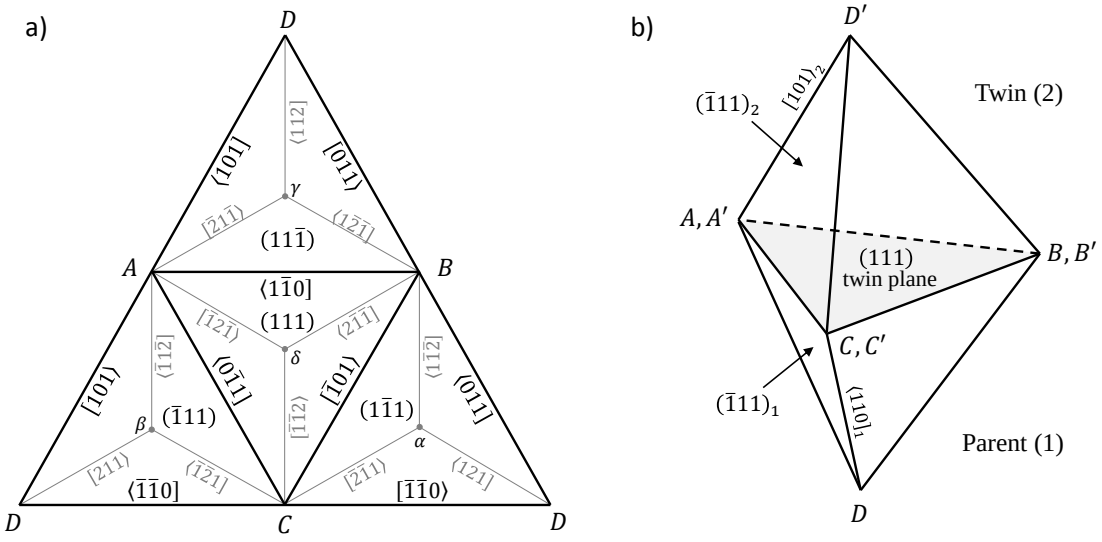


Figure 5.2: (a) Thompson's tetrahedron [229] centered in the (111) plane. The crystallographic relative orientations between the four {111} slip planes is recovered when the three external sides of the tetrahedron are folded such that points D at each corner meet in a single position. Perfect $\frac{1}{2}\langle\bar{1}10\rangle$ and Schokley partial $\frac{1}{6}\langle\bar{2}11\rangle$ dislocations are represented on each of their slip planes. The notation $[101\rangle$, for instance, instead of [101], is used to indicate the direction of the Burgers vector. (b) Double Thompson's tetrahedron to represent parent–(111) twin orientation relationship.

not present any particular interest for this study since it has been shown in MD simulations that such dislocation does not dissociate but readily cuts through the CTB by cross-slip in Cu [119]. The two remaining 60° $\frac{1}{2}[101]$ and $\frac{1}{2}[110]$ dislocations exhibit symmetric properties with respect to the interface and are therefore expected to behave similarly upon intersection. The different possible reactions that can occur upon intersection are extensively discussed in [275].

5.1.3.2 Simulation settings

To study the transmission mechanism across a CTB and its effects, the setting showed in figure 5.3 is used. Initially, a single $\frac{1}{2}[110]_1(\bar{1}11)_1$ straight dislocation source (green dashed line) is introduced in the $A6$ (using Schmid and Boas notation) slip system of the parent crystal (1) of a nano-twinned FCC Copper bi-crystal with (111) CTB. In the rest of this section, subscripts 1 and 2 associated with the planes and directions

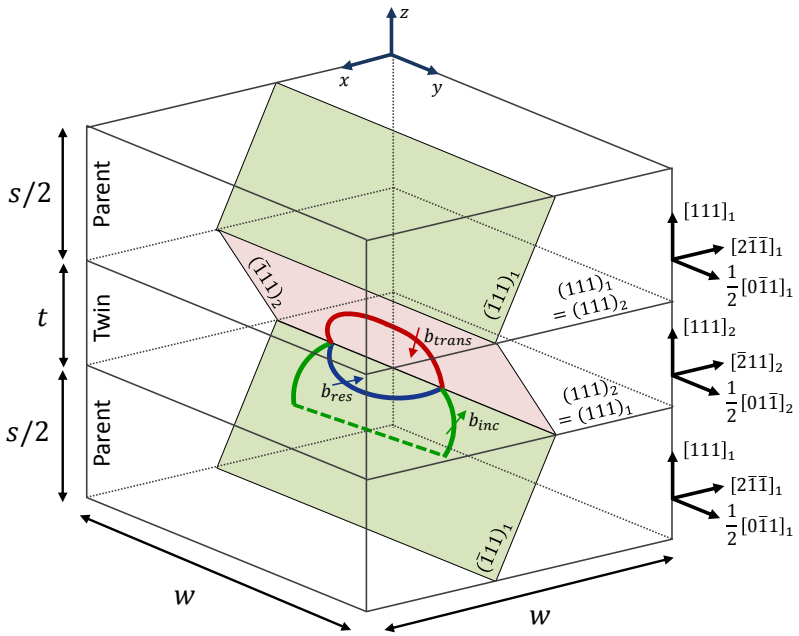


Figure 5.3: Bicrystal simulation setting: a type I twin with (111) twinning plane and volume fraction $f = t/(s+t)$ is introduced within a FCC Copper crystal. Initially, a straight dislocation source (green dashed line) with Burgers vector $\vec{\boldsymbol{b}}_{inc} = \frac{1}{2}[110]_1$ is introduced on the $(\bar{1}11)_1$ plane of the parent phase (1). Upon transmission (see Section 5.1.4) the incoming dislocation dissociates into a red transmitted dislocation with $\vec{\boldsymbol{b}}_{trans} = \frac{1}{2}[101]_2$ propagating on the $(\bar{1}11)_2$ plane of the twin and a residual twinning dislocation $\vec{\boldsymbol{b}}_{res} = \frac{1}{6}[\bar{2}11]_2$ gliding on the (111) interface.

indicate the lattice on which they are expressed, i.e. (1) corresponding to the parent phase and (2) to the twinned lattice. In this setting, the normal to the $(111)_1 = (111)_2$ interface is aligned with the $\vec{\boldsymbol{z}}$ axis of the global frame, and $\vec{\boldsymbol{x}}$ and $\vec{\boldsymbol{y}}$ axes correspond to the $[\bar{2}11]_1$ and $[0\bar{1}1]_1$ directions of the parent lattice (1), respectively. The twinned crystal (2) corresponds to a rotation of 180° of the parent lattice (1) around the $\vec{\boldsymbol{z}}$ axis such that the $[2\bar{1}\bar{1}]_2$ and $[01\bar{1}]_2$ are aligned with the $\vec{\boldsymbol{x}}$ and $\vec{\boldsymbol{y}}$ axes, respectively. The total height of the parent crystal is given by s and that of the twin by t, such that the volume fraction f occupied by the twin is expressed as $f = t/(s+t)$.

Upon transmission, the incoming dislocation line has a 60° character and its core is considered to be constricted. As mentioned earlier, this assumption is consistent with MD simulations in which the leading and trailing partials are seen to recombine

in a perfect dislocation at the CTB [118], and allows for circumventing the difficulties associated with the treatment of partial dislocations in DDD simulations.

For the sake of simplicity, the crystal is assumed to be elastically isotropic so that the effects of elastic anisotropy are neglected when transmission occurs. Under such assumption, the twin solely corresponds to a rotation of the crystallographic structure in which no elastic mismatch is generated. As a result, the regular DDD code developed in Chapter 2 will be employed using the non-singular isotropic formulation.

In the simulations performed in this work, a constant stress $\boldsymbol{\sigma}^{ext}$ is applied to the simulation volume, such that:

$$\boldsymbol{\sigma}^{ext} = \begin{pmatrix} 0 & \sigma_{xy} & 0 \\ \sigma_{xy} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{5.3}$$

The shape of the applied stress tensor in (5.3) is chosen such that the Schmid factor m is large on the incoming $A6$ slip system ($m = -0.33$) and vanishes on the (111) twin plane ($m = 0$). To investigate the transmission mechanism under different situations, different values for the magnitude of the stress σ_{xy} are chosen.

Simulations are further performed for several twin thicknesses t: $0a$ (no twin), $100a$, $500a$ and $1000a$, where $a = 0.3634nm$ is the lattice spacing in FCC Cu. Note that periodic conditions are used and that the size of the primary simulation volume $V = (s + t)w^2$, where $w = 5000a$ is the length of the simulation volume along the x and y axis (see figure 5.3), is held constant for all simulations. Therefore, any variation in the twin thickness t equivalently corresponds to a variation in the twin fraction f.

The material parameters used to model isotropic FCC Copper are reported in table 5.1.

Table 5.1: Material parameters used to model isotropic FCC Copper.

Material parameter	Value
Shear modulus	$\mu = 45$ GPa
Poisson's ratio	$\nu = 0.33$
Friction stress	0 MPa
Drag coefficient	10^{-4} Pa.s
Lattice parameters	$a = 3.634 \times 10^{-10}$ m

5.1.3.3 Numerical implementation

The principal difficulty in studying dislocation-twin interactions in DDD simulations resides in the development and implementation of local interfacial rules to model the dissociation and transmission mechanisms across the twin boundary. As illustrated in figure 5.3, the twinned crystal is considered as a specific layer in which the crystallographic orientation differs from that of the parent phase, and the interface separating both phases is assumed to be perfectly coherent. Thus, a dislocation initially gliding in slip system s_{inc} in the parent grain can generally not continue gliding along the same plane and direction as it crosses the interface, simply because its plane and direction do not necessarily exist as such in the twinned rotated crystal.

The local rules to model transmission across a CTB that have been implemented directly follow from the prediction framework suggested by Lee et al. [145] presented in Section 5.1.2 and involve geometrical and energetic considerations. The numerical process implemented to treat dislocation interaction with CTBs is as follows. After nodal velocities are calculated in the DDD code (see Section 2.5), dislocation nodes that cross the interface upon motion are first stopped at the interface along the direction of their velocity to attain the configuration depicted in figure 5.4(a). Then, the criteria proposed by Lee et al. [145] are used to determine the most favorable slip system s_{trans} for the portion of dislocation to be transmitted. Thus, (1) the transmitted slip plane with normal $\vec{\boldsymbol{n}}_{trans}$ is selected so as to minimize the misorientation $\theta = \arccos(\vec{\boldsymbol{n}}_{inc} \cdot \vec{\boldsymbol{n}}_{trans})$ between the incoming and transmitted planes, while (2) the

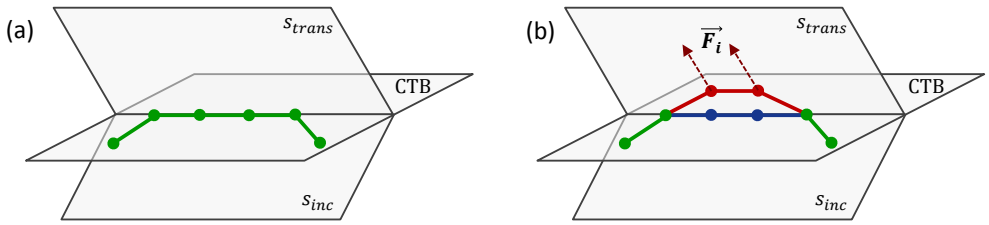


Figure 5.4: Schematic of the numerical treatment of dislocation transmission across a coherent twin boundary (CTB). (a) When crossing the boundary, dislocation nodes in the incoming slip system s_{inc} are first stopped at the interface. (b) If a favorable slip system s_{trans} is found, a trial transmitted configuration is formed by moving interfacial nodes on the transmitted system (red nodes) while inserting residual nodes (blue nodes), if required. Then, forces on transmitted nodes are computed to ensure glide will take place in the transmitted grain. If not favorable, the configuration is set back to its initial setting in (a).

transmitted direction $\vec{\boldsymbol{b}}_{trans}$ is chosen so as to minimize the norm of the residual Burgers vector $\vec{\boldsymbol{b}}_{res} = \vec{\boldsymbol{b}}_{inc} - \vec{\boldsymbol{b}}_{trans}$ that is required to conserve Burgers continuity upon transmission involving a change in slip direction.

If a favorable slip system s_{trans} for transmission is found, a trial transmitted configuration is created by dissociation of the initial dislocation portion stopped at the boundary: interfacial nodes are moved onto the transmitted slip system at a small distance from the interface and a supplementary residual dislocation line with Burgers $\vec{\boldsymbol{b}}_{res}$ is inserted along the interface, if $\|\vec{\boldsymbol{b}}_{res}\| \neq 0$, as illustrated in figure 5.4(b). Then, forces on transmitted dislocation nodes are computed to ensure that the new configuration is energetically favorable and will not drive the transmitted line back to the interface so as to recombine with the residual dislocation and retrieve its undissociated state. If forces on the transmitted line tend to drive the dislocation away from the interface, the transmission is considered as energetically favorable, and the trial transmitted configuration is conserved. However, if forces on the transmitted line act to drive the line towards the interface, the configuration is set back to its initial undissociated state and the dislocation is stopped at the interface. In that case, the transmission process will be reconsidered at the next time step.

In this process, the force calculation on transmitted dislocation lines ensures that dislocations are transmitted solely when local stress conditions allow for such events. Here, the stress includes both the contributions from the microstructure, i.e. from the other dislocation lines present in the volume, and the applied loading. Thus, when the internal stresses and the applied loading are not sufficient for transmission to occur, long untransmitted dislocation portions typically spread along the interface (this will be later shown in figure 5.6(a)). A simple approach to treat transmission would consist in testing for the simultaneous transmission of all consecutive interfacial dislocation nodes so as to determine whether all of them are transmitted or none of them are. However, when interfacial dislocations become long, local transmission conditions might be met solely along a portion of $m \leq n$ nodes of the initial n consecutive interfacial nodes, such that treating all dislocation nodes as a whole is not satisfactory. For this reason, a recursive algorithm has been developed to sequentially test for the transmission of $n, n-1, n-2, ...$ nodes of the interfacial dislocation, such as to ensure that a portion with $m \leq n$ dislocation nodes can be transmitted if conditions are satisfied locally (e.g. see figure 5.6(b)).

Further, it is to be noticed that in addition to the crystallographic reorientation of the twin with respect to the parent, the rotation of the twin crystal may induce coherency stresses due to the elastic mismatch between both phases. In the case of elastic isotropy, the rotation of the twin crystal does not induce any elastic mismatch, such that forces on both sides of the interface can be computed using the non-singular isotropic formulation developed by Cai et al. [35] implemented in the regular DDD code developed in Chapter 2. Nonetheless, elastic mismatch will arise in general for anisotropic crystals. In this case, the isotropic stress formulation cannot be used and computationally costly bicrystal formulations must be employed [107, 2]. To alleviate this, the heterogeneous DDD-FFT approach developed in Chapter 3 could be employed. However, additional developments would be required to properly implement

the recursive approach to compute forces on transmitted dislocation lines. For this reason, and for the sake of simplicity, the regular DDD code developed in Chapter 2 will be employed in this work to study dislocation-CTB interactions in isotropic crystals.

5.1.4 Results and discussion

For each of the simulations, the sequence of events resulting from the interaction of the dislocation with the CTB is illustrated by the snapshots in figure 5.5 and is detailed in the following. Under constant applied stress, the initial dislocation bows out following the Frank-Read source process and propagates until reaching the CTB. At first, transmission does not systematically occur as the stress state is not necessarily sufficient to allow it. Instead, the portion of the dislocation in contact with CTB spreads into a straight line with a 60° character. Eventually, a pile-up forms as a consequence of the Frank-Read source process continuing behind the CTB front until the stress becomes sufficiently high for the transmission to occur. Naturally, as will be shown in the coming section, the rate at which transmission occurs depends on the value of the applied stress σ_{xy}.

Upon transmission, the observed reaction is described by the following relation:

$$\frac{1}{2}[110]_1(\bar{1}11)_1 \rightarrow \frac{1}{2}[101]_2(\bar{1}11)_2 + \frac{1}{6}[\bar{2}11]_2(111)_2 \tag{5.4}$$

where subscripts 1 and 2 denote the parent and the twin phases, respectively, in which the crystallographic quantities are expressed. In relation (5.4), $\vec{\boldsymbol{b}}_{inc} = \frac{1}{2}[110]_1$, $\vec{\boldsymbol{b}}_{trans} = \frac{1}{2}[101]_2$ and $\vec{\boldsymbol{b}}_{res} = \frac{1}{6}[\bar{2}11]_2$. Since the residual Burgers vector lies in the (111) interface, it is equivalently expressed as $\vec{\boldsymbol{b}}_{res} = \frac{1}{6}[2\bar{1}\bar{1}]_1$ in the parent crystal. Relation (5.4) essentially states that, upon intersection, the perfect incoming 60° dislocation on system (A6) in the parent dissociates into another perfect 60° dislocation on system (A3) in the twin and a residual Shockley partial dislocation on the interface. This

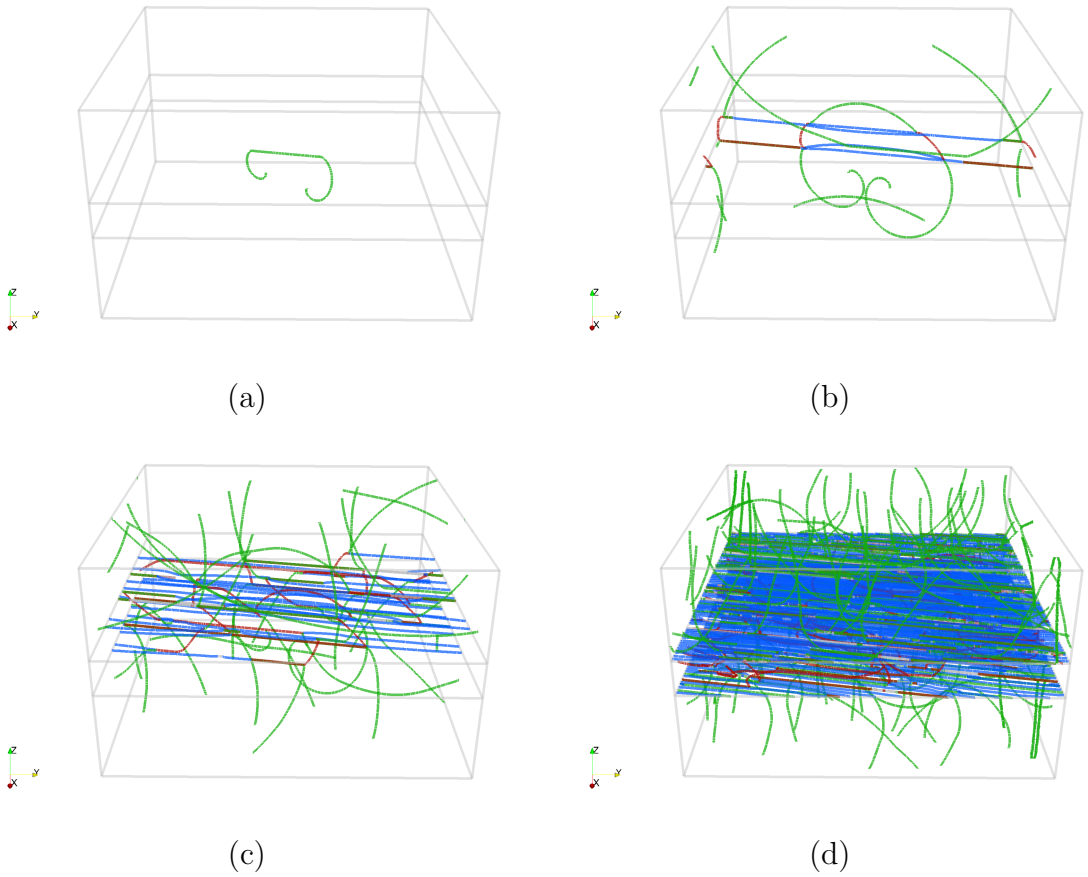


Figure 5.5: Snapshots depicting the different steps of dislocation-CTB interactions. (a) The dislocation source first stops at the CTB until (b) the stress becomes sufficient for transmission to occur. (c) Transmission events further proceed (d) eventually generating a dense network of parallel interfacial dislocations.

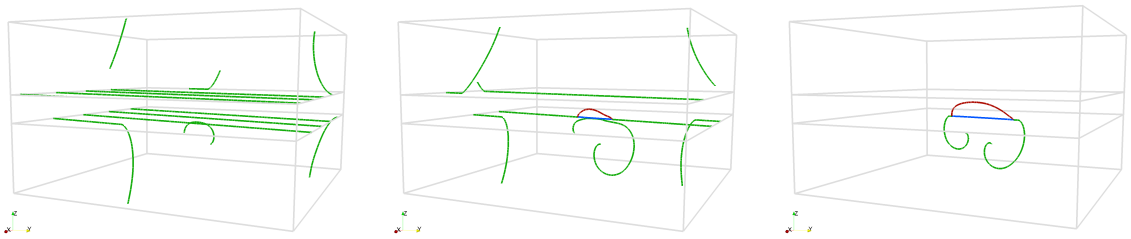
reaction is consistent with that observed in MD [119, 118]. Further, given that the Burgers vector $\vec{\boldsymbol{b}}_{res} = \frac{1}{6}[\bar{2}11]_2$ of the Shockley partial is contained in the (111) CTB plane, reaction (5.4) produces a glissile dislocation which has the same Burgers vector than a twinning dislocation in FCC crystals, and that produces a step whose height is equal to the spacing of the (111) planes. Thereby, such reaction could theoretically evidence a twin growth mechanism via the generation of glissile Shockley partials along the interface. The rate at which dislocations are generated from transmission across the interface is investigated in the next section.

5.1.4.1 Dislocation multiplication and plastic activity

In the current simulations, the transmission of a single intial dislocation source is studied for different values of applied stress σ_{xy}. For each case, the value of the applied stress is chosen below the critical transmission stress $\sigma_{crit} = 410$ MPa such that transmission cannot occur spontaneously. The value of the critical stress obtained in this work has been determined as the value beyond which transmission of the dislocation line occurs when the latter first contacts the interface.

Snapshots of the transmission process of a single dislocation source are reported in figure 5.6. As expected, the time after which transmission occurs – or the conditions in which transmission occurs – are found to depend on the magnitude of the applied stress σ_{xy}. Thus, for $\sigma_{xy} = 190$ MPa, transmission does not occur and the initial dislocation spread along the interface as the Frank-Read source proceeds while the stress induced by the newly formed source resulting from the loop self-annihilation is not sufficient to push the interfacial dislocation across the boundary. In that case, supplementary sources will be required so as to form a pile-up in front of the interface for transmission to occur. For $\sigma_{xy} = 270$ MPa, the initial source similarly spreads along the interface. However, the new source created from self-annihilation of the initial source then pushes the portion of the previously deposited interfacial dislocation as it reaches the boundary. As a result, transmission initiates locally where stress concentrations arising from the pile-up are the most important. For $\sigma_{xy} = 380$ MPa, the process is nearly identical, except that the sole bowing of the initial source is sufficient for the transmission to occur. This result is logical since, in that case, the applied stress is very close to the critical value $\sigma_{crit} = 410$ MPa.

Besides, the influence of the applied stress is not limited to the activation of the transmission process. The evolutions of dislocation densities associated with the present DDD simulations as a function of time (simulations are performed under constant stress) are reported in figure 5.7. Interestingly, as evidenced in figure 5.7(b)



(a) $\sigma_{xy} = 190$ MPa (b) $\sigma_{xy} = 270$ MPa (c) $\sigma_{xy} = 380$ MPa

Figure 5.6: Proceeding of the transmission of a single dislocation source across the CTB as a function of the applied stress σ_{xy}. In (a), the dislocation source spreads along the interface and transmission does not occur. In (b), transmission occurs after the interfacial dislocation line is pushed by a second source. In (c), the sole bowing-out of the initial source is sufficient for transmission to occur.

for $\sigma_{xy} = 380$ MPa, it is found that the sole presence of a twin induces a greater increase in the total dislocation densities (solid lines) than in a single crystal where $t = 0$. This can be explained by the rapid multiplication of dislocations at the CTB when transmission occurs. Essentially, as described by relation (5.4), this mainly results from the fact that an initial incoming dislocation dissociates into two new dislocations upon transmission. This phenomenon has two direct consequences. First, it raises the internal stress state in the vicinity of the CTB promoting further transmission events. Second, the profusion of transmission events leads to a rapid multiplication of partial dislocations moving on the CTB. Such mechanism was already pointed out in [47], in which the rapid consumption of the pile-up by the twin was suggested to eventually mitigate the tendency of hardening by the Hall-Petch effect. This observation is also consistent with the very recent work of Fan et al. [82]. Thereby, such mechanism theoretically evidences a twin growth mechanism via the generation of glissile Shockley partials along the interface. This interpretation is reinforced by the significant number of transmission events occurring in a relatively short lapse of time, as attested by the increase in interfacial dislocation densities (dashed lines) in figure 5.7, and by the fact that these are the sole product of a single initial dislocation source.

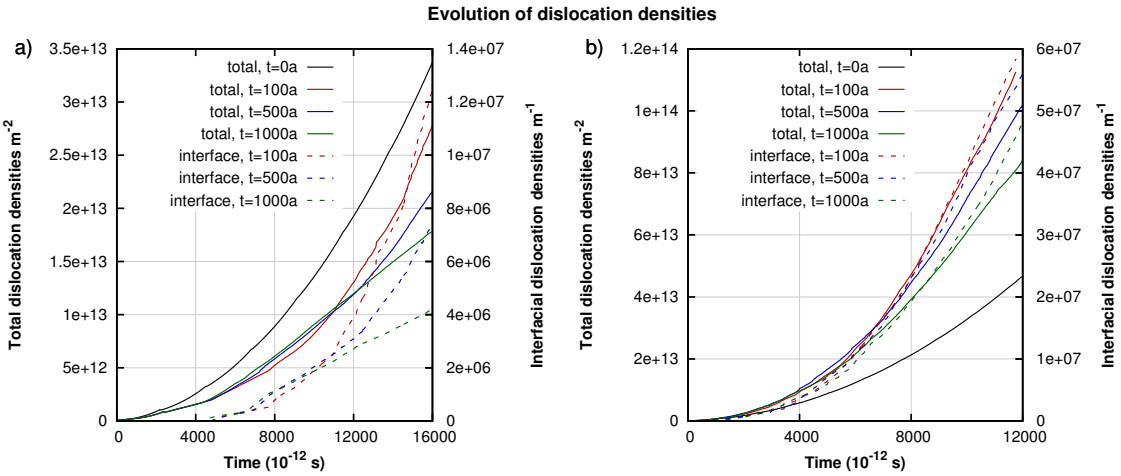


Figure 5.7: Evolution of the total (solid lines, in m^{-2}) and interfacial (dashed lines, in m^{-1}) dislocation densities with time as a function of the twin thickness for (a) $\sigma_{xy} = 270$ MPa and (b) $\sigma_{xy} = 380$ MPa. Results show that the presence of the twin (i.e. cases where $t > 0$) induces a greater increase of total dislocation densities than when no twin is present ($t = 0$) in (b), and that the multiplication of interfacial dislocation densities is proportional to the evolution of the total densities.

Another interesting point can be observed by comparing the densities evolutions as a function of the twin thickness t. It clearly appears from figure 5.7 that a thinner twin produces a more important total dislocation density. The same observation holds for the evolutions of interfacial densities, whose rates of increase are all the more important than the twin fraction is small, and whose evolutions appear to be closely correlated to the total densities for the case $\sigma_{xy} = 380$ MPa. To obtain a better understanding of the relation between total and interfacial densities, the evolutions of the different populations of dislocations composing the total density, namely the densities in the parent, in the twin and on the interface, are plotted in figure 5.8. For each of the populations, the densities are given as the ratio of the total length of dislocation lines over the whole simulation volume. This figure reveals that the density of interfacial dislocations becomes the most important contribution to the total density, followed by dislocations in the parent and in the twin.

Further, besides the evolution of dislocation densities, another interesting aspect

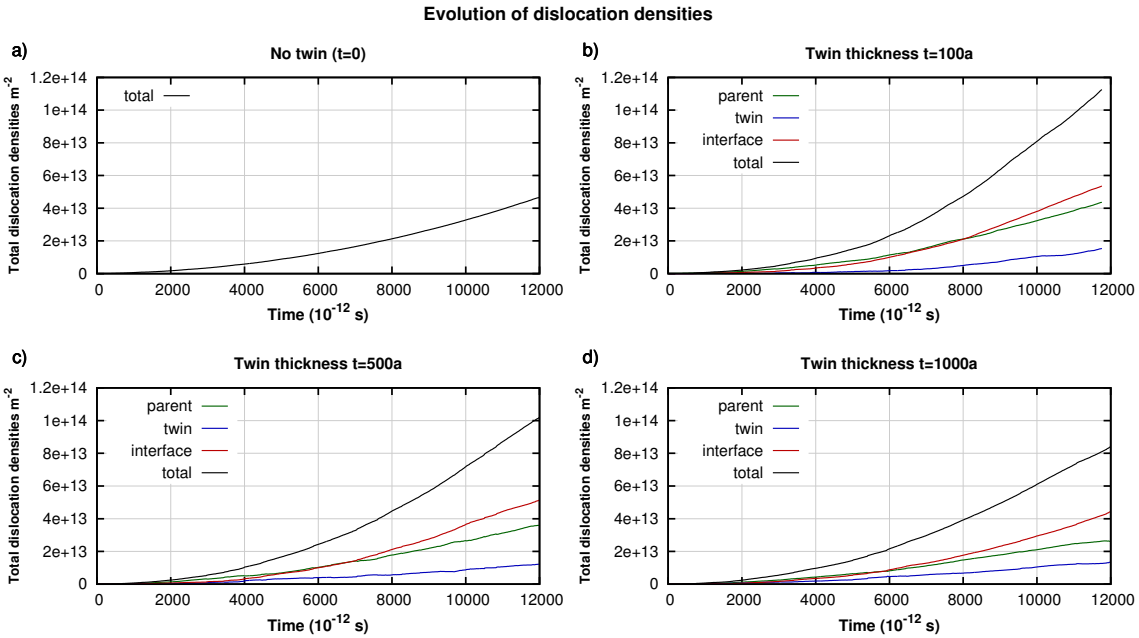


Figure 5.8: Evolution of the densities (in m^{-2}) of the different populations of dislocations with time as a function of the twin thickness for $\sigma_{xy} = 380$ MPa. For each population, the densities are calculated as the ratio of the total length of dislocation lines over the simulation volume V.

to investigate is the plastic activity. For this purpose, the plastic shear rates $\dot{\gamma}$ produced by the different populations are reported figure 5.9. First, the figure shows a decrease in the total plastic activity with increasing twin thickness t, which is consistent with the lower rate of dislocation multiplication observed in figure 5.8 as the twin fraction increases. However, it is interesting to notice that, although interfacial dislocations are the most important contribution to the total density, those do not contribute much to the plastic activity in this case. Instead, incoming dislocations in the parent crystal account for most of the plastic activity. This result has several origins and consequences. First, in the current simulation setting, the applied stress state was purposely chosen so as to obtain a zero Schmid factor on the interface along the direction of the residual Burgers (see Section 5.1.3.2). This implies that the motion of twinning dislocations is solely driven by internal stresses arising from other dislocations. Second, it is to be noticed that the the norm of twinning Burgers vector

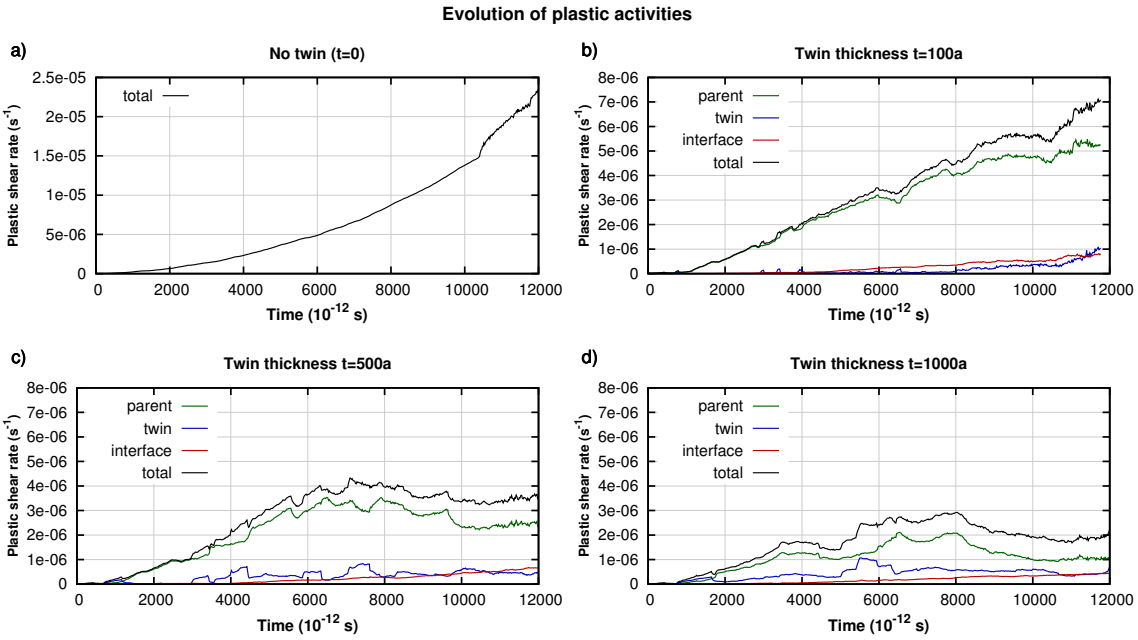


Figure 5.9: Evolution of the plastic shear rate $\dot{\gamma}$ (in s^{-1}) of the different populations of dislocations with time as a function of the twin thickness for $\sigma_{xy} = 380$ MPa.

$\|\vec{\boldsymbol{b}}_{res}\| = 0.148$nm is smaller than that of perfect $\frac{1}{2}[110]$ dislocations ($\|\vec{\boldsymbol{b}}\| = 0.257$nm), such that individual twinning dislocations can accommodate less deformation upon motion than their perfect counterparts, and hence are expected to generate less plastic activity under the same stress. Consequently, despite their ability to glide along the interface, residual twinning dislocations resulting from transmission events do not systematically produce the plastic activity required for twin growth. Specifically, in cases where the direction of the applied stress does not favor their motion, internal stresses alone are not expected to generate significant interfacial plastic activity. Instead, interfacial dislocations can be regarded as stress risers (see Section 5.1.4.2).

5.1.4.2 Evolution of internal stresses in the twin

In the previous paragraph, the origin of the antagonistic evolutions of dislocation density and plastic activity associated with twinning dislocations has been discussed. In this section, the focus is placed on the evolution of the interfacial densities and the resulting internal stresses as a function of the twin fraction. Thus, as reported in figure

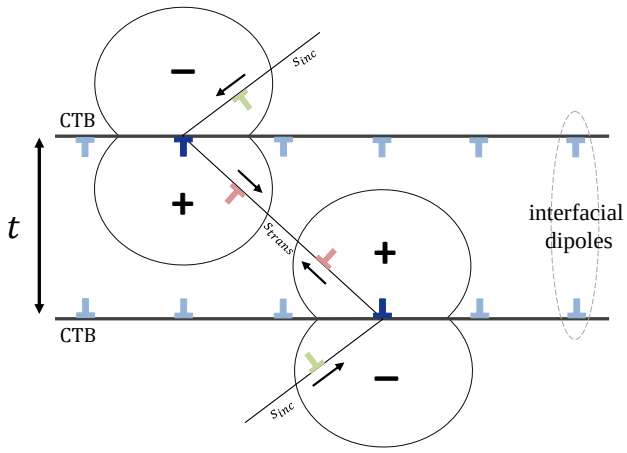
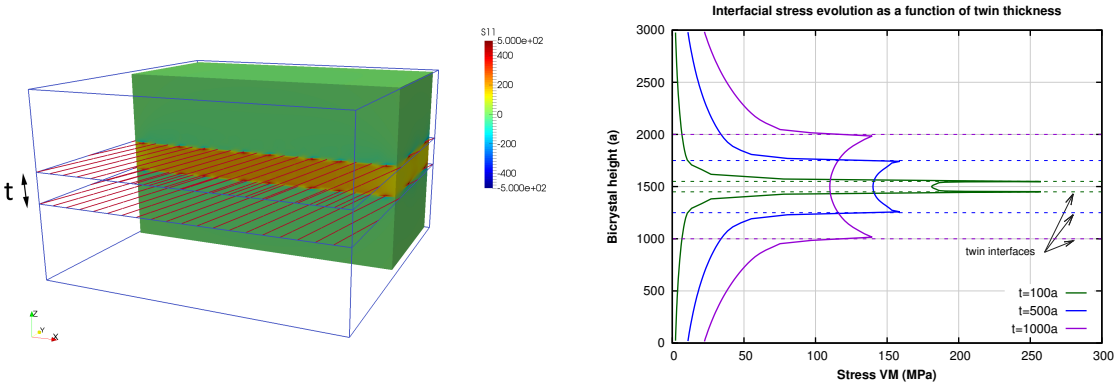


Figure 5.10: Schematic of the generation of interfacial twinning dislocation dipoles upon dislocation transmission across CTB.

5.7(b), it is observed for $\sigma_{xy} = 380$ MPa that a higher twin fraction leads to a lower density of interfacial dislocations. To understand this result, the stress state induced by the partial interfacial dislocations alone is investigated. First, as illustrated in figure 5.5(d), the subsequent multiplication of transmission events eventually leads to the formation of a dense network of interfacial dislocations. Remarkably, those dislocations are quasi-parallel, especially because the crystallography constrains transmission to initially produce pure edge interfacial dislocations. In addition, the line directions of dislocations forming on both edges of the twin are opposite. Consequently, dense arrays of dipolar edge partial dislocations are generated from transmission across the CTB, as schematized in figure 5.10. Interestingly, such arrays have been recently used to describe semi-coherent interfaces via the use of the quantized Frank-Bilby equation [237, 239], and have led to the development of models for the design of patterned interface [238]. Given that partial dislocations are here generated on a single system whose Burgers vector is comprised in the interface, tilt or twist interfaces cannot be reconstructed from this configuration, and the later solely corresponds to pure slip, i.e. inducing twin growth/shrinking.

To assess the stress distribution generated by the interfacial dislocations alone,



(a) (b)

Figure 5.11: (a) Idealized network of parallel interfacial dipolar dislocations. (b) Average Von Mises stress distribution along the height of the bicrystal as a function of the twin thickness. Positions of the twin interfaces are plotted in dashed lines. It is observed that the average stress concentration is inversely proportional to the twin thickness t.

the average stress induced by the idealized dipolar parallel arrays of density $\rho_{int} = 2 \times 10^7 m^{-1}$ shown in figure 5.11(a) is investigated. Figure 5.11(b) shows the resulting Von Mises stress profile averaged at different heights within the bicrystal for several twin thicknesses. The results reveal that, as the twin thickness increases, the average internal stress produced by the interfacial dislocations alone decreases. This is because the distance between interfacial dislocation dipoles resulting from reactions occurring on opposite CTBs is equal to the twin thickness. Therefore, the thinner the twin is, the higher the internal stress concentrations within the twin are. Furthermore, it is seen that the maximum stress concentrations are reached at the CTB, further favoring transmission in their vicinity. Consequently, under a given applied stress, the number of transmission events inversely scales with the twin thickness, hence the higher densities obtained for the case $t = 100a$ in figures 5.7 and 5.8. Conversely, it is observed that internal stresses generated in the parent phase are more important and decay at a slower rate when the twin fraction is important. Thus, interfacial dislocations associated with thin twins generate high stress concentrations in the twin domain while far-field stresses are vanishing rapidly in the parent phase.

5.1.5 New perspectives on dislocation-twin boundary interactions

At higher scales, the observations made in this study are generally not accounted for, and, overall, only few models have attempted to incorporate dislocation-twin interactions in a constitutive fashion or to establish a link with twin-growth. Interestingly, in the work of Capolungo et al. [41] where a slip-assisted twin growth model was proposed, it was suggested that twin growth was predominantly resulting from slip-independent (SI) mechanisms. Specifically, it was suggested that the dislocation activity of partial twinning dislocations at the twin boundary was not sufficient to sustain alone the rate of twin growth. However, it was noticed that the storage coefficient k_1^t in the twin was required to be approximately 2 times larger than that in the parent so that experimental responses can be fitted. Although such result is going against intuition – the presence of the twin boundary is expected to affect both the mean free paths of the parent and the twin in the same way –, it was attributed to a greater disorder in the twin leading to a vast number of initial defects. In line with this interpretation, El-Kadiri, Oppedal and co-workers suggested a transmutation model, in which the higher storage rate of dislocations in the twin is attributed to the increasing density and multiplicity of dislocation types initially in the parent to be incorporated in the twin as the latter grows over the parent phase [78, 189]. In this work, the supplementary internal stress contribution essentially generated in the twin and induced by interfacial dislocations provides an additional explanation for the higher storage rate in the twin.

Further, in all these constitutive models, the precise effect of the dislocation multiplication at the interface is not considered, and the interface is regarded as a CTB during twin growth. The important density at the interface that is observed in this work suggests another behavior. First, it must be recalled that the presence of interfacial Shockley dislocations produce steps and leave stacking faults behind as they are pushed away by other partials. Second, the important dislocation density and the

reactions between interfacial dislocations (e.g. formation of junctions or patterns) suggest important modifications in the character of the interface. Thus, the interface is expected to rapidly become incoherent. Such conclusions have been recently supported by experimental TEM observations on metallic nanocrystals [269]. Further, it is to be noted that simulations are here performed for a single dislocation source. In real crystals, it is expected that dislocations on the 9 incoming systems intersecting plane (111) would similarly dissociate and that the resulting core reactions between the induced partials on the interface would produce an increasingly significant modification of the interface structure as transmission proceeds. Although a precise description of the interface cannot be easily modeled in DDD simulations such that the following consequence is not apparent in the present simulations, it is to be expected that the transmission mechanism expressed in relation (5.4) will eventually stop – or at least will be hindered – by the progressive decoherence of the interface.

In light of the above, a different perspective can be adopted regarding the different sequence of events and the resulting behavior pertaining to dislocation-twin boundary interactions. First, the incoming dislocations in the parent phase pile up against the CTB until the stress state is sufficiently high to allow for transmission. Then, transmission occurs, leading to a rapid increase in the dislocation density, whereby the rate of dislocation generation can exceed that observed when no twin is present. During transmission, full dislocations propagate in the twin, while partial dislocations glide along the twin plane, creating steps, both favoring twin growth. In the meantime, the increase in the internal stress induced by interfacial dislocations leads to increasing the dislocation storage rate in the twin. However, the rapid multiplication of interfacial dislocations briefly transforms the coherent interface into an incoherent boundary, eventually leading to the shutdown of the transmission mechanism. After this point, SI mechanisms take over to sustain twin growth.

In some extent, the investigations of this work may also provide an explanation for the overall hardening obtained in nano-twinned FCC metals. Following the classical interpretation, the transmission at twin boundary tends to produce a softening effect by the promotion of twin growth via the creation of mobile interfacial twinning dislocations, thereby acting against the hardening produced by the Hall-Petch effect. However, it has been seen that, when not favorably oriented, interfacial dislocations generate little plastic activity. In addition, the transmission mechanism is expected to become rapidly difficult to operate due to the incoherency of the interface, such that hardening by pile-ups is overall dominant. Therefore, the complexity of the transmission mechanism highlights the difficulty in incorporating such global picture into higher-scale models. This is probably why a simple Hall-Petch law is not always able to describe it well, especially because it does not account for the initial multiplication of dislocations and the rise of internal stresses in the twin that accompany transmission when the twin boundary remains sufficiently coherent to allow it.

5.2 Dislocation-particles interactions

5.2.1 Motivation

5.2.1.1 Particle-strengthening

Particle strengthening, whereby arrays of precipitates are introduced within a crystal matrix, has been extensively studied in the literature from experimental and theoretical standpoints [183]. Essentially, interactions between moving dislocations and precipitates constitute a very important source of strain hardening in metallic materials, and are particularly dominating in the case of alloys.

In details, particles in the matrix can be of different types – shearable or non-shearable, coherent or incoherent –, of different shapes, and of different spatial distributions, and their effect on the mechanical behavior depends on these factors.

Specifically, second-phase particles are expected to play an important role in hindering dislocations via two aspects. First, in the case of impenetrable obstacles such as incoherent particles, dislocations are stopped upon contact. In that case, bypassing of particles only occurs when dislocations entirely bow around the obstacle to self-annihilate on the other side. Second, differences in the elastic moduli between the matrix and the precipitates give rise to elastic interaction fields that remotely affect the behavior and propagation of dislocation segments. In that case, several factors such as the shape of the obstacle and its distance to the dislocation plane govern the dislocation-particle elastic interaction.

In order to obtain a better understanding of precipitation hardening, several modelling capabilities have been developed at different scales over the past decades. At the constitutive scale, models such as the Dispersed Barrier Hardening [210] and Bacon, Kocks and Scattergood (BKS) [15] have been proposed to account for the collective influence of particle strengthening. However, these models typically rely on simple approaches such as the Orowan bowing of dislocation segments [87, 125], not suited to fully capture the complexity of these interactions. Consequently, refinements of the current obstacle-strengthening models are yet to be developed [171, 221]. To this end, DDD simulations have been employed to obtain insights on individual interactions and on their collective effect so as to delineate enhanced higher-scale predictive models.

5.2.1.2 Previous DDD studies from the literature

Several DDD studies have been dedicated to the interactions between dislocations and particles, in which details were accounted for to a greater or lesser extent. For instance, simple line tension approximations were used to study particles strengthening as a function of particles spacing [88], size [137] and strength [76, 58]. Including more details, Mohles et al. successfully studied the propagation of dislocation lines

within an array of coherent and incoherent shearable particles using DDD simulations in which obstacles were modelled by a supplementary stress contribution arising from an antiphase boundary energy density [171, 173, 172]. While this supplementary contribution was added to the Peach-Koehler force on dislocation lines inside the particles, the latter was not considered outside the particles, such that elastic interaction effects were ignored. Similarly, in the work of Queyreau and co-workers where the Orowan by-passing mechanism on incoherent carbides was studied in BCC reactor pressure vessel steels, no stress field was associated to the impenetrable obstacles [199]. More recently, the regular DDD code developed in this thesis was further used in [221] to assess the validity of particle-hardening models for a combination of voids and self-interstitial atom loops. In this last work, elastic interactions with voids were modelled using an atomistically-informed breakaway angle.

Nonetheless, a few DDD simulations have readily incorporated particles elastic interaction effects. This was first achieved in the seminal work of Shin and co-workers [219, 217, 218]. In their approach, the superposition method proposed by Van der Giessen and Needleman [232] (see Section 2.4.2) was further extended to account for second-phase elastic inclusions in the matrix. By coupling the DDD simulations to a FEM code, the heterogeneous stress field generated by the elastic mismatch between the matrix and the precipitate were accounted for in the form of image forces so as to investigate the interaction of dislocations with cubical [219] and spherical [217] precipitates. Using the same superposition approach, interaction forces between a straight dislocation line and spherical particles [216] and voids [106] were calculated. However, such approach is computationally very intensive due to the fine FEM meshes that are required, and studies are therefore limited to the investigation of static or relaxed configurations and cannot be practically extended to perform dynamic simulation up to relevant levels of strain. As an alternative approach to the problem, the DCM-FEM model [147] was used to model the plastic deformation in metal matrix

composites [100], and investigate plasticity in nickel-based single-crystal superalloys in which matrix channels were formed by the presence of precipitate phases [235, 236]. However, as mentioned in Section 3.2, although inherently accounting for heterogeneous elasticity, the DCM-FEM approach is limited to a coarse representation of precipitates and cannot be employed to finely model particles. Furthermore, most of the current methods are limited to use elastic isotropy.

5.2.1.3 Challenges and objectives

In light of the above, current DDD approaches need to be refined and their limitations need to be addressed so as to (1) incorporate more details when studying interactions between particles and dislocations, (2) extend their time and length scales such that refined models to be incorporated at the constitutive level can be delineated. Particularly, an accurate and efficient treatment of image forces arising from the elastic mismatch between the matrix and the particles needs to be incorporated.

For such purpose, and to address the limitations of current approaches, the heterogeneous extension of the DDD-FFT tool developed in Chapter 3 of this thesis appears as a perfectly suited tool. With this approach, it is expected that a level of details never achieved so far can be incorporated to study interactions between dislocation and particles.

This section is organized as follows. First, the elastic interactions between straight dislocations and particles are investigated in a static setting using the elasto-plastic heterogeneous DDD-FFT approach, for which numerical aspects are discussed in Section 5.2.2. Then, the method is employed in the dynamic case to examine the relaxation of a dislocation line near a shearable particle in Section 5.2.3. Finally, perspectives on the present method's ability to address current open challenges are discussed in Section 5.2.4.

5.2.2 Elastic interactions

As a first application, the heterogeneous DDD-FFT approach is used to study elastic dislocation-particle interactions in a static setting. By assigning different elastic properties to the voxels partitioning the simulation volume, the method is inherently capable of solving for the stress field associated with the heterogeneous elasto-plastic continuum. Further, the numerical efficiency associated with FFT-based solver allows for the use of very fine grids, thereby producing high-resolution results compared to what is currently achievable with DDD-FEM approaches. For a detailed description of the heterogeneous DDD-FFT formulation and its numerical implementation, the reader is referred to Sections 3.5.2 and 3.6.3. As a result, elastic interaction fields between a dislocation line and a particle due to the elastic mismatch between phases – giving rise to image forces on dislocation lines – are directly accounted for in the stress field computed at the Fourier grid points.

In this first section, several aspects of the method and the results are studied. Thus, the elastic interactions obtained for different particle stiffnesses are compared, and the computational efficiency and the numerical convergence of the different heterogeneous iterative schemes presented in Section 3.5.3 are assessed. For this study, HCP Mg, whose material parameters used in this section are given in table 5.2, is chosen as a paradigm material. As reported in table 5.2, transversely isotropic elastic constants are used. Although the corresponding anisotropic ratio $A = 0.979$ (see definition in equation (3.63)) is very close to unity, deviations from isotropic elasticity in the resulting stress field are clearly perceptible, especially because using an isotropic elastic approximation requires averaging the elastic coefficients to obtain two independent values whose choice is not unique (e.g. Reuss or Voigt bounds).

The two settings that are investigated are illustrated in figure 5.12. First, a prismatic loop is introduced on one side of a bicrystal at a distance d from the interface. Then, the interaction with the same prismatic loop is investigated when a

Table 5.2: Material parameters used to model transversely isotropic HCP Mg.

Material parameter	Value
Elastic constants	$C_{11} = C_{22} = 59.74$ GPa
	$C_{33} = 61.70$ GPa
	$C_{44} = C_{55} = C_{66} = 16.39$ GPa
	$C_{12} = 26.24$ GPa
	$C_{13} = C_{23} = 21.70$ GPa
Lattice parameters	$a = 3.21 \times 10^{-10}$ m
	$c/a = 1.623$

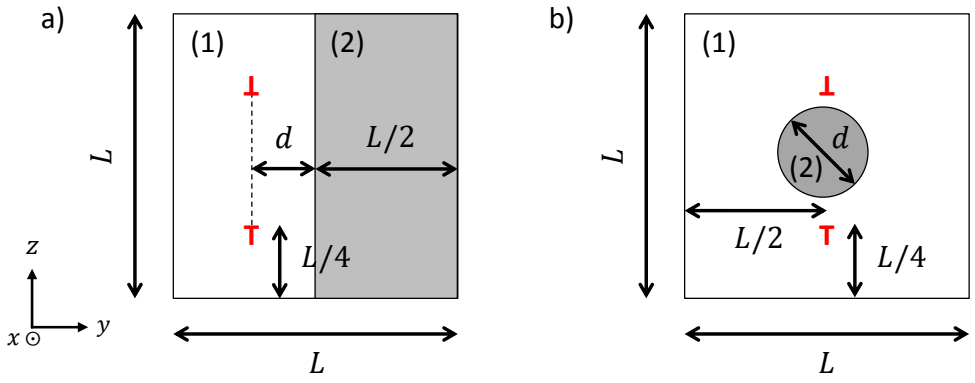


Figure 5.12: Settings used to investigate the heterogeneous elasto-plastic DDD-FFT framework. (a) A prismatic loop is introduced on the left side (1) of the bicrystal at a distance d from the interface. The position of the dislocation core on the (yz) plane is located with the $\perp$ symbols. (b) A prismatic loop and a spherical inclusion (2) are simultaneously introduced in matrix (1).

spherical particle is introduced in the center of the simulation volume.

5.2.2.1 Dislocation in a bi-crystal

In this first setting, a prismatic loop is introduced on the (xz) plane in the left side of a bicrystal at a distance d from the interface (see figure 5.12(a)). The left crystal (1) is chosen as a pure HCP Mg crystal with transversely isotropic elastic constants reported in table 5.2. In contrast with the procedure described in figure 3.4, the width of the prismatic loop in the x direction is chosen as the volume size L in this direction such as to end up with two edge segments parallel to the x direction – by periodicity the two complementary edge segments parallel to the z direction mutually

annihilate on the edge of the simulation volume. As a result, the initial configuration in the left crystal (illustrated along a yz slice in figure 5.12(a)) includes a dislocation dipole composed of two edge $\frac{1}{3}[\bar{1}2\bar{1}0]$ dislocations on (0001) basal planes.

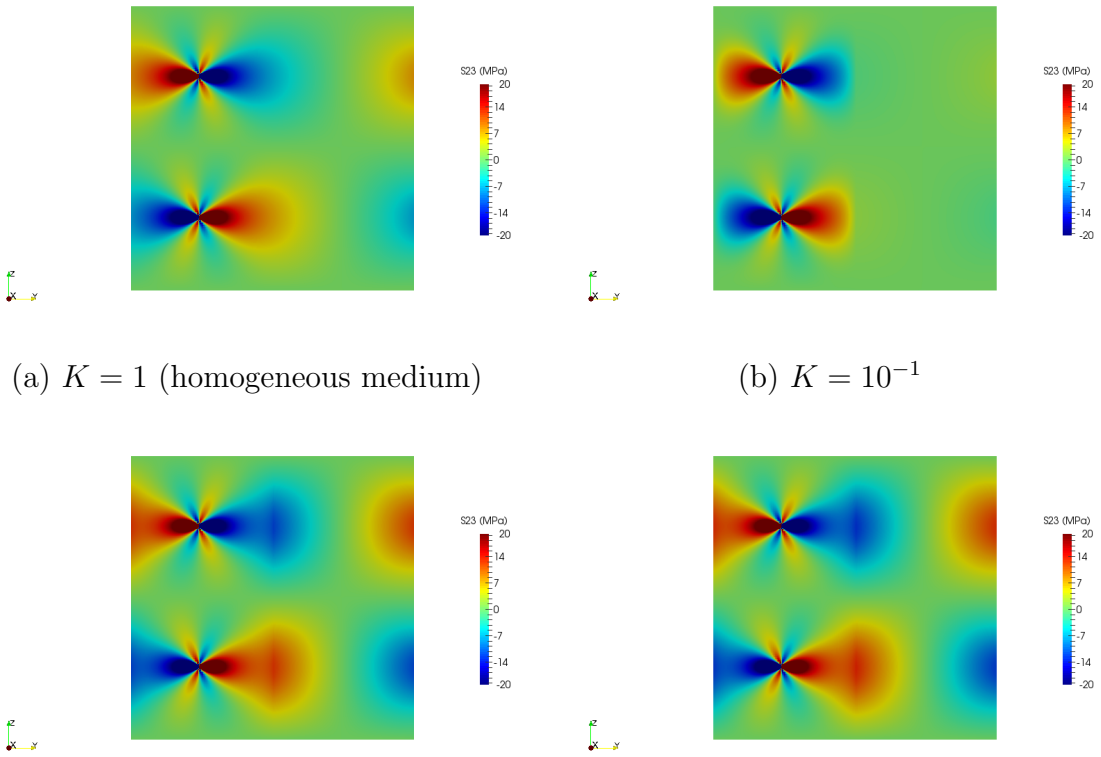
To simulate the interaction of the dislocation with a large square particle, the right crystal (2) is chosen as a coherent HCP crystal with the same lattice spacing a as the Mg crystal (1), but whose contrast K in the elastic constants is given as:

$$K = \frac{C^2_{11}}{C^1_{11}} = \frac{C^2_{12}}{C^1_{12}} = \frac{C^2_{13}}{C^1_{13}} = \frac{C^2_{33}}{C^1_{33}} = \frac{C^2_{44}}{C^1_{44}} \tag{5.5}$$

where C^1_{ij} and C^2_{ij} denote the elastic constants of the left Mg crystal (1) and that of the particle (2), respectively.

The periodic simulation volume is chosen as a square box of side $L = 2000a$ where $a = 0.321$nm is the lattice parameter of Mg, discretized into 128^3 voxels, and the backward scheme (W) is used. First, the dislocation is positioned at the center of the left crystal, i.e. at $d = L/4$. The results at convergence obtained for the σ_{23} component along a slice at $x = L/2$ under no loading ($\boldsymbol{E} = \boldsymbol{0}$) for different contrasts K are plotted in figure 5.13.

Here, the purpose of this analysis does not rely in the assessment of the validity of the method – such is done in Appendix G.2 and to the author's knowledge no direct analytical solution exists – but rather to provide insights on the influence of heterogeneous elasticity on the stress field of dislocations. Thus, when the dislocation lies in a stiffer medium than the right crystal, i.e. for $K \leq 1$, a contraction at the edge of the resulting stress field occurs, as attested by the case reported in figure 5.13(b) for $K = 10^{-1}$. Such result is consistent since the softer crystal can withstand less stress than when the medium is homogeneous. Conversely, the edge of the stress field appears as stretched when the right crystal is stiffer than the first phase, as plotted in figure 5.13(c) for $K = 10$. The case $K = 10^3$ reported in figure 5.13(d) is interesting as no clear difference appears from a visual comparison with $K = 10$.



(a) $K = 1$ (homogeneous medium)

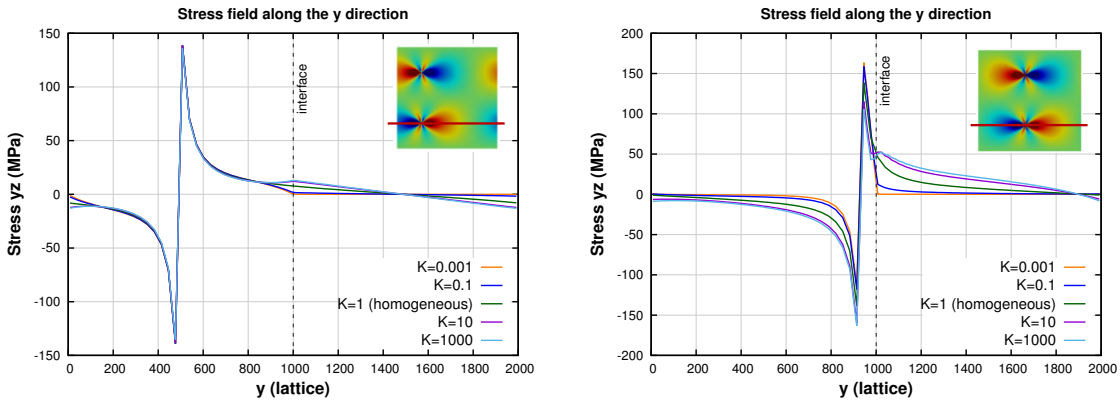
(b) $K = 10^{-1}$

(c) $K = 10$

(d) $K = 10^3$

Figure 5.13: Results for the σ_{23} stress component produced by a prismatic dislocation loop in an elastic bicrystal at a distance $d = L/4$ from the interface (see figure 5.12(a)) and for different contrast values K. The reported results are obtained with the backward (W) scheme for a resolution of 128^3 voxels after convergence is reached for a precision $\epsilon^{tol} = 10^{-4}$ (see equation (3.81)).

A finer comparison can be made by looking at figure 5.14(a) where the evolution of the stress along a y-line is given for different values of contrast K. As shown, the stress evolutions for $K = 10$ and $K = 10^3$ are extremely close, and it is further seen (not reported here) that increasing the value of K has no further effect on the stress evolution. Therefore, there exists an upper limit (here $K \approx 10^3$) beyond which the stress field asymptotically tends towards the evolution reported for $K = 10^3$. The same is observed for the cases $K \leq 0$. Thus, the case $K = 10^{-3}$ represents the lower asymptotic limit in which the stress in the right crystal vanishes, and below which lower values of K lead to the same stress evolutions.



(a) $d = L/4$ (b) $d = L/32$

Figure 5.14: Evolution of the σ_{23} stress component produced by a prismatic dislocation loop in an elastic bi-crystal for different contrast values K and for a resolution of 128^3 voxels. The evolution is plotted along the red y-line shown in the insets when the dislocation is positioned at a distance (a) $d = L/4$ and (b) $d = L/32$ from the interface located at abscissa $y = L/2 = 0.5$.

Regarding numerical aspects, the number of iterations to reach convergence strongly scales with the magnitude of the contrast K, as attested by the results provided in table 5.3. Surprisingly, it is also to be noted that, in comparison with the purely elastic results obtained in Appendix G.2, the choice of the gradient operator has no effect on the convergence rate when a distribution of plastic strain exists. Instead, it solely has an influence on the Gibbs phenomenon, as already reported for the case of an homogeneous medium (e.g. see figure G.2). Further, the results show that while the *accelerated* scheme provides a faster convergence than the *basic* scheme, the *conjugate-gradient* method consistently yields superior efficiency, which is especially remarkable for high contrasts. Consequently, the *conjugate-gradient* scheme will be preferably used in the heterogeneous DDD-FFT formulation.

Besides, as can be suspected from the results in figure 5.14(a), the alteration of the initially homogeneous stress profile as a function of K is intrinsically linked to the distance of the dislocation core to the interface – or more generally, to the distribution of the plastic strain with respect to the heterogeneous elastic distribution. The resulting stresses obtained when the dislocation is positioned at a closer distance

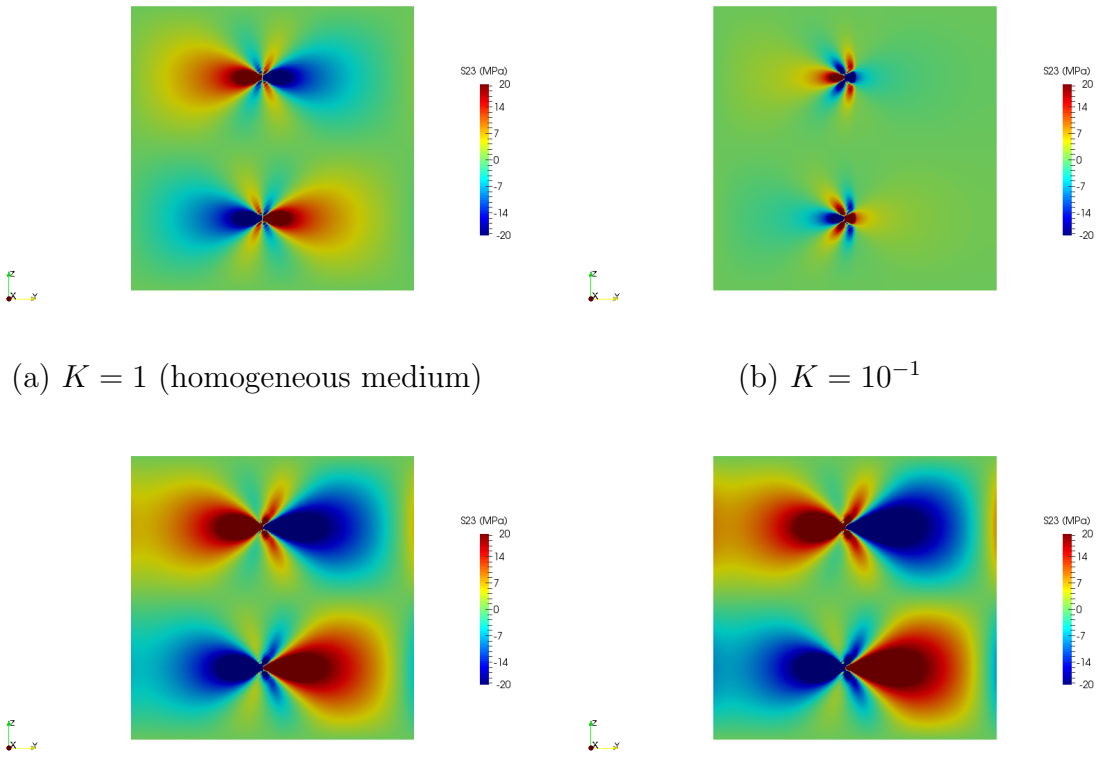
Table 5.3: Effect of the different iterative and gradient operators schemes on the number of iterations N to reach convergence for a precision $\epsilon^{tol} = 10^{-4}$ and for different contrasts K between both crystals. NC indicates that the scheme does not converge.

Iterative scheme	Gradient operator	$K = 10^{-3}$	$K = 10^{-1}$	$K = 2$	$K = 10$	$K = 10^2$	$K = 10^3$
basic	continuous	NC	85	16	73	611	4951
accelerated	continuous	28	19	11	42	326	2653
conjugate gradient	continuous	10	9	7	10	17	20
basic	centered (C)	NC	85	16	73	611	4951
accelerated	centered (C)	28	19	11	42	326	2653
conjugate gradient	centered (C)	10	9	7	10	17	20
basic	backward (W)	NC	85	16	73	611	4951
accelerated	backward (W)	28	19	11	42	326	2653
conjugate gradient	backward (W)	10	9	7	10	17	20
basic	rotational (R)	NC	85	16	73	611	4951
accelerated	rotational (R)	28	19	11	42	326	2653
conjugate gradient	rotational (R)	10	9	7	10	17	20

$d = L/32$ from the interface are reported in figures 5.14(b) and 5.15. As expected, the modification of the stress distribution as a function of the contrast K is significantly more pronounced than for $d = L/4$. Although this result cannot be interpreted as a validation of the heterogeneous DDD-FFT approach, it nevertheless provides insights on the method's ability to handle elastic interactions between dislocation lines and inclusions. However, recall that the formulation solely ensures that the mechanical equilibrium (3.71) is respected (within the tolerance) everywhere in the medium – including in the vicinity of the interface –, and the question of whether a supplementary interfacial condition must be included is not trivial. A complete validation of the approach would require a comparison with an analytical formulation, which is not directly available. Finally, it is observed that the number of iterations to reach convergence is identical to that reported for $d = L/4$ in table 5.3. Therefore, it is expected that, overall, the convergence rate would be rather linked to the contrast K than to the plastic distribution.

5.2.2.2 Dislocation interaction with spherical inclusion

As a second application, the interaction between a prismatic dislocation loop and a spherical inclusion is investigated. The configuration of the simulation illustrated in

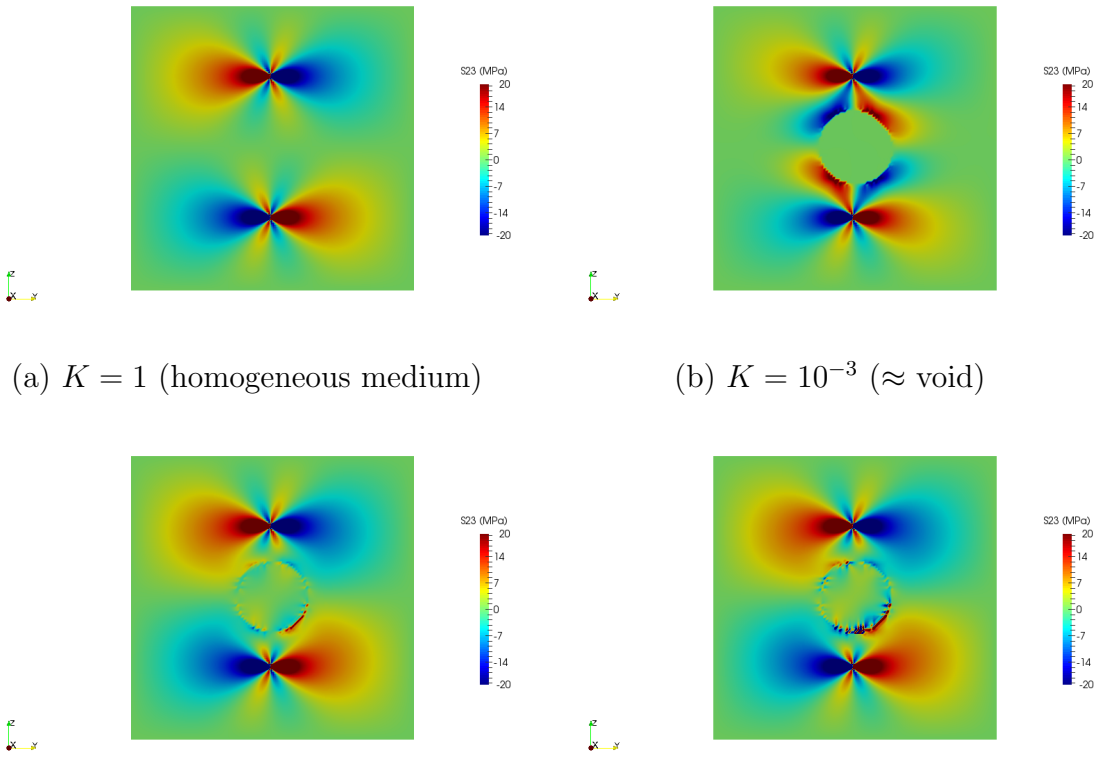


(a) $K = 1$ (homogeneous medium) (b) $K = 10^{-1}$

(c) $K = 10$ (d) $K = 10^{3}$

Figure 5.15: Results for the σ_{23} stress component produced by a prismatic dislocation loop in an elastic bicrystal at a distance $d = L/32$ from the interface (see figure 5.12(a)) and for different contrast values K. The reported results are obtained with the backward (W) scheme for a resolution of 128^3 voxels after convergence is reached for a precision $\epsilon^{tol} = 10^{-4}$.

figure 5.12(b) is as follows: a dislocation loop and a spherical particle of diameter d are simultaneously introduced at the center of a matrix phase. The resulting stress field obtained on a (yz) slice at $x = L/2$ under no loading for different values of contrasts K and for $d = L/4$ are plotted in figure 5.16. The effect of the elastic interaction with the inclusion on the stress field of the dislocation is clearly apparent and differs according to the contrast. Thus, for $K = 10^{-3}$ where the inclusion is porous and can be assimilated to a void, the side lobes of the stress field are attracted to the void while the main lobes are repelled and pushed backwards. Conversely, when the particle is stiffer than the matrix, as in the case $K = 10^3$ for instance, the edges of



(a) $K = 1$ (homogeneous medium)

(b) $K = 10^{-3}$ ($\approx$ void)

(c) $K = 10$

(d) $K = 10^3$

Figure 5.16: Results for the σ_{23} stress component along a (yz) slice resulting from the interaction between a prismatic dislocation loop and a spherical inclusion of diameter $d = L/4$ for different contrasts K. The reported results are obtained with the backward (W) scheme for a resolution of 128^3 voxels after convergence is reached for a precision $\epsilon^{tol} = 10^{-4}$.

the main lobes of the stress field are attracted by the inclusion.

Although no Gibbs oscillations are produced, the stress field obtained at the boundary of the inclusion is jagged. This simply results from the discrete description of the spherical inclusion: in this example, all voxels whose center position lies within the sphere of diameter $d = L/4$ are assigned the inclusion elastic properties, while the other voxels are assigned that of the matrix. As a result, the outer contour of the inclusion is approximated by a succession of small square surfaces (facets of the voxels) forming a staircase delimitation. To attenuate the stress gradients in the vicinity of the inclusion boundary, a smoother description of the contour could be adopted.

For instance, an interpolation based on the volume ratio of each voxel covered by the inclusion could be used to define the elastic properties of the voxels lying in the vicinity of the boundary.

Regarding the convergence rate, it is observed that, independently of the size of the inclusion size d, the number of iterations to reach convergence when using the conjugate-gradient scheme systematically lies within a 10% interval around the values reported in table 5.3 in the case of the bi-crystal. For instance, similar convergence results were observed for an array of 10×10 regularly spaced spherical inclusions of diameter $d = L/50$ (not reported here). Therefore, the number of iterations to convergence is expected to be independent of the geometry and volume fraction of the inclusion(s), but to be primarily a function of the contrast K.

Finally, it must be noticed that, as illustrated throughout this section, the heterogeneous DDD-FFT approach provides an efficient way to perform simulations in which the elastic effects of dislocation–particles can be included. Such simulations were already permitted with original DCM model, although precipitates were limited to be represented by coarse square inclusions [235, 236]. In this work, the higher resolutions offered by the FFT-based solver allows for a finer description of particles (e.g. small precipitates).

5.2.3 Dynamic interactions

In this section, the heterogeneous DDD-FFT approach is used to study dislocation-particle interactions in a dynamic context. Specifically, the relaxation of an initially straight edge dislocation in the vicinity of a particle introduced on its slip plane is examined. For this purpose, the setting illustrated in figure 5.17 is employed. As in Section 5.2.2, a prismatic dipole is initially introduced in a HCP Mg crystal with properties given in table 5.2. Upon insertion, the prismatic loop is elongated along the x-axis such that both segments parallel to the z-axis annihilate by periodicity.

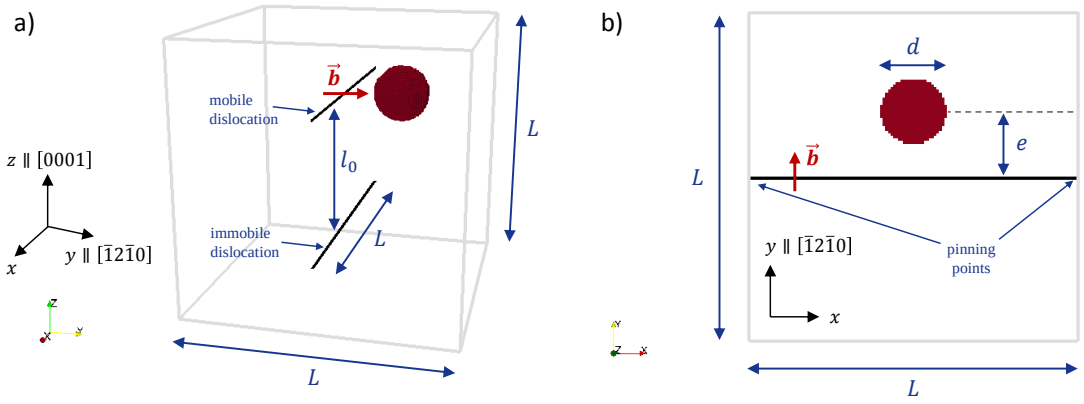
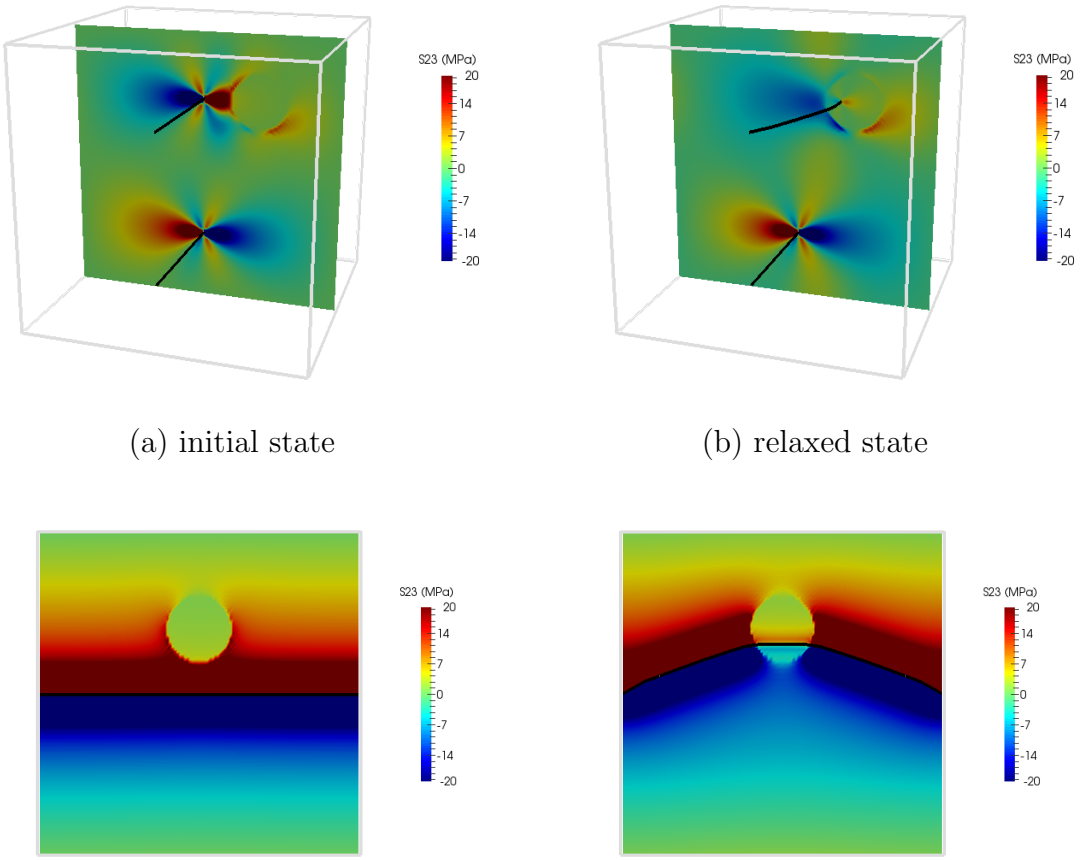


Figure 5.17: Setting used to investigate the relaxation of a basal edge dislocation in the vicinity of a spherical particle in a HCP Mg crystal. (a) A prismatic dislocation loop with Burgers vector $\vec{\boldsymbol{b}}_{\langle a \rangle} = \frac{1}{3}[\bar{1}2\bar{1}0]$ and of length $L \times l_0$ is initially introduced on the (xz) plane at the middle of a simulation volume of size $V = L^3$. By periodicity, both dislocation segments parallel to the z-axis annihilate. (b) Top view in the (0001) basal plane of the top mobile dislocation. A spherical particle of diameter d is introduced at a distance e from the line pinned at the edges.

As a result, two dipolar edge $\frac{1}{3}[\bar{1}2\bar{1}0]$ dislocations separated by a distance l_0 are initially obtained on (0001) basal planes. A spherical shearable particle of diameter $d = L/5$ and elastic contrast K is introduced in the (0001) basal slip plane of the top dislocation at a distance $e = L/5$ from its line. To avoid unconstrained motion, the top dislocation is pinned at the edge of the simulation volume, while the bottom one is held immobile.

Results for a porous particle with contrast $K = 10^{-1}$ are illustrated in figure 5.18. In this dynamic simulation, no external strain is imposed ($\boldsymbol{E} = \boldsymbol{0}$) and a resolution of 128^3 voxels is used. As expected, the dislocation line is attracted by the porous inclusion to reduce the elastic energy of the system. Since the particle is shearable, the dislocation bows out as its middle portion is dragged into the particle until reaching an equilibrium position. For a non-shearable precipitate, the dislocation would have stopped at the surface of the particle. In that case, a special numerical treatment



(a) initial state

(b) relaxed state

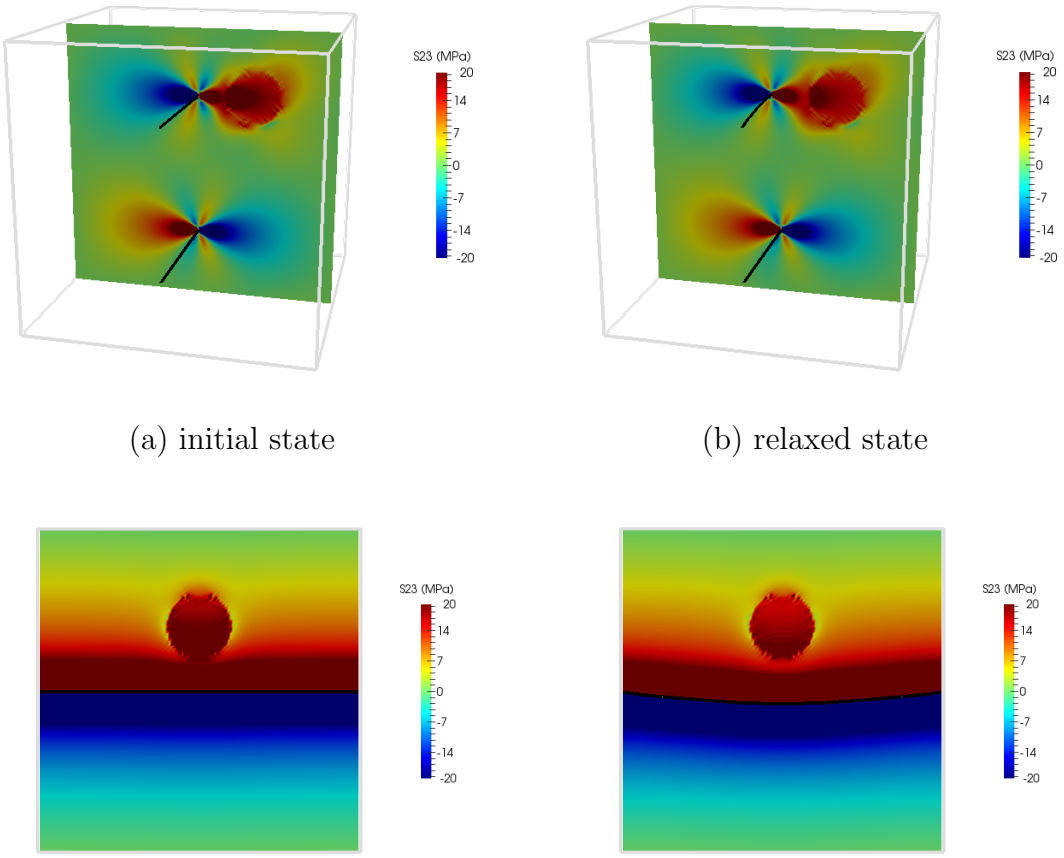
(c) initial state, slip plane view

(d) relaxed state, slip plane view

Figure 5.18: Snapshots of the (a) initial and (b) final states of the relaxation of an initially straight edge basal dislocation in HCP Mg in the vicinity of a porous inclusion with contrast $K = 10^{-1}$ for a resolution of 128^3 voxels. Upon relaxation, the dislocation is attracted by the particle and the middle part of the line is dragged inside it to a minimum energy state. The driving σ_{23} stress component resulting from the interaction between the dislocation and the inclusion is plotted on a (yz) slice at $x = L/2$. σ_{23} stress fields on the (0001) slip plane of the dislocation are shown in (c) and (d).

such as that employed in [199] should be implemented to ensure that dislocation stop upon contact with the particle.

Results for the relaxation of the dislocation in the vicinity of a stiff particle with contrast $K = 10$ are reported in figure 5.19. Conversely to the porous inclusion, the stiff inclusion repels the dislocation line. As a result, the middle portion of the line



(a) initial state

(b) relaxed state

(c) initial state, slip plane view

(d) relaxed state, slip plane view

Figure 5.19: Snapshots of the (a) initial and (b) final states of the relaxation of an initially straight edge basal dislocation in HCP Mg in the vicinity of a stiff particle with contrast $K = 10$ for a resolution of 128^3 voxels. Upon relaxation, the dislocation is repelled by the particle and the line bows out until reaching equilibrium. The driving σ_{23} stress component resulting from the interaction between the dislocation and the inclusion is plotted on a (yz) slice at $x = L/2$. σ_{23} stress fields on the (0001) slip plane of the dislocation are shown in (c) and (d).

– i.e. the closet to the particle – is first pushed away from the particle, after which the self-force comes into play to give the dislocation line a curved shape (see figure 5.19(d)) so as to minimize the elastic energy of the system.

Through these two relaxation examples, the inherent ability of the heterogeneous DDD-FFT method to incorporate image forces resulting from heterogeneous elasticity

is assessed in the dynamic case. With respect to the DCM-FEM approach, the current approach developed in this thesis is computationally more efficient (see Section 3.7) and allows for finer resolutions that enable a precise description of stress fields and particle morphologies in heterogeneous elasticity. Furthermore, as a full-field method, the current approach allows for a direct visualization of mechanical fields within the whole simulation volume, as reported in figures 5.18 and 5.19 for instance.

5.2.4 Conclusion and perspectives

In Sections 5.2.2 and 5.2.3, a first application of the heterogeneous DDD-FFT method developed in Section 3.5 was successfully dedicated to the investigation the interactions between dislocations and particles of various shapes in transversely isotropic HCP Mg for static and dynamic configurations. Through these diverse examples, the promising potential of the DDD-FFT tool developed in this thesis – which is capable of simultaneously treating anisotropic and heterogeneous elasticity while being numerically efficient – was demonstrated.

Naturally, the results presented in this section constitute a first application and further studies will be undertaken using heterogeneous DDD-FFT approach. Particularly, it is expected that this method could be successfully employed to investigate the effect of the shape (plate, rod, sphere, see figure 5.20) and orientation of precipitates as barriers to dislocation propagation in HCP Mg, so as to assess the validity of the Orowan strengthening models postulated in [184]. For instance, Nie suggested that precipitate plates on prismatic planes should induce higher strengthening compared to all types of possible precipitates in the basal planes, by providing the most efficient barrier to basal slip and $\{10\bar{1}2\}$ twin propagation [185]. However, such speculations were never investigated at the DDD scale in HCP Mg. In doing this, it is further expected that hardening trends could be extracted to be further incorporated into constitutive laws, thereby guiding microstructural design for fabricating Mg alloys

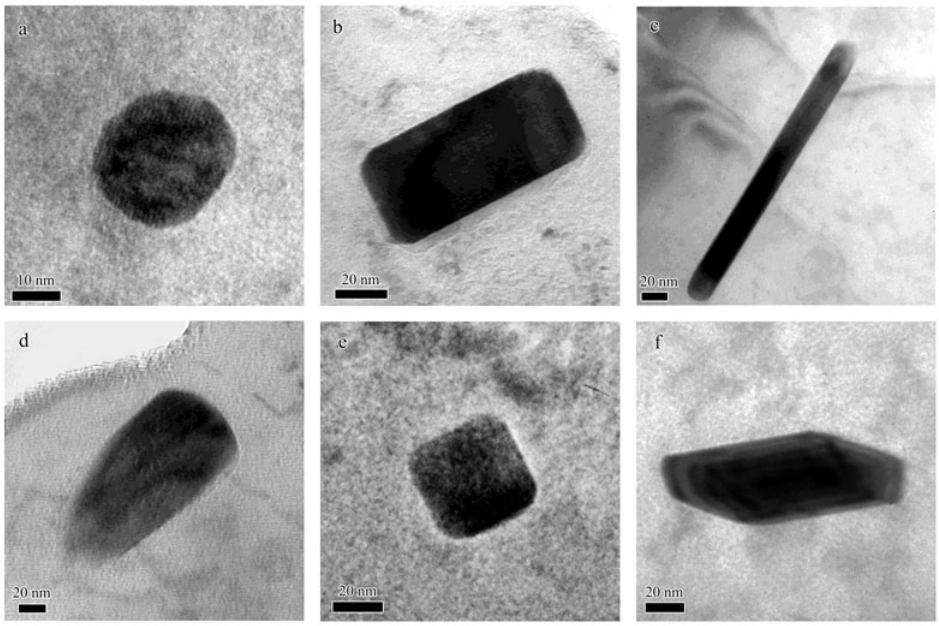


Figure 5.20: Examples of various shapes obtained for Al-Mn-(Mg) particles in AZ91 magnesium alloy: (a) irregular globule, (b) short rod, (c) long rod, (d) tear-drop, (e) rounded rectangle, (f) irregular hexagon. Reproduced from [268].

with higher strength.

In addition, the method could be employed to study dislocation-particle interactions in various metals, and should be particularly useful to examine such mechanisms in low-symmetry crystals in which the amount of elastic anisotropy is important. Furthermore, as demonstrated in this section, the current method is not limited to model precipitates, but can be employed to examine interactions with voids and phases of different shapes. For instance, DDD simulations in bi-crystals or multi-layered laminates could be performed, and the effect of anisotropic elastic mismatch to study dislocation-twin interactions –which was not accounted for in the study presented in Section 5.1 – could be examined.

Finally, it is important to notice that, in contrast to the applications presented in this section, the heterogeneous DDD-FFT method can virtually be employed to treat an infinite number of phases simultaneously, thereby paving the way towards performing full-scale DDD simulations in polycrystalline materials.

CHAPTER VI

CONNECTING DDD SIMULATIONS TO CONSTITUTIVE LAWS: THE HYBRID MODEL

6.1 Introduction

The main objective of this chapter is to delineate novel routes allowing for the connection between DDD and dislocation density based models so as to incorporate in a more direct fashion the information pertaining to the collective effect of unit processes and extracted at lower scales, with the intent to improve the predictive capabilities of constitutive laws. For such purpose, a slip-driven dislocation density-based crystal plasticity model – called the Hybrid Model –, that extends current formulations to incorporate more details on the physics of dislocations is developed.

The primary motivation for this work arises from the various limitations associated with current models. Specifically, it has been observed that current approaches generally focus on describing the evolution of stored dislocations, whereas evolution of mobile dislocations, acting as plastic carriers, is often disregarded. Furthermore, the increasing number of fitting parameters accompanying the incorporation of additional mechanisms in current models may paradoxically lead to a disconnection from the physics of dislocations. Specifically, it has been shown that a given stress-strain response can be identically reproduced by the same model with two different sets of parameters, while predicted internal state variables evolve in drastically distinct manners [22]. In addition, it was reported that predicted dislocation densities often do not concur with experimental observations [49]. In light of the above, a new constitutive model is proposed. The goal of this work is to develop a model that intrinsically accounts for the physics of dislocations, such as to incorporate the treatment of their

motion, interactions and transformations.

The idea is thus to allow for a quantification of the different dislocation densities in a constitutive fashion. To this end, the dislocation density in each slip system is decomposed into glissile and stored dislocations, and interconnected rate equations are introduced for their evolution. This is achieved by introducing a two step-approach in which all glissile non-polar dislocations on a given slip system are represented by a virtual dislocation loop which evolution is modelled by a dislocation dynamics approach, while transformations of dislocations from glissile to stored, resulting from short-range dislocation-dislocation interactions, are based on phenomenological relations informed by dislocation dynamics simulations on dislocation pair interactions.

With this, the proposed Hybrid Model connects DDD to larger scale dislocation based density constitutive models in ways that (1) it will benefit from a reduction in the number of fitting parameters with respect to current dislocation density based models and (2) it should lay the framework of models capable of predicting the mechanical responses under complex loading paths (i.e. strain path changes, shock, Bauschinger effect, etc.).

As a first application, the Hybrid Model is utilized for predicting the stress-strain response of a single crystal aluminum as a function of its orientation, slip activity, and junction formation, all with a single set of parameters.

6.2 Dislocation populations

The core idea of dislocation density based models relies on introducing a distinction between different dislocation types and deriving specific sets of evolution laws for each of them. As each individual dislocation population dominantly contributes to plasticity at different stages of deformation and under different loading conditions, the predictive capabilities of the models ideally increases with the sophistication of the dislocation type description. In general, several possible decompositions of the

total dislocation density have been proposed. Among them, distinctions based on the polarity of the dislocations are often considered. Thus, statistical dislocations – for which the volume integral over all dislocations of the inner product between their Burgers vector and their line tangent is null – are referred to as non-polar dislocations and are distinguished from geometrically necessary dislocations (GNDs), or polar dislocations, whose presence produces a net Burgers vector [187, 53, 10] and that are required to accommodate lattice incompatibilities and misorientations [129, 74, 55, 6].

As a result, the dislocation density at a given material point in the Hybrid Model is decomposed into distinct populations, as shown in figure 6.1. First, each density ρ contains both polar (p) and non-polar (np) densities, i.e.:

$$\rho = \rho_p + \rho_{np} \tag{6.1}$$

Each non-polar and polar density further consists of both glissile g and stored s dislocations, which gives:

$$\begin{aligned} \rho_{np} &= \rho_{np,g} + \rho_{np,s} \\ \rho_p &= \rho_{p,g} + \rho_{p,s} \end{aligned} \tag{6.2}$$

In this work, glissile dislocations refer to the subset of dislocations whose Burgers vectors lie on the glide plane and can glide in unconstrained motion (e.g., forward and backward). Both actively gliding and inactive dislocations satisfying these constraints are glissile. Stored dislocations are constrained to move in one direction (e.g., stalled in a repulsive state, piled-up against an obstacle) or are sessile.

Each type of dislocation population density in equations (6.1) and (6.2) can be considered the sum of the same type of density present on all slip systems α, $\alpha = 1, ..., n$, where n is the number of slip systems, such that:

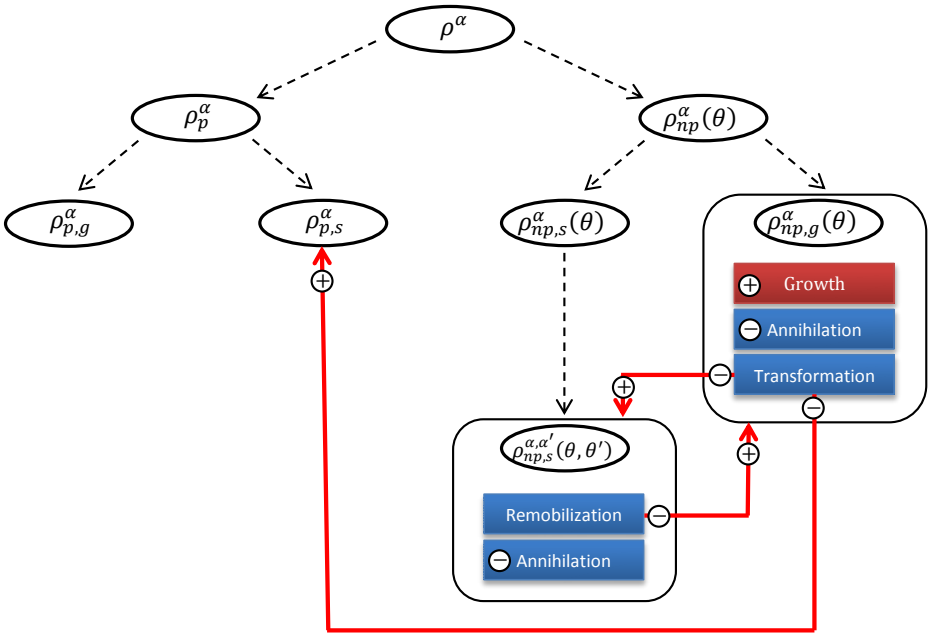


Figure 6.1: Dislocation decomposition and transformation mechanisms in the Hybrid Model.

$$\rho_i = \sum_{\alpha} \rho_i^{\alpha} \qquad \text{with } i = (p,g);(p,s);(np,g);(np,s) \tag{6.3}$$

where superscript α refers to the slip system. For instance, $\rho_{np,g}^{\alpha}$ refers to the non-polar, glissile density on system α.

6.3 Single-crystal kinematics

The relationship between the dislocation populations introduce in Section 6.2 and the deformation of the single crystal is discussed in this section. Following the analysis of Asaro and Rice [9], the total deformation gradient $\boldsymbol{F}$ relating the deformed configuration of a crystal to its undeformed reference state is decomposed as the product of an elastic and a plastic contribution:

$$\boldsymbol{F} = \boldsymbol{F}^{e} \cdot \boldsymbol{F}^{p} \tag{6.4}$$

where superscripts e and p refer to the elastic and plastic parts, respectively. While the elastic gradient $\boldsymbol{F}^e$ solely accounts for elastic stretches and contractions and rigid rotations of the reference crystal, the plastic part $\boldsymbol{F}^p$ describes the effects of the plastic activity (e.g. presence of dislocations) on the elastically undistorted reference configuration. Following the flow rule, the rate of change of the plastic deformation gradient is given by:

$$\boldsymbol{L}^p = \dot{\boldsymbol{F}}^p \cdot \boldsymbol{F}^{p-1} \tag{6.5}$$

When the plastic activity solely results from the motion of dislocations on their respective slip planes, the plastic velocity gradient $\boldsymbol{L}^p$ can be written as the sum of the contributions from dislocations of different polarities as:

$$\boldsymbol{L}^p = \boldsymbol{L}^p_{np} + \boldsymbol{L}^p_p \tag{6.6}$$

where $\boldsymbol{L}^p_{np}$ and $\boldsymbol{L}^p_p$ respectively denote the contributions from non-polar (statistical) and polar (geometrically necessary) dislocations. Considering statistical dislocations on all n slip systems of the crystal, tensor $\boldsymbol{L}^p_{np}$ is given as:

$$\boldsymbol{L}^p_{np} = \sum_{s=1}^{n} \dot{\gamma}^s \left(\vec{\boldsymbol{m}}^{\,s} \otimes \vec{\boldsymbol{n}}^{\,s}\right) \tag{6.7}$$

where the scalar quantity $\dot{\gamma}^s$ is the shear strain rate due to the motion of glissile non-polar dislocations on slip system s, and $\vec{\boldsymbol{m}}^{\,s}$ and $\vec{\boldsymbol{n}}^{\,s}$ denote the unit slip direction and unit normal to the slip plane. Note that equation (6.7) is based solely on kinematics and thus applies to both a discrete and a statistical description of dislocation-mediated plasticity. In the case of polar dislocations, $\boldsymbol{L}^p_p$ is expressed as:

$$\boldsymbol{L}^p_p = \sum_{s=1}^{n} \boldsymbol{\alpha}^s \times \vec{\boldsymbol{V}}^s \tag{6.8}$$

where $\boldsymbol{\alpha}^s$ is the dislocation density tensor (usually called the geometrically necessary dislocation tensor or Nye's tensor) associated with system s and $\vec{\boldsymbol{V}}^s$ is the velocity vector of dislocations on s. For the rest of this chapter, it is interesting to notice that dislocation density tensor $\boldsymbol{\alpha}^s$, whose definition is given in equation (3.60), vanishes for statistical dislocations when considered over a representative volume element in which their net Burgers is null [187, 131, 6]. In equation (6.8), the polar dislocations can be either glissile or stored.

In this framework, the Green-Lagrange tensor $\boldsymbol{E}^e$ can be taken as a measure of the elastic strain. Thus, it is defined from the elastic part of the deformation gradient as:

$$\boldsymbol{E}^e = \frac{1}{2}\left(\boldsymbol{F}^{eT} \cdot \boldsymbol{F}^e - \boldsymbol{I}_2\right) \tag{6.9}$$

where $\boldsymbol{I}_2$ denotes the second-order identity tensor. The second Piola–Kirchoff stress tensor $\boldsymbol{T}$ is related to the strain tensor via:

$$\boldsymbol{T} = \boldsymbol{C} : \boldsymbol{E}^e \tag{6.10}$$

where $\boldsymbol{C}$ denotes the fourth-order tensor of elastic moduli. From there, the symmetric Cauchy stress $\boldsymbol{\sigma}$ is obtained from the stress tensor $\boldsymbol{T}$ via the elastic deformation gradient as:

$$\boldsymbol{\sigma} = \det\left(\boldsymbol{F}^{eT}\right)^{-1} \boldsymbol{F}^e \cdot \boldsymbol{T} \cdot \boldsymbol{F}^{eT} \tag{6.11}$$

Further, at each step of the deformation, the crystal orientation in the current configuration is given by the rotation matrix $\boldsymbol{R}$ computed from the elastic deformation gradient as:

$$\boldsymbol{R} = \boldsymbol{F}^e \cdot \boldsymbol{R}_0 \tag{6.12}$$

where $\boldsymbol{R}_0$ is the initial rotation matrix given by the orientation of the crystal. Note that equation (6.12) holds when elastic strains are assumed to be small, which is generally the case in metals.

6.4 Work hardening modelling

Sections 6.2 and 6.3 provide a general framework, including both non-polar and polar dislocations, within which a constitutive law for single crystal deformation can be grounded. In this work, only the contribution from glissile and stored non-polar dislocations will be considered, as it is chosen to emphasize the coupling between DDD and dislocation density based models. In a general case, polar dislocations would have to be considered. However, in this work, the application of the current model is limited to the study of uniaxial tension of single crystals, such that it can be acknowledged that under those conditions, lattice rotations, which are accounted for in this formulation, will be homogenous. Hence no polar dislocations need to be generated.

The non-polar densities are assigned a line orientation, denoted by θ, which corresponds to the minimum angle between the dislocation line and its Burgers vector. As such, $\rho^{\alpha}_{np}(\theta)$ denotes the density of non-polar dislocations on system α with an orientation θ. The total non-polar density on system α, ρ^{α}_{np}, is thus given by:

$$\rho^{\alpha}_{np} = \int_0^{2\pi} \rho^{\alpha}_{np}(\theta) d\theta \tag{6.13}$$

Operations that govern the evolution of $\rho^{\alpha}_{np}(\theta)$ are shown on the right-hand side of figure 6.1. First, let us consider the glissile component of $\rho^{\alpha}_{np}(\theta)$. The glissile non-polar dislocation density $\rho^{\alpha}_{np,g}(\theta)$ increases in line length via dislocation glide (*growth* in figure 6.1). Glissile non-polar dislocations can also interact with other non-polar dislocations, those from other systems and with other orientations. These interactions lead to a loss in glissile dislocation density either via their annihilation with other

glissile dislocations or transformation into stored non-polar dislocations (see figure 6.1) and are dependent on the relative orientation of the two interacting dislocations. The latter transformation can happen, for instance, when a pair of dislocations form a junction or become locked in a repulsive state that prevents forward motion. The loss in non-polar glissile density via these orientation-dependent transformation interactions directly results in a gain (+) in non-polar stored density. Such glissile-stored transformations have been envisioned in the work of Roters and co-workers [206]. As shown in figure 6.1, orientation-dependent interactions can also lead to a loss (-) in stored non-polar dislocation density via annihilation with other stored dislocations and remobilization. The latter, remobilization, can occur in many ways. For instance, dislocations stored in repulsive and junction states can be remobilized by a simple change in the loading direction, such as in a Bauschinger test, or by a local fluctuation in the stress fields. Junctions can unzip, or a pair of dislocations in a repulsive state can move apart via a change in the sign of the resolved shear stress [166, 135]. The non-polar stored dislocation density $\rho^{\alpha}_{np,s}(\theta)$ is the outcome of all possible interactions with other dislocations, some leading to an increase (+) and other leading to a decrease (-), i.e. it can be written as:

$$\rho^{\alpha}_{np,s}(\theta) = \sum_{\alpha',\theta'} \rho^{\alpha,\alpha'+}_{np,s}(\theta,\theta') - \rho^{\alpha,\alpha'-}_{np,s}(\theta,\theta') \tag{6.14}$$

where $\rho^{\alpha,\alpha'}_{np,s}(\theta,\theta')$ is the density of non-polar stored dislocations on α with orientation θ due to an interaction with dislocations on system α' and with orientation θ'.

In the following, the growth, annihilation, and transformation operations involved in the evolution of non-polar glissile dislocations are discussed first (Section 6.4.1), followed by a similar description of the evolution of non-polar stored dislocations (Section 6.4.2) and the annihilation and *remobilization* interactions that are involved in equation (6.14). Hereinafter, glissile and stored dislocations are non-polar only.

6.4.1 Non-polar glissile dislocation density evolution

Each oriented non-polar glissile dislocation density $\rho^{\alpha}_{np,g}(\theta)$ evolves with the deformation as a result of glide and interactions with other dislocations. The incremental change in $\rho^{\alpha}_{np,g}(\theta)$ is expressed as:

$$d\rho^{\alpha}_{np,g}(\theta) = d\rho^{\alpha+}_{np,g}(\theta) - d\rho^{\alpha-}_{np,g}(\theta) \tag{6.15}$$

where superscripts + and − refer respectively to processes leading to an increase and decrease in the glissile line length. The processes involved in each are described in turn below.

6.4.1.1 Glissile line density increase

To evaluate the increase in the non-polar density $d\rho^{\alpha+}_{np,g}(\theta)$ for a given orientation, the following decomposition can be adopted:

$$d\rho^{\alpha+}_{np,g}(\theta) = \sum_{\alpha'} \frac{\partial \rho^{\alpha+}_{np,g}(\theta)}{\partial \gamma^{\alpha'}} d\gamma^{\alpha'} \tag{6.16}$$

where $\partial\gamma^{\alpha'}$ is the increment in slip with respect to time given by $\partial\gamma^{\alpha'} = \dot{\gamma}^{\alpha'}\partial t$. The slip rate $\dot{\gamma}^{\alpha}$ on any slip system α is directly related to the product of the density of glissile dislocations, the Burgers vector and the dislocation velocity according to Orowan's law, such that:

$$\dot{\gamma}^{\alpha} = \int_0^{2\pi} \rho^{\alpha}_{np,g}(\theta) b^{\alpha} \|\vec{\boldsymbol{V}}^{\alpha}(\theta)\| d\theta \tag{6.17}$$

where b^{α} is the magnitude of the Burgers vector and $\|\vec{\boldsymbol{V}}^{\alpha}(\theta)\|$ is the magnitude of the dislocation velocity directed normal to the loop.

Several processes can cause an increase in the glissile dislocation density $d\rho^{\alpha+}_{np,g}(\theta)$, such as line expansion by glide and remobilization of stored dislocations into glissile dislocations. The individual contributions from glide (glide) and remobilization

(remob) on α are additive such that:

$$\frac{\partial \rho_{np,g}^{\alpha+}(\theta)}{\partial \gamma^{\alpha'}} = \frac{\partial \rho_{np,g}^{\alpha+,\text{glide}}(\theta)}{\partial \gamma^{\alpha'}} + \frac{\partial \rho_{np,g}^{\alpha+,\text{remob}}(\theta)}{\partial \gamma^{\alpha'}} \tag{6.18}$$

Nucleation of glissile dislocations, propagation via cross-slip, and an increase in the number of sources via double cross-slip can also result in an increase in glissile line density. For simplicity, these are not treated, but could, in principle, be added later. The increase in line length due to dislocation glide $d\rho_{np,g}^{\alpha+,\text{glide}}(\theta)$, whose quantification is not is non-trivial, is discussed next in detail and that due to remobilization $d\rho_{np,g}^{\alpha+,\text{remob}}(\theta)$ will be discussed later in connection with the recovery of stored non-polar dislocations in Section 6.4.2.

In the following, the numerical quantification of the increase in non-polar dislocation line density is presented. In the current framework, the increment in the total glissile non-polar density on a due to glide $d\rho_{np,g}^{\alpha+,\text{glide}}$ is given by:

$$d\rho_{np,g}^{\alpha+,\text{glide}} = \int_0^{2\pi} d\rho_{np,g}^{\alpha+,\text{glide}}(\theta) d\theta \tag{6.19}$$

In order for the total $\rho_{np,g}^{\alpha}$ to remain non-polar as it grows, line continuity must be maintained, i.e., by definition, the corresponding dislocation density tensor $\boldsymbol{\alpha}^{\alpha}$ must vanish when averaged over a representative volume element. The line closure constraint leads to a coupling among the increments in all glissile, non-polar oriented dislocation densities $\rho_{np,g}^{\alpha}(\theta), \theta = \{0, ..., 2\pi\}$, belonging to slip system α. In the work of Arsenlis and Parks [7] and Cheong and Busso [44], the non-polar glissile density on each slip system is modelled as a square loop in which two opposing sides have pure screw characters and the other two sides pure edge. Propagation of one of the edge segments by Δx leads to an increase Δx in the length of both screw segments. Thus, the resulting increase in the dislocation density $\Delta\rho$ in a volume V containing this loop is equal to $\Delta\rho = 2\Delta x/V(m^{-2})$. In this picture, the loop is

discrete. However, the square loop can also be conceptually extended to represent the edge and screw dislocation densities on system α. In this sense, it becomes a virtual square dislocation. The relationship between the increments in the screw and edge dislocation densities on system α such that the line closure constraint is satisfied, can be determined analytically [7].

In the present approach, rather than constraining the line orientation on slip system a to solely edge and screw segments, a continuous description of dislocation orientations θ on slip system α is used. Thus, all the oriented densities $\rho^{\alpha}_{np,g}(\theta)$ composing the total density $\rho^{\alpha}_{np,g}$ on system α are represented by a single virtual dislocation loop. Each point along the loop represents a different oriented density $\rho^{\alpha}_{np,g}(\theta)$ with the same Burgers vector $\vec{\boldsymbol{b}}^{\alpha}$, but different θ and tangent vector $\vec{\boldsymbol{t}}(\theta)$. If slip system α is active and gliding, the total $\rho^{\alpha}_{np,g}$ will increase; however, densities of different orientation belonging to this system could grow at different rates, since some lines with specific orientations could be more likely to grow, transform, or annihilate than others (figure 6.1). Recall that, during growth, the change in the total $\rho^{\alpha}_{np,g}$ must be divergence free to ensure that the density remains non-polar; that is, for all deformation steps, the following circuit integral along a virtual loop for slip system α must be null:

$$\oint \vec{\boldsymbol{b}}^{\alpha} \cdot d\vec{\boldsymbol{t}}(\theta) = 0 \tag{6.20}$$

The above condition can be inferred from the definition of the Nye's tensor $\boldsymbol{\alpha}^{\alpha}$ in equations (3.60) to (3.62) given as a function of the Burgers vector. More generally, it can also be noticed that any non divergent-free change in $\rho^{\alpha}_{np,g}$ can be interpreted as a transformation of statistical dislocations into polar (GNDs) densities. Accordingly, in order to maintain line continuity, a change in a θ-oriented density $d\rho^{\alpha+,\text{glide}}_{np,g}(\theta)$ due to glide may necessitate changes in others θ'-oriented densities $d\rho^{\alpha+,\text{glide}}_{np,g}(\theta')$ belonging to α. In summary, representation of the non-polar glissile dislocation content on a given

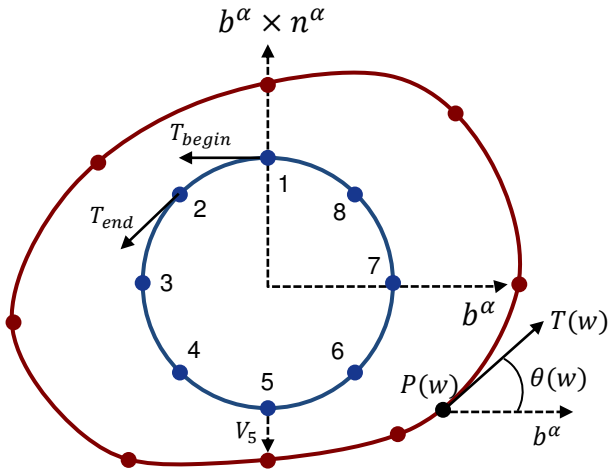


Figure 6.2: Virtual dislocation loop discretization. The blue loop is the initial loop and the red loop is the loop after some deformation. Note that the loop remains closed but changes in shape.

system α by use of an oriented density $\rho^{\alpha}_{np,g}(\theta)$ poses the problem of the quantification of $d\rho^{\alpha+,\text{glide}}_{np,g}(\theta)$ for all θ, such that line continuity is ensured and Burgers vectors are conserved among all oriented densities belonging to slip system α.

In the case where cubic splines are used to define dislocation loops so as to obtain a continuous description of orientation θ, no analytical solution can be found to determine the increment of dislocation density due to the propagation of a segment line, as opposed to what can easily be done when dealing with only pure edge/screw densities. To overcome this problem, a method based on using DDD is developed in order to calculate $d\rho^{\alpha+,\text{glide}}_{np,g}(\theta)$ due to glide under the constraint imposed by line closure, thereby alleviating the difficulty of deriving an analytical expression.

Figure 6.2 illustrates the virtual loop representation of the glissile line oriented density on a given slip system α. The initial state is presented by the blue loop. Its circular shape signifies that on system α the initial configuration is given by the uniform distribution $\rho^{\alpha}_{np,g}(\theta) = \rho^{\alpha}_{np,g}/2\pi$ for all $\theta \in [0, 2\pi]$, which models an initially random distribution of a large number of Frank-Read sources of equal length.

Due to glide, the virtual loop changes shape from the blue loop to the red loop.

To calculate this expansion, a nodal DDD approach in which each virtual dislocation loop is discretized into N_s splines connected to dislocation nodes is employed. To this end, the regular DDD code developed in Chapter 2 is used to treat the evolution of close dislocation loops representing the densities distribution on each slip system α. Because the loop represents the non-polar densities, it must remain closed at all times in the DDD code. To this end, the formulation employed in the DDD code used here only treats closed loops. For the sake of clarity, at each step, the DDD code computes the new position of the dislocations nodes, but the connection between the nodes is not altered, such that the loops remain closed as they were in the previous step. Each position along the splines represents an oriented dislocation density $\rho^{\alpha}_{np,g}(\theta)$ and a change in the length of the spline corresponds to a non-zero $d\rho^{\alpha+,\text{glide}}_{np,g}(\theta)$ for all θ along the spline. The nodal velocities are governed by the over-damped equation of motion (2.49) whose variational form given by equation (2.50) is expressed as:

$$\oint \delta\vec{\boldsymbol{x}} \cdot (\vec{\boldsymbol{f}}^{pk} - \boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}}) \, |d\vec{\boldsymbol{x}}| = 0 \tag{6.21}$$

where the integral is carried out along the closed circuit of the dislocation loop, $\vec{\boldsymbol{v}}$ is the velocity vector of the dislocation that acts normal to the dislocation line, $\boldsymbol{\mathcal{B}}$ is the drag coefficient matrix for which a simple expression is given in equation (C.46), and $\delta\vec{\boldsymbol{x}}$ is the virtual displacement of line $d\vec{\boldsymbol{x}}$. For more details on the variational approach to dislocation motion, the reader is referred to Appendix C.1. The Peach-Koehler force $\vec{\boldsymbol{f}}^{pk}$ is the force exerted on each dislocation density (represented by line $d\vec{\boldsymbol{x}}$) by the macroscopic stress $\boldsymbol{\sigma}$ and defined from (2.46) as:

$$\vec{\boldsymbol{f}}^{pk} = \left(\boldsymbol{\sigma} \cdot \vec{\boldsymbol{b}}^{\alpha}\right) \times \vec{\boldsymbol{t}}(\theta) \tag{6.22}$$

As such, the driving force of dislocation motion does not directly account for the stress field resulting from interactions with the loop itself and other loops, since the

loop is a virtual representation, not an actual discrete dislocation loop. Instead, the influence of the surrounding dislocation network on the velocity for a given $\rho^{\alpha}_{np,g}(\theta)$ is accounted for in each deformation step by transformation processes as described in the next paragraph. The velocity for each density is then used to calculate the slip rates, strain rates, and rates of annihilation, transformation, and remobilization according to a set of evolution equations, described hereinafter and in the glissile line removal paragraph.

In the DDD calculation in which the dislocation loop illustrated in figure 6.2 is discretized into a finite number of segments, the shape of each segment line ij (i.e. position and tangent) describing each virtual loop can be represented via the use of the generalized coordinates $\vec{\boldsymbol{q}}_{ij}$ introduced in equations (2.28)–(2.32) [96, 97]. For more details on discrete nodal dislocation representation, please refer to Section 2.3.1. In this work, a cubic spline representation for the virtual loop is used, and the position $\vec{\boldsymbol{x}}_{ij}$ and tangent vector $\vec{\boldsymbol{t}}_{ij}$ of each segment ij are thus given by equations (2.28) and (2.29), respectively.

To illustrate the passage from a loop representation to a dislocation density representation, consider again the two virtual loops in figure 6.2; one shown in blue and the other in red. The orientation of each segment is related to the tangent vector by:

$$\theta(s) = \arccos\left(\frac{\vec{\boldsymbol{b}}^{\alpha}}{\|\vec{\boldsymbol{b}}^{\alpha}\|} \cdot \vec{\boldsymbol{t}}(s)\right) \tag{6.23}$$

where s is the curvilinear coordinate along the spline. The total glissile dislocation density on system α, $\rho^{\alpha}_{np,g}$, is the integral of the position vector $\vec{\boldsymbol{x}}$ over the entire loop. From this, the oriented dislocation density is given by:

$$\rho^{\alpha}_{np,g}(\theta) = \rho^{\alpha}_{np,g} \cdot f(\theta) \tag{6.24}$$

where $\rho^{\alpha}_{np,g}$ corresponding to the line length of virtual loop α is given from equation

(2.32) as:

$$\rho^{\alpha}_{np,g} = \sum_{ij}^{N_s} \int_0^1 \left(\sum_{a,b=1}^{4} \vec{\boldsymbol{q}}^{a}_{ij} \cdot N'_a(s) N'_b(s) \cdot \vec{\boldsymbol{q}}^{b}_{ij} \right)^{1/2} ds \tag{6.25}$$

where the derivatives of the shape functions $N'_a(s)$ for cubic splines are expressed in equation (2.31), and where $f(\theta)$ denotes the distribution function of θ, which can be obtained by a least-square fit to essentially invert system (2.30).

At each step, the equation of motion (6.21) allows for the determination of the distribution of the velocity vectors $\vec{\boldsymbol{v}}^{\alpha}(\theta)$ for all orientations θ belonging to a single virtual loop for a given slip system α (see Section 2.5 for more details on the numerical evaluation of nodal velocities). This velocity is then updated to $\vec{\boldsymbol{V}}^{\alpha}(\theta)$ to account for the influence of the current dislocation (obstacle) network via the rate at which dislocations can overcome obstacles via thermal activation. In this case, $\vec{\boldsymbol{V}}^{\alpha}(\theta)$ is generally expressed as a product of an average rate term and the temperature T dependent probability $P^{\alpha}(T)$ of overcoming an obstacle by thermal activation:

$$\vec{\boldsymbol{V}}^{\alpha}(\theta) = \vec{\boldsymbol{v}}^{\alpha}(\theta) \cdot P^{\alpha}(T) \tag{6.26}$$

The probability $P^{\alpha}(T)$ for thermally activated dislocation motion most often is modelled using Arrhenius-law, for which several expressions have been proposed [126, 33, 127, 44, 45, 162]. In this work, the thermally activated glide probability model as suggested by Busso [33] is adopted, and its expression reads:

$$P^{\alpha}(T) = \exp\left(-\frac{F_0}{kT} \left\langle 1 - \left\langle \frac{\tau^{\alpha}_{eff}}{\tau^{\alpha}_{crit}} \right\rangle^{p} \right\rangle^{q} \right) \tag{6.27}$$

where k and F_0 denote Bolzmann's constant and the Helmholtz free activation energy, exponents $p \in [0, 1]$ and $q \in [1, 2]$ characterize the shape of the activation energy profile [127], and function $\langle x \rangle = \max(x, 0)$.

The basic phenomenological arguments directly follow from the work of Kocks and

co-workers [126, 127]. The stress terms τ^{α}_{eff} and τ^{α}_{crit} denote respectively the effective driving stress and the maximum strength at which dislocations can be mobilized without the assistance of thermal activation. These two terms have been described in several ways. For instance, in Busso [33] and Cheong and co-workers [44, 45], τ^{α}_{eff} is set as the difference between the resolved shear stress τ^{α}_{res} and a back-stress term related to the density of stored dislocations. Alternatively, one can relate both the back-stresses and the critical resolved shear stress to different dislocation populations, such as in Ma et al. [162], who related dislocations parallel to the slip system to back stresses and forest dislocations that cut-through the slip system to the critical resolved shear stress. Following these ideas, in this work, the driving stress τ^{α}_{eff} is simply defined as:

$$\tau^{\alpha}_{eff} = |\tau^{\alpha}_{res}| - S^{\alpha} \tag{6.28}$$

where S^{α} is the athermal forest resistance. Both τ^{α}_{crit} and S^{α} at the deformation temperature are scaled from their values at 0 K, τ_0 and S^{α}_0, respectively, and the ratio of the shear moduli at the deformation temperature to 0 K, i.e.:

$$\begin{aligned} \tau^{\alpha}_{crit} &= \tau_0 \frac{\mu}{\mu_0} \\ S^{\alpha} &= S^{\alpha}_0 \frac{\mu}{\mu_0} \end{aligned} \tag{6.29}$$

For S^{α}_0, the Fransiosi and Zaoui extension [90] of the Taylor formulation to multi-slip systems is used, such that:

$$S^{\alpha}_0 = \mu b \sqrt{\sum_{\alpha'} a^{\alpha\alpha'} \rho^{\alpha'}_{np,s}} \tag{6.30}$$

where $\rho^{\alpha'}_{np,s}$ denotes the non-polar stored dislocations on system α' that contribute to slip resistance via the strength-inter-action matrix $a^{\alpha\alpha'}$.

6.4.1.2 Glissile line density removal

Glissile line density removal $d\rho_{np,g}^{\alpha-}$ occurs by either annihilation between glissile dislocations, transformation of glissile dislocations into stored dislocations via junction formation or repulsion between glissile dislocations. Whether two glissile dislocations repel, form a junction, annihilate, or none of the above depends on their relative orientation. The sum of the individual contributions from different interactions and all possible paired interactions between system α and system α' yields an expression for $d\rho_{np,g}^{\alpha-}(\theta)$:

$$d\rho_{np,g}^{\alpha-}(\theta) = \sum_{\alpha',\theta'} d\rho_{np,g}^{\alpha,\alpha'-}(\theta,\theta') \tag{6.31}$$

where $d\rho_{np,g}^{\alpha,\alpha'-}(\theta,\theta')$ describes the density of glissile dislocations on system α with orientation θ that are transformed via interactions with all dislocations on system α'. It is the sum of the contributions of junction formation, repulsive interactions and annihilation:

$$d\rho_{np,g}^{\alpha,\alpha'-}(\theta,\theta') = d\rho_{np,g,jun}^{\alpha,\alpha'-}(\theta,\theta') + d\rho_{np,g,ann}^{\alpha,\alpha'-}(\theta,\theta') + d\rho_{np,g,int}^{\alpha,\alpha'-}(\theta,\theta') \tag{6.32}$$

where the subscripts refer to the end states, that is, jun, ann, or int for junction, annihilation and immobilization by repulsion, respectively. Note that each term in equation (6.32) is generally not a symmetric function of α and α'. The increment $d\rho_{np,g,end}^{\alpha,\alpha'-}(\theta,\theta')$ for each end state can be seen to generally follow:

$$d\rho_{np,g,end}^{\alpha,\alpha'-}(\theta,\theta') = H_{end}^{\alpha\alpha'}(\theta,\theta')\frac{L^{\alpha\alpha'}(\theta,\theta')U(\theta,\theta')dN^{\alpha\alpha'}}{V} \tag{6.33}$$

where the transformation-interaction matrix $H_{end}^{\alpha\alpha'}(\theta,\theta')$ is zero or unity depending on whether the interaction leads to the corresponding end-state, $end = \{jun, ann, int\}$. For example, $H_{jun}^{\alpha\alpha'}(\theta,\theta')$ for equals unity if the interaction leads to a junction and

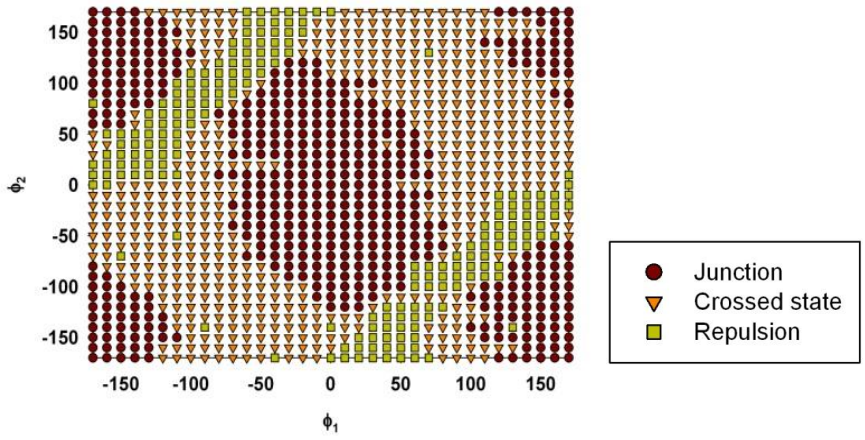


Figure 6.3: Example of a junction formation map for the intersection of two prismatic dislocations in HCP Zr. The map is computed by DDD following the method used in reference [39]. Such a map depicts the nature of the intersection event (junction formation, crossed state, or repulsion) as a function of the orientation of the two interacting dislocations (where ϕ_i denotes the angle between dislocation line i and its Burgers vector).

equals zero otherwise. Similarly for $H_{ann}^{\alpha\alpha'}(\theta,\theta')$ or $H_{int}^{\alpha\alpha'}(\theta,\theta')$, which take on value of 1 if the interaction leads to annihilation or a repulsive state and zero if not. Maps that define the outcomes of paired $\alpha-\alpha'$ dislocation-dislocation interactions in (θ,θ') space have been calculated by DDD [164, 165] for FCC crystals. As an example, figure 6.3 shows a map that indicates whether the interaction of two dislocations with different orientations form a junction or attain a repulsive state or neither (*cross-state*). Such maps have been calculated for Cu [135], Mg [39], and Zr [176].

The fraction on the right-hand side of equation (6.33) is the total line length, given by $L^{\alpha\alpha'}(\theta,\theta')U(\theta,\theta')dN^{\alpha\alpha'}$, per unit volume V involved in the interaction. $L^{\alpha\alpha'}(\theta,\theta')$ is the dislocation length involved in the interaction between two glissile segments of orientation θ and θ' and belonging to system α and α', $U(\theta,\theta')$ is the fraction of $L^{\alpha\alpha'}(\theta,\theta')$ that is transformed, and $dN^{\alpha\alpha'}$ is the number of interactions events that occurred within the increment. $L^{\alpha\alpha'}(\theta,\theta')$ can be approximated as to be proportional to the length of the initially shorter segment or the smaller of the two populations involved, such that one can write $L^{\alpha\alpha'}(\theta,\theta')/V = k\cdot\min\left(\rho_{np,g}^{\alpha}(\theta),\rho_{np,g}^{\alpha'}(\theta')\right)$, where k

is a fitting parameter. The fraction $U(\theta, \theta')$ has been observed in DDD simulations to decrease with an increase in misorientation angle of the two initial segments [164]. Therefore, as a first approximation, $U(\theta, \theta')$ can be expressed as a simple function of the misorientation between the lines, e.g. $U(\theta, \theta') = |\cos(\theta - \theta')|$. As a result, the increment $d\rho^{\alpha,\alpha'-}_{np,g,end}(\theta, \theta')$ for each end state can be expressed as:

$$d\rho^{\alpha,\alpha'-}_{np,g,end} = H^{\alpha\alpha'}_{end} \cdot k \left|\cos(\theta - \theta')\right| \cdot \min\left(\rho^{\alpha}_{np,g}(\theta), \rho^{\alpha'}_{np,g}(\theta')\right) \cdot dN^{\alpha\alpha'} \tag{6.34}$$

As a first approximation, $dN^{\alpha\alpha'}$, corresponding to the number of interaction events between systems α and α', can be given as:

$$dN^{\alpha\alpha'} = k_0 \sqrt{\sum_{\alpha''} d^{\alpha'\alpha''}(\rho^{\alpha''}_{np,g} + \rho^{\alpha''}_{np,s})} \cdot |d\gamma^{\alpha}| = \frac{k_0}{\lambda^{\alpha'}} |d\gamma^{\alpha}| \tag{6.35}$$

where $k_0 |d\gamma^{\alpha}|$ corresponds to the average distance travelled by dislocations on a due to the slip on that system, and

$$\lambda^{\alpha'} = \frac{1}{\sqrt{\sum_{\alpha''} d^{\alpha'\alpha''}(\rho^{\alpha''}_{np,g} + \rho^{\alpha''}_{np,s})}} \tag{6.36}$$

is the mean free path on system α', i.e. the average distance a glissile dislocation is likely to travel before being stopped by another dislocation (either glissile or stored). The matrix $d^{\alpha'\alpha''}$ is a *length-interaction* matrix, which engenders a dependence of the mean free path on the two slip systems that are interacting. The interaction is considered to encompass glissile-glissile and glissile-stored interactions. Generally, the stronger the interaction is, the smaller the mean free path will be. Substituting equation (6.35) into (6.50) introduces a material parameter $k_1 = k \cdot k_0$, which is synonymous with the coefficient for the rate of trapping of glissile dislocations or rate of storage of dislocations in other models [127, 25, 132].

Notice that with equations (6.31)-(6.35), it is possible that, as the overall dislocation density on all systems becomes so large, the fraction of glissile dislocations transformed into stored dislocation can reach one and thus the rate of change in the glissile dislocation density zero.

6.4.2 Non-polar stored dislocation density evolution

As for the case of glissile densities, the evolution of non-polar stored dislocation density on slip system α of a given orientation h results from the interactions with other densities on other slip systems and orientations:

$$d\rho^{\alpha}_{np,s}(\theta) = \sum_{\alpha',\theta'} d\rho^{\alpha,\alpha'}_{np,s}(\theta,\theta') \tag{6.37}$$

The non-polar stored dislocation intersection density evolves via two main phenomena: (1) junction formation and (2) repulsive interactions. The strength of both junction and repulsive states are dependent on the relative orientations of the two interacting segments. Here, it is proposed to formally account for the evolution of each as a function of their orientation relationship (θ,θ'). Accordingly, the following decomposition is used:

$$d\rho^{\alpha,\alpha'}_{np,s}(\theta,\theta') = d\rho^{\alpha,\alpha'}_{np,s,jun}(\theta,\theta') + d\rho^{\alpha,\alpha'}_{np,s,int}(\theta,\theta') \tag{6.38}$$

where the subscripts j and int refer to junction and repulsive interactions, respectively. The evolution of each is governed by a balance of storage and removal rates:

$$\begin{aligned} d\rho^{\alpha,\alpha'}_{np,s,jun}(\theta,\theta') &= d\rho^{\alpha,\alpha'+}_{np,s,jun}(\theta,\theta') - d\rho^{\alpha,\alpha' recov}_{np,s,jun}(\theta,\theta') - d\rho^{\alpha,\alpha' remob}_{np,s,jun}(\theta,\theta') \\ d\rho^{\alpha,\alpha'}_{np,s,int}(\theta,\theta') &= d\rho^{\alpha,\alpha'+}_{np,s,int}(\theta,\theta') - d\rho^{\alpha,\alpha' recov}_{np,s,int}(\theta,\theta') - d\rho^{\alpha,\alpha' remob}_{np,s,int}(\theta,\theta') \end{aligned} \tag{6.39}$$

In the above, the first terms with superscript + in each evolution equation refer to the rates of increase in storage due to junction (jun) or repulsive interactions (int).

Via the transformation mechanism proposed in this work, they also correspond to the decrement in the glissile dislocation density in equation (6.32). As a result, the transformation mechanisms from glissile to stored densities write:

$$d\rho^{\alpha,\alpha'+}_{np,s,jun}(\theta,\theta') = d\rho^{\alpha,\alpha'-}_{np,g,jun}(\theta,\theta')$$
$$d\rho^{\alpha,\alpha'+}_{np,s,int}(\theta,\theta') = d\rho^{\alpha,\alpha'-}_{np,g,int}(\theta,\theta') \tag{6.40}$$

These mechanisms lead to stage II hardening. The second terms in equation (6.39) with superscript *recov* represent the decrease due to line removal by dynamic recovery, which is characteristic of stage III. To model dynamic recovery, we employ the model suggested by Kocks and Mecking [126, 127]:

$$d\rho^{\alpha,\alpha' recov}_{np,s,jun}(\theta,\theta') = k_2 \cdot \rho^{\alpha,\alpha'}_{np,s,jun}(\theta,\theta') \cdot |d\gamma^{\alpha}|$$
$$d\rho^{\alpha,\alpha' recov}_{np,s,int}(\theta,\theta') = k_2 \cdot \rho^{\alpha,\alpha'}_{np,s,int}(\theta,\theta') \cdot |d\gamma^{\alpha}| \tag{6.41}$$

where k_2 depends on strain rate and temperature [25]. The example later considers a fixed temperature and strain rate and thus, k_2 is treated as a fitting parameter. Note that with the present formulation, k_2 is a single parameter valid for all slip systems α, whereas it is usually treated as a mode-dependent parameter k_2^{α} in current constitutive models (e.g. in reference [25]). As before, $|d\gamma^{\alpha}|$ is the increment of shear due to gliding on system α.

The last terms in equation (6.39) with superscript *remob* represent the loss in stored non-polar density due to remobilization and hence correspond to a gain in the glissile non-polar density, such that:

$$d\rho^{\alpha+,remob}_{np,g}(\theta) = \sum_{\alpha',\theta'} \left[d\rho^{\alpha,\alpha' remob}_{np,s,jun}(\theta,\theta') + d\rho^{\alpha,\alpha' remob}_{np,s,int}(\theta,\theta') \right] \tag{6.42}$$

As mentioned earlier, remobilization corresponds to the unzipping of junctions or

unlocking of repulsive states due to a change in local stress state. As such, remobilization is particularly important and non-negligible in the case of complex loading states or when undergoing a change in loading path. The rate of unzipping depends on the local stress state relative to the strength of the junction. The junction strength is a function of the type of junction (e.g. Lomer-Cottrell lock vs Hirth lock), junction length, and line misorientation. In the case of FCC metals, junction strength decreases with junction line length and in turn, the length of the junction formed decreases with increasing line misorientation. Further for HCP metals, as shown in DDD simulations on Mg single crystals, the junction strength can exhibit significant anisotropy [39].

To capture these effects, the rate of remobilization is expressed as the product of the current dislocation junction density, the fraction that is remobilized $U(\theta, \theta')$, and the probability of junction unzipping $Y(\tau^{\alpha,\alpha'})$ given the junction type and the current resolved shear stress:

$$d\rho_{np,s,jun}^{\alpha,\alpha' remob}(\theta, \theta') = \rho_{np,s,jun}^{\alpha,\alpha'}(\theta, \theta') U(\theta, \theta') Y(\tau^{\alpha}, \tau^{\alpha'}) \tag{6.43}$$

where τ^{α} and $\tau^{\alpha'}$ respectively denote the resolved shear stresses on α an α'. $Y(\tau^{\alpha}, \tau^{\alpha'})$ is defined such that it equals unity if the stress state exceeds the strength of a given junction and zero otherwise. The non-polar dislocations stored as result of repulsive interaction can be remobilized via a simple change in the load direction (at the local scale). Similar to equation (6.43), the rate of remobilization of repulsive states is:

$$d\rho_{np,s,int}^{\alpha,\alpha' remob}(\theta, \theta') = \rho_{np,s,int}^{\alpha,\alpha'}(\theta, \theta') U(\theta, \theta') S(\tau^{\alpha}, \tau^{\alpha'}) \tag{6.44}$$

where $S(\tau^{\alpha}, \tau^{\alpha'})$ takes the value one if either resolved shear stresses on α or α' changes in sign. Both $Y(\tau^{\alpha}, \tau^{\alpha'})$ and $S(\tau^{\alpha}, \tau^{\alpha'})$ can, in principle, be determined from strength-interaction maps, such as the one in figure 6.3, obtained from DDD calculations.

Sections 6.2 to 6.4 provide the main constitutive equations of the Hybrid Model. In the following, the numerical implementation of the model is discussed and a simple application on uniaxial tension on FCC Copper is presented to highlight the capabilities of the model. Further, note that a simplified version of the model will be used and its constitutive equations will be presented in Section 6.4.3.

6.4.3 Simplified formulation

In Sections 6.4.1 and 6.4.2, the complete set of evolution and transformation laws were given for non-polar glissile and stored dislocations as a function of their orientation. However, there may be some situations in which a simplification to the model is desired and the degrees of freedom associated with all possible orientation pairs (θ, θ') can be reduced to consider no orientation dependence.

On top of providing a lighter formulation of the Hybrid Model that should allow to grasp the essence of the model more directly than from the previous sections, the removal of densities orientations also alleviate numerical difficulties (see Section 6.5.1) by restraining the shape of the virtual loops (see figure 6.2) to perfect circles. Further, if one restricts this simplified version to be used in monotonic loading conditions, such as done for single crystal aluminium in Section 6.6, then no remobilization mechanism needs to be considered.

As a result, the simplified version of the model solely accounts for the modelling of non-polar glissile and stored overall densities per system based on the Kocks-Mecking approach [127]. Thus, the various mechanisms affecting their evolution, that may either lead to an increase (+) or a decrease (-) in the dislocation densities, can be described in a simpler fashion by the following set of rate laws for each system s:

$$\text{Evolution of glissile } (g) \text{ densities:} \qquad \dot{\rho}^{\alpha}_{np,g} = \dot{\rho}^{\alpha+}_{np,g} - \dot{\rho}^{\alpha-}_{np,g} \tag{6.45}$$

$$\text{Transformation (trapping):} \qquad \dot{\rho}^{\alpha-}_{np,g} = \dot{\rho}^{\alpha+}_{np,s} \tag{6.46}$$

$$\text{Evolution of stored } (s) \text{ densities:} \qquad \dot{\rho}^{\alpha}_{np,s} = \dot{\rho}^{\alpha+}_{np,s} - \dot{\rho}^{\alpha-}_{np,s} \tag{6.47}$$

in which the increase $d\rho^{\alpha+}_{np,g}$ of glissile dislocations due to glide is quantified using circular virtual loops. Further, in this case, the loss in non-polar glissile density via the transformation mechanisms introduced in relations (6.40) directly results in the gain in non-polar stored density. Their quantification, corresponding to equation (6.31) in the complete version, reduces to:

$$d\rho^{\alpha-}_{np,g} = d\rho^{\alpha+}_{np,s} = \sum_{end} \sum_{\alpha'} d\rho^{\alpha,\alpha'-}_{np,g,end} \tag{6.48}$$

where $d\rho^{\alpha,\alpha'-}_{np,g,end}$ denotes the increment of transformed glissile non-polar densities stored on system α due to the interaction with system α' and for $end = \{jun, ann, int\}$. Following expression (6.33), it is here simply given by:

$$d\rho^{\alpha,\alpha'-}_{np,g,end} = P^{\alpha\alpha'}_{end} \cdot \frac{L^{\alpha\alpha'} U dN^{\alpha\alpha'}}{V} \tag{6.49}$$

where $P^{\alpha\alpha'}_{end}$ is the probability that the intersection between lines due to interaction between systems α and α' results in a given end stage, i.e. $end = \{jun, ann, int\}$. $P^{\alpha\alpha'}_{end}$ can be obtained by integration of the transformation-interaction matrix $H^{\alpha\alpha'}_{end}(\theta, \theta')$ over the (θ, θ') space. The ratio $L^{\alpha\alpha'}U/V$ corresponds to the average length of the junction to be formed per unit volume when system α interacts with system α'. It is proportional to the lesser glissile dislocation density between system α and α', i.e. $\min\left(\rho^{\alpha}_{np,g}, \rho^{\alpha'}_{np,g}\right)$, such that:

$$d\rho^{\alpha,\alpha'-}_{np,g} = P^{\alpha\alpha'} \cdot k \cdot \min\left(\rho^{\alpha}_{np,g}, \rho^{\alpha'}_{np,g}\right) \cdot dN^{\alpha\alpha'} \tag{6.50}$$

Finally, the Kocks-Mecking approach for dynamic recovery [127] is employed to model the rate of annihilations of stored dislocations as:

$$d\rho_{np,s}^{\alpha-} = k_2 \cdot \rho_{np,s}^{\alpha-} \left| d\gamma^{\alpha} \right| \tag{6.51}$$

6.5 Numerical implementation

In this section, the numerical implementation of the Hybrid Model is briefly discussed. First, the representation of virtual loops using the DDD code is presented for FCC crystals, after which the coupling of the crystal plasticity framework with the finite element method is detailed.

6.5.1 Virtual loops

As introduced in Section 6.4.1, the concept of a virtual loop is used to calculate $d\rho_{np,g}^{\alpha+,glide}(\theta)$, the increase in the non-polar glissile dislocation density population of orientation θ due to glide. In the case of FCC crystals, the 12 $\langle 110 \rangle \{111\}$ are doubled. Practically, two polarized virtual loops with opposite Burgers vectors (+/-) are associated to each slip system α to ensure that growth of virtual loops is granted independently of the sign of the applied stress. Table 6.1 lists the 24 virtual slip systems used in this work along with their Schmid and Boas (SB) notation. The (+) and (-) in SB denotes the sign of the Burgers vector, the (+) being the conventionally adopted one in the literature.

Initially, each virtual loop is perfectly circular, i.e., the densities in each orientation are identical (see for instance, the blue loop in figure 6.2). This is intended to represent an initially random distribution of a large number of Frank-Read sources. At each deformation step, only the systems whose virtual loop length has increased are updated. Those systems whose loop length would shrink are forced to remain fixed in their previous position as if their average velocity is zero. Nonetheless, these zero-velocity loops do not contribute to the plastic shearing rate (see equation (6.17)).

Table 6.1: Properties and notation for the 24 virtual loops used in the model.

Slip plane	$\vec{\boldsymbol{b}}^{\alpha}$	Index	SB	$\vec{\boldsymbol{b}}^{\alpha}$	Index	SB
(111)	$\frac{1}{2}[\bar{1}10]$	1	B5+	$\frac{1}{2}[1\bar{1}0]$	13	B5-
(111)	$\frac{1}{2}[\bar{1}01]$	2	B4+	$\frac{1}{2}[10\bar{1}]$	14	B4-
(111)	$\frac{1}{2}[0\bar{1}1]$	3	B2+	$\frac{1}{2}[01\bar{1}]$	15	B2-
$(1\bar{1}1)$	$\frac{1}{2}[110]$	4	D6+	$\frac{1}{2}[\bar{1}\bar{1}0]$	16	D6-
$(1\bar{1}1)$	$\frac{1}{2}[011]$	5	D1+	$\frac{1}{2}[0\bar{1}\bar{1}]$	17	D1-
$(1\bar{1}1)$	$\frac{1}{2}[\bar{1}01]$	6	D4+	$\frac{1}{2}[10\bar{1}]$	18	D4-
$(11\bar{1})$	$\frac{1}{2}[\bar{1}10]$	7	C5+	$\frac{1}{2}[1\bar{1}0]$	19	C5-
$(11\bar{1})$	$\frac{1}{2}[101]$	8	C3+	$\frac{1}{2}[\bar{1}0\bar{1}]$	20	C3-
$(11\bar{1})$	$\frac{1}{2}[011]$	9	C1+	$\frac{1}{2}[0\bar{1}\bar{1}]$	21	C1-
$(\bar{1}11)$	$\frac{1}{2}[110]$	10	A6+	$\frac{1}{2}[\bar{1}\bar{1}0]$	22	A6-
$(\bar{1}11)$	$\frac{1}{2}[0\bar{1}1]$	11	A2+	$\frac{1}{2}[01\bar{1}]$	23	A2-
$(\bar{1}11)$	$\frac{1}{2}[101]$	12	A3+	$\frac{1}{2}[\bar{1}0\bar{1}]$	24	A3-

For loops associated with active slip systems, the loops will expand and deform, and will not necessarily remain perfectly circular, as suggested by the red loop in figure 6.2, since some lines with specific orientations could be more likely to grow, transform, or annihilate than others. Once the new non-polar glissile densities $\rho^{\alpha}_{np,g}(\theta)$ have been computed for all orientations θ – based on the rates of annihilation and transformation–, the virtual loops must be reconstructed accordingly. During reconstruction, convexity and closure conditions of the loops are enforced such that (i) the density of each orientation can be uniquely retrieved and (ii) virtual loops represent the glissile non-polar dislocations densities. Therefore the passage between the virtual loop representation and the non-polar glissile dislocation densities proceeds in three steps: (i) the increase in glissile densities $d\rho^{\alpha+,glide}_{np,g}(\theta)$ due to glide is computed as described in Section 6.4.1, (ii) the removal $d\rho^{\alpha-}_{np,g}(\theta)$ due to interactions is computed using equations (6.31)–(6.35) and (iii) the virtual loops are reconstructed from the new densities $s\rho^{\alpha}_{np,g}(\theta)$ (see equation (6.15)).

In the case of the simplification introduced through equation (6.48) and (6.50) in

which line orientation dependence is removed, the virtual loops will remain perfectly circular as they grow. This simplification leads to a significant gain in computation time since (1) the suppression of orientation dependence enables us to work with only one density per slip system and per type, and (2) the number of discretization points required to describe the virtual loop can be reduced to a minimum.

Both approaches for updating $d\rho_{np,g}^{\alpha+,glide}(\theta)$ are employed in this work for the case of monotonic, uniaxial loading of a single crystal. In either case, an increase in the non-polar glissile density can be associated with the transformation of non-polar glissile densities into stored densities. The transformations are computed using the formulation expressed through equations (6.31), (6.33) and (6.35).

6.5.2 Crystal plasticity finite element coupling

Both the fully orientation-dependent and simplified orientation-independent versions have been implemented in a finite element (FE) framework such that equilibrium and boundary conditions are respected at each step. The general FEM framework to incorporate stress-strain formulations is presented in details in Appendix E. The algorithm for the fully orientation-dependent approach is provided in figure 6.4. The coupling between the FE and the DDD procedures is based on a two-step approach. First, the DDD code computes the evolution of the virtual loops solving the dynamics for a given stress state. The new values of every density population are then computed once interactions processes have been handled, yielding the plastic strain increment. Second, the FE code computes the displacement field making use of the plastic deformation gradient computed during the first step. This coupling is performed using an implicit scheme in which the stress state for an element at step t, $\boldsymbol{\sigma}(t)$, is obtained from the elastic deformation gradient $\boldsymbol{F}^e(t)$ according to equation (6.11). The latter is computed using the decomposition introduced in equation (6.4) where the total deformation gradient given by the imposed increment of deformation,

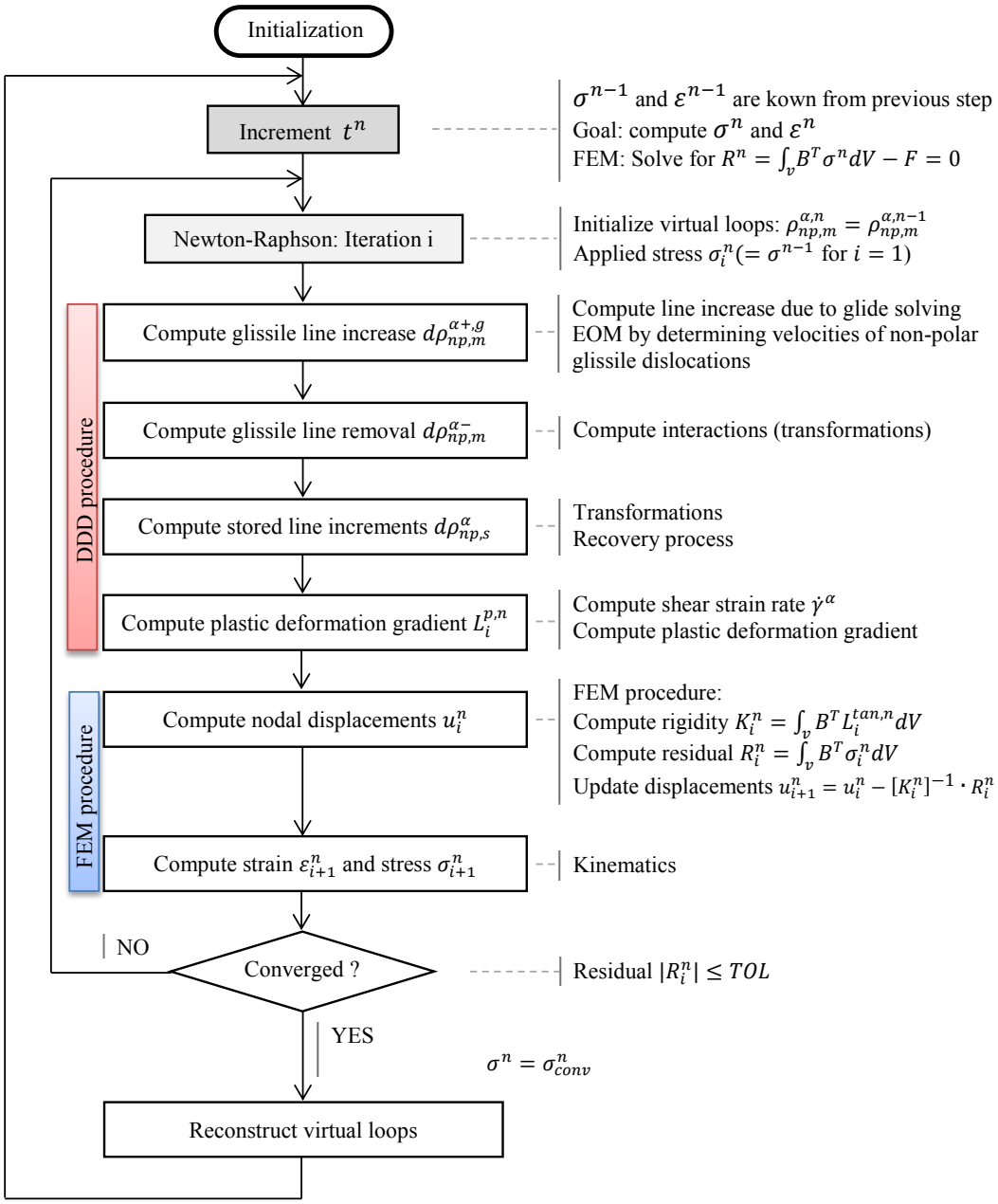


Figure 6.4: Flowchart of the coupling between DDD and FEM. Fundamental equation of non-linear FEM $\boldsymbol{R}^n = \int_V \boldsymbol{B}^T \boldsymbol{\sigma}^n dV - \boldsymbol{F} = \boldsymbol{0}$ directly follows from the principle of virtual work applied to a medium (V) subjected to displacement and stress boundary conditions (see Appendix E.4). Here $\boldsymbol{B}$ denotes the spatial partial derivatives matrix of the element discretized nodes and $\boldsymbol{\sigma}^n$ represents the stress at increment n given by equation (6.11). The use of an implicit iterative method (Newton-Raphson) to solve for the non-linear system $\boldsymbol{R}^n = \boldsymbol{0}$ requires the introduction of the tangent modulus (or material jacobian) $\boldsymbol{L}^{tan} = \partial \boldsymbol{\sigma} / \partial \boldsymbol{\epsilon}$.

and the plastic deformation gradient $\boldsymbol{F}^p(t)$ at the same step t using equation (6.7). A Newton–Raphson procedure is used to solve the non-linear dependence between the stress and the plastic strain. (The general non-linear FEM formulation for stress calculation is detailed in Appendix E.4.) Therefore, at the beginning of each increment t, the stress state $\boldsymbol{\sigma}(t)$ has to be defined everywhere in the FE mesh, as it constitutes an input for the next increment of the DDD code. As indicated by equations (6.21) and (6.22), $\boldsymbol{\sigma}(t)$ is the driving force used to determine the motion of the dislocation lines of the virtual loop.

6.6 Application to FCC single crystal aluminium

As a first application, the Hybrid Model described in the previous sections has been implemented to simulate uniaxial tension of single crystal aluminum. In the present formulation, only the evolution of glissile and stored non-polar dislocations are considered. Under uniaxial tension, the applied deformation does not impose any rotation, and in the calculations that follow, the imposed boundary conditions are such that crystal rotation is uniform, and hence no polar dislocations will be generated. Moreover, because the crystals are being loaded monotonically, the remobilization mechanism is assumed not to affect the mechanical response and is suppressed in the following calculations. Thus, for this application, the rate of growth of non-polar glissile dislocations is due solely to glide, and the rate of storage of non-polar stored dislocations is governed by the rate of annihilation, junction formation and immobilization of non-polar glissile dislocations.

6.6.1 Dislocation interactions

There are three interaction matrices introduced in the present hybrid model. The first is the strength-interaction matrix $a^{\alpha\alpha'}$ found in equation (6.30) in the definition of the critical resolved shear stress for system α. This interaction matrix describes the interaction between dislocations from all possible pairs of slip systems in an FCC

crystal that give rise to the glide resistance known as the critical resolved shear stress. Due to the symmetry of the FCC crystal, this matrix is composed of six independent coefficients, each of them accounting for a different type of interaction [90, 89]. An example matrix $a^{\alpha\alpha'}$ for FCC crystals is shown in table 6.2.

The two coefficients a_0 and a_1 account for in-plane interactions, whereas the four other coefficients a_2 through a_5 account for out-of-plane interactions. The first a_0 describes self-interactions, namely the interactions between dislocations belonging to the same system. Even when gliding on different parallel planes, like-signed, same system dislocations can increase their mutual glide resistance through long-range interactions via dipolar interactions [4]. Next, a_1 describes interactions between coplanar dislocations that have different Burgers vectors, while a_2 accounts for the collinear dislocations; that is, for dislocations that have the same Burgers vector but are lying on different planes. This last interaction is related to the ability of dislocations to cross-slip. Finally, a_3, a_4 and a_5 describe three types of junctions that are likely to be formed, namely the Hirth lock, the glissile junction and the Lomer-Cottrell lock, respectively. The glissile junctions are able to move since their resulting Burgers vector lies in the plane in which they are formed, but still largely contribute to the hardening of the forest [164]. Alternatively, the Hirth and Lomer-Cottrell locks form sessile junctions of different strengths, with the Hirth lock being the weaker of the two.

The components $a^{\alpha\alpha'}$ are taken from the results of DDD simulations conducted in reference [132]. These are reported in table 6.3. Although interaction coefficients evolve with the dislocation densities [65], they will be taken as constant during deformation.

Second, in equation (6.35), the length-interaction matrix $d^{\alpha\alpha'}$ is introduced to describe the increment in the number of intersection events occurring while systems α and α' are interacting. As described by equation (6.35), this number is related to the

Table 6.2: Strength-interaction matrix $a^{\alpha\alpha'}$ coefficients for FCC crystals.

Systems	A2	A3	A6	B2	B4	B5	C1	C3	C5	D1	D4	D6
A2	a_0	a_1	a_1	a_2	a_4	a_4	a_3	a_4	a_5	a_3	a_5	a_4
A3		a_0	a_1	a_4	a_3	a_5	a_4	a_2	a_4	a_5	a_3	a_4
A6			a_0	a_4	a_5	a_3	a_5	a_4	a_3	a_4	a_4	a_2
B2				a_0	a_1	a_1	a_3	a_5	a_4	a_3	a_4	a_5
B4					a_0	a_1	a_5	a_3	a_4	a_4	a_2	a_4
B5						a_0	a_4	a_4	a_2	a_5	a_4	a_3
C1							a_0	a_1	a_1	a_2	a_4	a_4
C3								a_0	a_1	a_4	a_3	a_5
C5			sym						a_0	a_4	a_5	a_3
D1										a_0	a_1	a_1
D4											a_0	a_1
D6												a_0

Table 6.3: Interaction coefficients computed via DDD simulations by Kubin and co-workers [132] and used for the present work.

Self a_0	Coplanar a_1	Collinear a_2	Hirth a_3	Glissile a_4	Lomer a_5
0.122	0.122	0.625	0.07	0.137	0.127

mean free path, which in turn depends on the average lengths of the two interacting dislocations segments. As such, it gives rise to an orientation dependence on the rate of storage of glissile dislocations. The length-interaction matrix $d^{\alpha\alpha'}$ exhibits the same structure as matrix $a^{\alpha\alpha'}$. It also describes the interaction between all possible pairs of slip systems and has six independent coefficients corresponding to the same interactions as those in matrix $a^{\alpha\alpha'}$. Although the values of its six coefficients may differ from those for matrix $a^{\alpha\alpha'}$, for the sake of simplicity, the coefficients of matrices $a^{\alpha\alpha'}$ and $d^{\alpha\alpha'}$ are assumed identical.

Third, the transformation-interaction matrix $H^{\alpha\alpha'}_{end}(\theta, \theta')$ was introduced in equation (6.33), and can also be directly determined from DDD simulations. Such maps, for which an example is given in figure 6.3, indicate the end-state attained, i.e. junction, annihilation, repulsive state or otherwise (*cross-state*), as a function of the orientations of the two interacting glissile segments. Studies performed on collinear interactions in reference [165] have also managed to provide maps for these interactions.

Table 6.4: Averaged intersection probabilities $P_{jun}^{\alpha\alpha'}$ taken for FCC aluminum.

Types of junctions	$P_{jun}^{\alpha\alpha'}$
Lomer-Cottrell lock	$P_{Lomer} = 0.593$
Glissile	$P_{glissile} = 0.547$
Hirth lock	$P_{Hirth} = 0.378$
Collinear interactions	$P_{collinear} = 0.895$

These maps not only inform on the nature of the interaction configuration between two oriented dislocation segments, but also on the length on junctions to be formed according to segments orientations: junctions lengths decrease with an increasing misorientation of the two interacting segments [164]. Here, the transformation-interaction coefficients will be taken as constant and independent of the length of the dislocations during plastic deformation. These are given in table 6.3. It should also be mentioned that very little is known about the interaction between coplanar slip systems, as no studies have been devoted to those interactions. Kubin et al. [132] suggest that the coplanar and self-interactions can be treated in a similar manner. For this reason, as listed in table 6.3, their interaction coefficients are taken as identical.

As mentioned, a simplified approach given in Section 6.4.3 will also be utilized. In this case, the probability of forming a junction or reaching a repulsive configuration $P_{end}^{\alpha\alpha'}$ (see equations (6.50) and (6.49)) can be related to $H_{end}^{\alpha\alpha'}(\theta, \theta')$ introduced in equation (6.33). For instance, $P_{jun}^{\alpha\alpha'}$ for junction formation can then be expressed as:

$$P_{jun}^{\alpha\alpha'} = \frac{\mathcal{A}_{jun}^{\alpha\alpha'}}{\mathcal{A}_{tot}^{\alpha\alpha'}} \tag{6.52}$$

where $\mathcal{A}_{jun}^{\alpha\alpha'}$ denotes all the mapped areas corresponding to interactions between systems α and α' that lead to junction formation, and $\mathcal{A}_{tot}^{\alpha\alpha'}$ is the total map area. For the FCC aluminum crystals modelled here, values of $P_{jun}^{\alpha\alpha'}$ obtained from junction formation maps of Madec and co-workers [164, 165] are reported in table 6.4.

6.6.2 Simulation setting

In the numerical model, the single crystal is represented by a single 3D 8-noded brick element. In order to simulate uniaxial tension of single crystals, the following boundary conditions have been adopted: the element, which is subjected to uniaxial traction on its upper face (along the Z-axis), is set up to represent 1/8th of a cube by taking advantage of the symmetries (FCC materials). Therefore, to reproduce the ideal tension boundary conditions, the lower face of the element is pinned along the traction direction and one of its vertices is pinned, whereas the others are blocked to avoid rotation of the lower face. The lateral sides of the entire cube are traction free. Tension is simulated by imposing a uniaxial strain rate $\dot{\epsilon}_{33}$ on the upper face. All boundary conditions on the 1/8th cube are schematically presented in figure 6.5.

In order to test the response of uniaxial tension to different crystal orientations, a crystal coordinate system has been implemented. Its rotation with respect to the global coordinate system (element) is performed via the use of Euler angles (using Bunge convention). The models were tested for four orientations: [001], [111], [112] and [123], to evaluate their ability to capture and predict the orientation-dependent tensile behavior of single-crystal aluminum. Results are compared to experiments conducted by Hosford et al. [115] at 273 K. Materials parameters used to simulate single crystal aluminum at 273 K are reported in table 6.5.

The initial density of non-polar glissile dislocation is set to $3.5 \times 10^{10} m^{-2}$ on each of the 24 virtual slip systems, which is equivalent to $7 \times 10^{10} m^{-2}$ on each of the 12 actual slip systems. The initial density of non-polar stored dislocations is set to the same value. Uniaxial tensile tests are simulated under a constant strain rate $\dot{\epsilon}_{33} = 3 \times 10^{1} s^{-1}$. The time step used in the simulation is set to $\Delta t = \times 10^{-8} s$ for both the FE and the DDD procedure. The drag coefficient $\boldsymbol{B}$ in equation (6.21) is chosen to be identical for edge and screw orientations, such that its value remains constant for all possible lines orientation. The Helmholtz free activation energy F0 found in

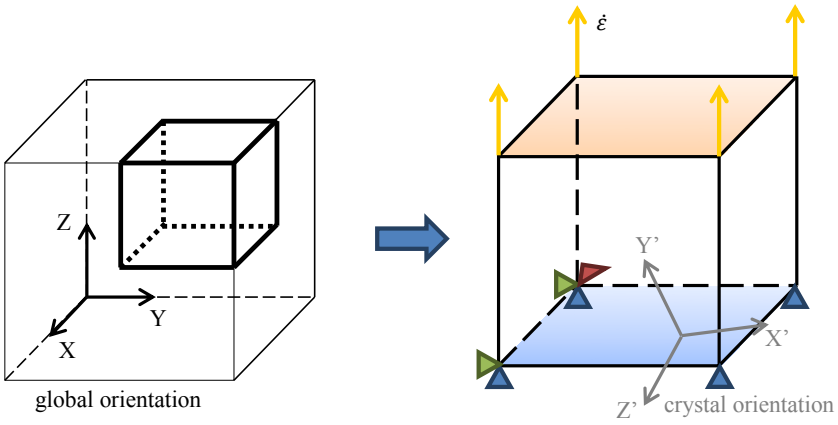


Figure 6.5: Schematic of the 1/8th cube element and its boundary conditions to simulate uniaxial tension on a single crystal Al.

Table 6.5: Materials parameters used to model single crystal aluminum.

Property	Value
Elastic constants	$C_{11} = 108$ GPa
	$C_{12} = 61.3$ GPa
	$C_{44} = 28.5$ GPa
Shear modulus	$\mu = 25$ GPa
Burgers vector	$b = 2.863 \times 10^{-10}$ m
Velocity exponents	$p = 0.5$ and $q = 1.5$
Drag coefficient	$B = 10^{-5}$ Pa.s
Lattice friction at 0K	$\tau_0 = 30$ MPa
Free activation energy	$F_0 = 5 \times 10^{-20}$ J
Trapping coefficient	$k_1 = 6.7 \times 10^{-3}$
Recovery coefficient	$k_2 = 1.7 \times 10^{-8}$

the Boltzmann distribution (equation (6.27)) is typically given in $0.05 \leq \frac{F_0}{\mu b^3} \leq 2$ whereas the exponents p and q are to be comprised in the ranges $0 \leq p \leq 1$ and $1 \leq q \leq 2$.

Finally, it is important to notice that the present model has only two adjustable material parameters, k_1 and k_2, that are independent of the slip systems. The first parameter represents the coefficient for the rate of trapping of glissile dislocations by other dislocations and the second the recovery processes between stored dislocations. These are determined by fitting the tension response in the perfectly aligned [111]

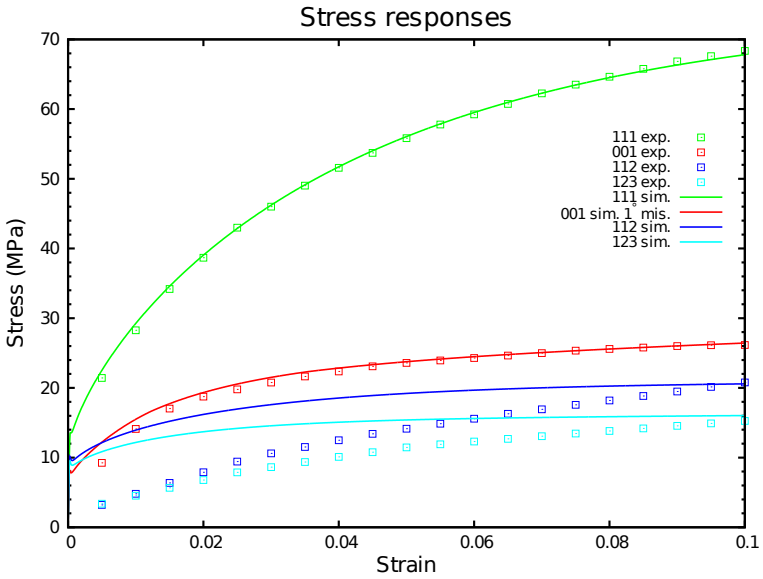


Figure 6.6: Stress vs strain predictions for the four tested crystal orientations using the simplified version of the model versus experimental data.

direction. Simulations of the [001], [112] and [123]-oriented crystals have then been performed using the same values for these two parameters. In doing so, we found that the best agreement of the model with experimental data have been obtained introducing a 1° misorientation for the [001]-oriented crystal. This slight adjustment is well within the expected error of the experiment. The orientations of specimens used in the experiments conducted by Hosford et al. [115] are likely to deviate from the perfect alignment up to 2°.

6.7 Results and discussion

Figure 6.6 compares the hybrid model with experiments conducted by Hosford et al. [115], for four crystal orientations: [001], [111], [112] and [123]. The model captures the strain hardening response and more importantly, the differences and order in the flow stresses among the four orientations. Notably, the response of the [001] orientation is fully reproduced.

Table 6.6: Schmid factors for the four different tested orientations.

System	SB	[001]	[111]	[112]	[123]
11	A2	0.408	0	0.136	0.117
12	A3	0.408	0.272	0.408	0.467
10	A6	0	0.272	0.272	0.350
3	B2	0.408	0	0.272	0.175
2	B4	0.408	0	0.272	0.350
1	B5	0	0	0	0.175
9	C1	0.408	-0.272	0	0
8	C3	0.408	-0.272	0	0
7	C5	0	0	0	0
5	B2	0.408	0.272	0.408	0.292
6	B4	0.408	0	0.136	0.117
4	B5	0	0.272	0.272	0.175

6.7.1 Slip activity and work hardening

The model predicts that the two more symmetrical orientations [001] and [111] deform plastically in multi-slip, whereas the other two orientations [112] and [123] deform predominantly by one or two slip systems (although the others are mildly active). This difference classifies these four orientations into two groups, which are discussed in turn below. The Schmid factors computed for the tested orientations are reported in table 6.6.

The [001] and [111] orientations deform in tension by activating eight and six slip systems, respectively. For the [001] oriented crystal, systems A2, A3, B2, B4, C1, C3, D1 and D4 are activated, whereas for the [111]-oriented one, systems A3, A6, C1, C3, D1 and D6 are activated. These systems have identical Schmid factors and are activated simultaneously at the onset of plastic deformation, marking the end of the elastic stage. Because of multi-slip, stage I is not present and strain hardening begins with stage II at a very high hardening rate. All the remaining non-active slip systems have a zero Schmid factor. The [111]-oriented crystal exhibits the highest hardening rate among the four orientations, and a flow stress at 10% strain that is

roughly four times higher than in any other orientation. However, unlike the [111]-oriented crystal, the hardening rate of the orientation [001] (with eight equally active slip systems) drops drastically after 2–3% strain, such that the stress level has almost reached an asymptotic value at 10% strain.

It has been observed experimentally that at most half of the slip systems with non-zero Schmid factors are actually activated during tensile deformation for [001] and [111] orientations [113]. For the [111] orientation, Franciosi and Zaoui [90] observed that only three distinct slip systems are activated. The selection of activated slip systems has been argued based on the types of junctions that could form. In [90], at the beginning of plastic deformation, the preferred combinations of systems involved were those that do not form junctions. Using both DDD and continuum models, Devincre et al. [64] found that the selection of slip systems proceeds in such a way that collinear interactions between a slip system and its cross-slip system are avoided. Collinear junctions form when collinear slip systems, which have the same Burgers vector but lie on different planes, interact. These junctions are by far the strongest, as seen through the a_2 coefficient in table 6.3, which is more than five times higher than the second highest glissile junction. In the present hybrid model, the junction types formed and slip activity are predicted and their relationship is discussed in the next section.

For orientations [112] and [123], one to two slip systems dominate and accommodate most of the strain, although all systems with non-zero Schmid factors are actually activated during the course of the deformation. This provides an explanation for why stage I does not manifest in either orientation. In the case of the [112]-oriented crystal, it has been observed experimentally for FCC materials that a small single glide stage I could occur, with secondary slip on the conjugate slip system (the maximum resolved shear stress is identical for two slip systems in this direction). This has been reported for Copper in [90] and appears clearly in the experiments conducted in

[226]. However, in the case of aluminum, stage I is not present as double conjugate slip occurs at the onset of the plastic deformation [90]. Similarly in the case of tension along the [123] direction for aluminum at 273 K, experiments in [115] show that single glide stage I is nonexistent such that stage II immediately occurs after the onset of plasticity, whereas at 4.2 K and 77 K, stage I is clearly present until 2–3% strain.

The main discrepancy of the model is found in the yield stress of the [112] and [123] orientations, measured to be around 2–3 MPa, but predicted to be 10 MPa. The model presented here follows a rate-dependent formulation and hence the elasto-plastic transition is sensitive to strain rate. Therefore, imposing lower strain rates is expected to lead to a decrease in the predicted yield stress.

6.7.2 Junction formation and work hardening

In the present formulation, junctions form as a result of the interactions between glissile dislocations. Among them, we account for both active-active (interactions between two active systems) and active–inactive (interactions between an active and a non-active system) interaction pairs. Active-inactive paired interactions contribute to the junction densities to a lesser extent than the active-active pairs, as the rates of transformation from glissile dislocations to stored dislocations is directly limited by the smaller of the two interacting populations (see equation (6.50)). However, these interactions are responsible for reducing the density of the latent systems, despite their inactivity, and more importantly contribute to dislocation storage. This directly characterizes the latent-storage process. Through these various interactions, different types of junctions form, and the present hybrid model determines which junctions form based on the orientation of the two slip systems and frequency of their interaction. Both are tied to the number and distribution of active slip systems in the crystal.

The [111]-oriented crystal has the highest hardening rate due to the large fraction

Table 6.7: Active slip systems and resulting interaction types predicted by the model. In (a), active slip systems are ranked in terms of predominance. For the [001] and [111]-oriented crystals, the activation of each slip system is identical due to the symmetrical configuration.

Traction direction	[001]	[111]	[112]	[123]
(a) Active slip	8	6	8	9
systems	A2, A3, B2, B4, C1, C3, D1, D4	A3, A6, C1, C3, D1, D6	A3, D1, A6, B2, B4, D6, A2, D4	A3, A6, B4, D1, B5, D6, B2, D4, A2
(b) Number of distinct active-active pairs interactions	28	15	28	36
Collinear	4 (14%): A2-B2, A3-C3, B4-D4, C1-D1	3 (20%): A3-C3, A6-D6, C1-D1	3 (11%): A2-B2, A6-D6, B4-D4	3 (8%): A2-B2, A6-D6, B4-D4
Coplanar	4 (14%): A2-A3, B2-B4, C1-C3, D1-D4	3 (20%): A3-A6, C1-C3, D1-D6	7 (25%): A2-A3, A2-A6, A3-A6, B2-B4, D1-D4, D1-D6, D4-D6	9 (25%): A2-A3, A2-A6, A3-A6, B2-B4, B2-B5, B4-B5, D1-D4, D1-D6, D4-D6
Lomer-Cottrell	4 (14%): A2-D4, A3-D1, B2-C3, B4-C1	3 (20%): A3-D1, A6-C1, C3-D6	4 (14%): A2-D4, A3-D1, A6-B4, B2-D6	6 (17%): A2-D4, A3-B5, A3-D1, A6-B4, B2-D6, B5-D1
Glissile	9 (32%): A2-B4, A2-C3, A3-B2, A3-C1, A3-D4, B2-D4, B4-D1, C1-D4, C3-D1	6 (40%): A3-C1, A3-D6, A6-C3, A6-D1, C1-D6, C3-D1	11 (39%): A2-B4, A2-D6, A3-B2, A3-D4, A3-D6, A6-B2, A6-D1, A6-D4, B2-D4, B4-D1, B4-D6	12 (33%): A2-B4, A2-B5, A2-D6, A3-B2, A3-D6, A6-B2, A6-D1, A6-D4, B2-D4, B4-D1, B4-D6, B5-D4
Hirth locks	7 (25%): A2-C1, A2-D1, A3-B4, B2-C1, B2-D1, B4-C3, C3-D4	0 (0%)	3 (11%): A2-D1, A3-B4, B2-D1	6 (17%): A2-D1, A3-B4, A3-D4, A6-B5, B2-D1, B5-D6

of strong junctions, particularly the strongest collinear junctions, formed from its slip activity. In the [111]-oriented crystal, the six activated systems lead to 15 distinct systems interactions, as reported in table 6.7. Among them, the collinear, coplanar and Lomer locks interactions account for the total interactions in the same proportion (20% each), whereas the glissile junctions occur twice as often, representing 40% of the overall interactions. However, the weakest junction, the Hirth lock, cannot form for the given set of activated systems. As a result, interactions leading to the harder rather than weaker junction configurations occur more often, which in turn gives rise to the highest hardening rate in stage II among the four orientations tested.

The [001]-oriented crystal activates the largest number of slip systems, namely eight, leading to the largest number of distinct interaction types predicted among all

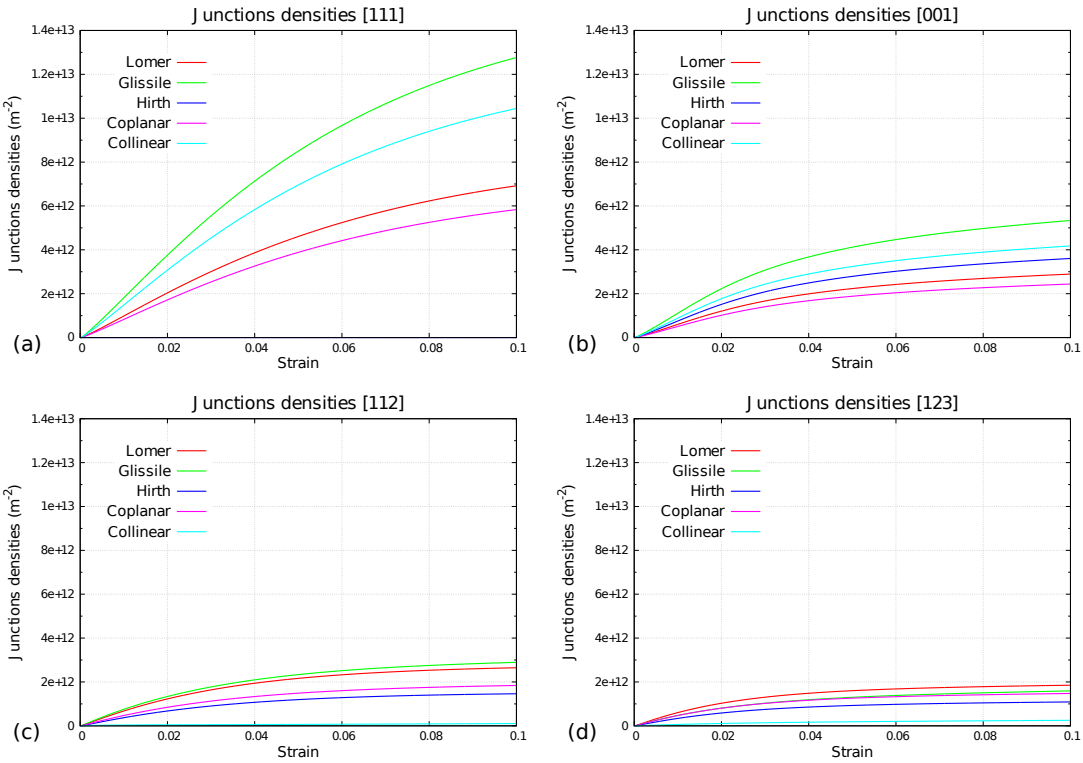


Figure 6.7: Predicted densities of junctions populations for the four orientations.

four orientations. As in the [111]-oriented crystal, glissile junctions are the dominant junction type (32% of the total interactions) and the collinear, coplanar and Lomer interactions form in equal proportions. However, unlike in the [111]-oriented crystal, this orientation favors the formation of Hirth locks, as shown in figure 6.7. This large proportion of weak junctions is partly responsible for the lower hardening of the [001]-oriented crystal during stage II than the [111]-oriented crystal.

The model also captures the lower hardening rate for the [123] orientation than the [112] orientation in agreement with experiment. Although eight and nine slip systems are activated during the course of the deformation for the [112] and [123]-oriented crystals, respectively, the slip activity is inhomogeneous. Only two systems A3 and D1 are found to be predominant for the [112] orientation, and only one system A3 in the [123] orientation. The model predicts that compared to that of the [123] orientation, the slip activity of the [112] orientation produces more collinear junctions and glissile junctions, which are commonly considered as being some of the strongest

ones. As a result, stage II hardening for the [112] orientation is calculated higher than the one for the [123] direction. The extent of straining involved in stage II, however, is the same; in both the [112] and [123] orientations, stage II transitions to stage III at around 4–5% strain.

6.7.3 Glissile dislocation density evolution

One advantage of the hybrid single crystal constitutive law is that it contains evolution laws for both glissile and stored dislocation densities. For both [111] and [001] orientations, it is found that the evolution of glissile densities on all active slip systems are identical (see figure 6.8(a) and (b)). This corresponds to the fact that the velocities of the dislocations are uniform among the active system throughout deformation. For the [112] and [123] orientations, slip is inhomogeneous, with one to two systems being the most active among several systems of lesser activity. Consequently, glissile dislocation velocities are distributed, and do not evolve at the same rate, as observed on figure 6.8(c) and (d).

The total dislocation density and the ratio of glissile-to-stored densities predicted by the model are given in figure 6.9. The total densities predicted by the model ranges from $1 \times 10^{14} m^{-2}$ for the [123]-oriented crystal to $3.2 \times 10^{14} m^{-2}$ for the [111]-oriented crystal at 10% strain. X-ray measurements conducted on copper and aluminum single crystals by Hordon et al. [114] have shown dislocation densities to reach magnitude between 10^{14} and $10^{15} m^{-2}$ for a plastic shear strain of 30%. More recently, Groma et al. [103] have measured dislocations densities of $10^{14} m^{-2}$ using X-ray line profile analysis on compressed Cu single crystal. Other techniques such as TEM imaging have also been used by Staker and Holt [223] where the dislocation density in a Cu sample deformed at 10% strain in tension was measured to be $1.18 \times 10^{14} m^{-2}$. Similar results were obtained by Heuser [109], using neutron scattering techniques, who measured $1.9 \times 10^{14} m^{-2}$ at 16% compression of Cu single crystal. Thus, the total

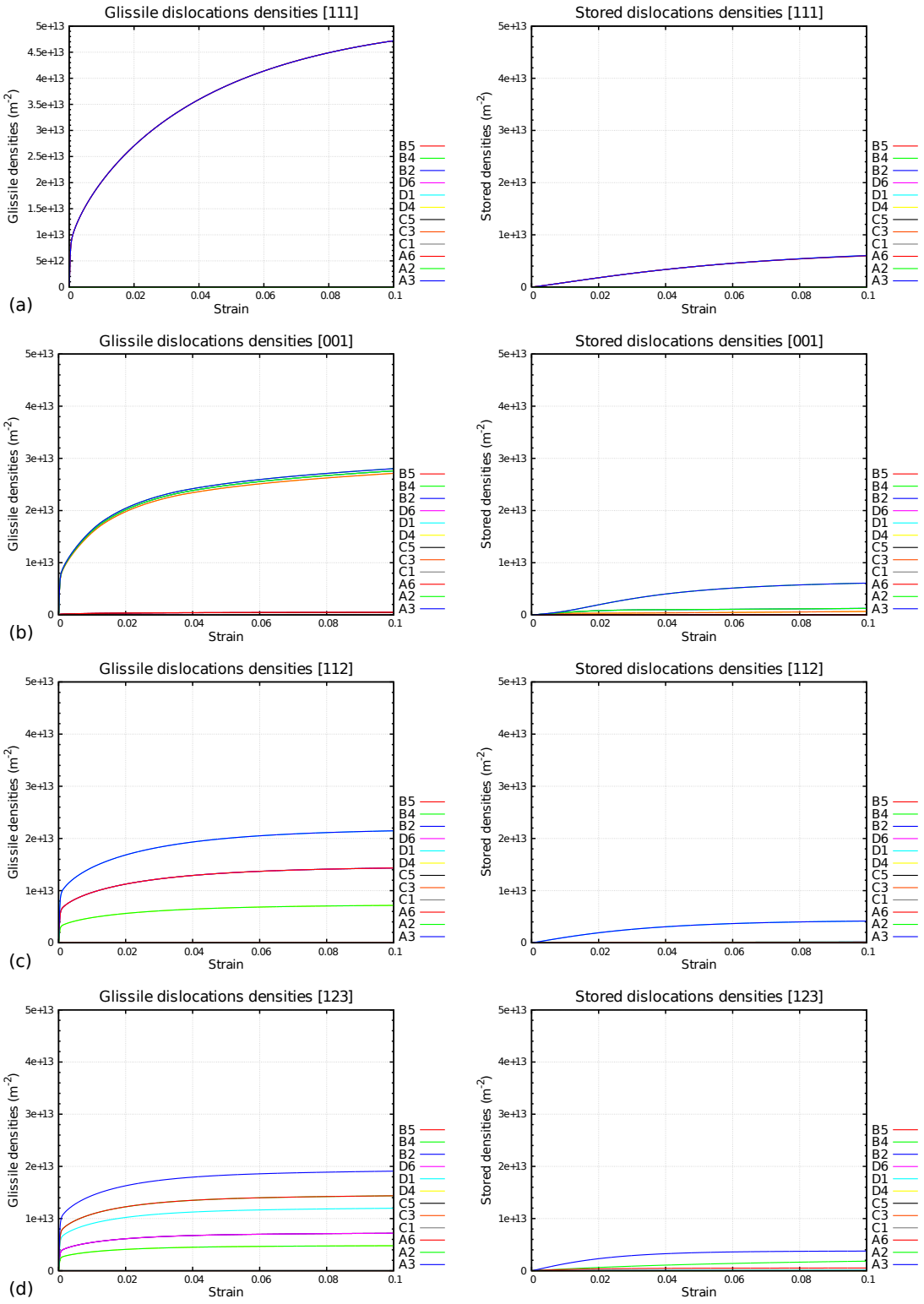


Figure 6.8: Predicted densities of glissile and stored non-polar dislocations for the four orientations.

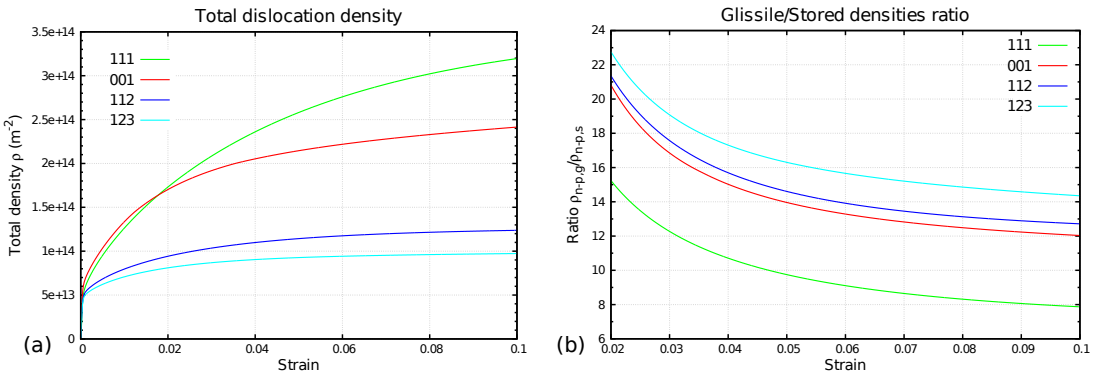


Figure 6.9: (a) Total densities of non-polar dislocations and (b) ratio of the glissile to the stored dislocation densities.

dislocation densities reported in figure 6.9(a) are in good agreement with the ones obtained experimentally.

Calculations for these monotonic loading conditions indicate that during plastic deformation, most dislocations are glissile. In other words, the stored densities comprised of junctions and repulsive states make up a small fraction of the total dislocation density. As shown in figure 6.8 and figure 6.9(b), the model predicts glissile dislocation densities that are roughly one order of magnitude greater than those predicted for stored dislocations. This result largely agrees with many DDD simulations of single crystal deformation [248, 249, 250]. In the present model, the calculated velocities of the glissile dislocations are distributed and so the glissile distribution includes a velocity range from the immobile, inactive systems to the slowly moving dislocations that contribute little strain to the very fast glissile dislocations that accommodate most of the strain [248]. The stored densities correspond only to the small portions of dislocations that have formed junctions or that are stalled by a repulsive interaction.

Apart from a few exceptions [276, 206, 11, 249, 250, 38] wherein evolution laws for glissile and stored dislocations were also introduced, in most single crystal constitutive laws, the glissile density is assumed constant (having reached a saturation value) and/or some fraction of the total density. Under this assumption, only the evolution

of stored dislocations is treated [127, 25, 29, 122]. For instance, in the work of Devincre and Kubin [70] and Kubin et al. [132, 133], it is assumed that glissile dislocations account for no more than 5% to 10% of the total densities, such that the density of stored dislocations can be approximated by the total density. Alternatively, in the work of Arsenlis and Parks [7], all dislocations are assumed glissile. The glissile densities and their evolution govern both the plastic shearing rate via Orowan's law and the glide resistance.

6.7.4 Discussion

In order to quantify the effects of latent-storage and latent-hardening independently, simulations were performed on a [111]-oriented crystal in two specific cases: (a) only self-storage is accounted for, i.e. transformations from glissile to stored dislocations is solely due to interactions of dislocations belonging to the same slip system families, and (b) only self-hardening is accounted for, i.e. all other coefficients of the strength-interaction and length-interaction matrices are set to zero.

Figure 6.10(A) shows the stress responses of the [111]-oriented crystal in cases (a) and (b) where the parameters k_1 and k_2 are equal to that of the initial fitted response. As expected, for both simulations (a) and (b), the hardening rate is higher when full latent-storage and latent-hardening are taken into account. In case (a) the transformations from glissile to stored dislocations result exclusively from coplanar interactions. As all densities are contributing to the hardening of the material, case (a) has the highest flow stress as well as the lowest stored dislocation density, as shown in figure 6.10(B). In case (b) in which the strength and length-interaction coefficients of table 6.3 are set to zero (only self-hardening is present), the hardening rate is reduced substantially (by approximately half). Interestingly in this case (b), the density of stored dislocations is about two times greater than for the initial configuration (full hardening). Also, the total glissile dislocation density is the lowest among the three

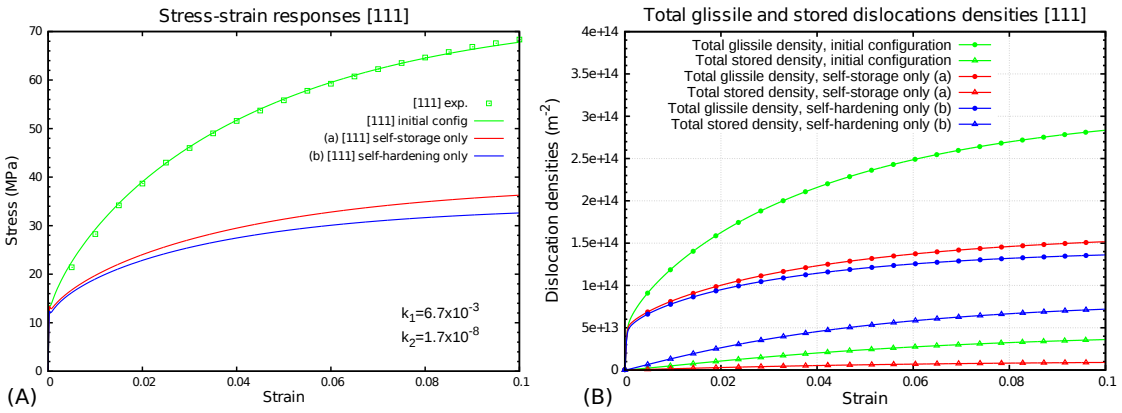


Figure 6.10: (A) [111] stress responses and (B) total dislocation densities where (a) self-storage and (b) self-hardening are only taken into account for the same material parameters k_1 and k_2.

Table 6.8: Values of the trapping coefficient k_1 and the recovery coefficient k_2 to achieve the same stress response.

	Initial configuration	(a) Self-storage only	(b) Self-hardening only
Trapping coefficient k_1	$k_1 = 6.7 \times 10^{-3}$	$k_1 = 2.1 \times 10^{-2}$	$k_1 = 2.4 \times 10^{-2}$
Recovery coefficient k_2	$k_2 = 1.7 \times 10^{-8}$	$k_2 = 1.7 \times 10^{-8}$	$k_2 = 1.2 \times 10^{-8}$

configurations, and because only self-interactions contribute to strain hardening, the flow stress is the lowest as well.

For these two simulations (a) and (b), the two constitutive law parameters k_1 and k_2 have also been fitted such that the mechanical response of the [111]-oriented crystal concur with experimental results. Values of parameters k_1 and k_2 are reported in table 6.8 for simulations (a) and (b), and for the initial configuration presented in figure 6.6, where latent-storage and latent-hardening are fully accounted for.

In case (a) the results show that in order to achieve the same level of stress, the amount of transformations needs to be identical to that of the initial case. Hence, as shown in figure 6.11(a), the total densities of glissile and stored dislocations are nearly identical to the case where latent-storage is fully accounted for. However, in case (a), the stored dislocations are uniquely comprised of coplanar junctions, whereas they are comprised of glissile, collinear, Lomer and coplanar junctions populations in the case where transformation coefficients are chosen according to table 6.4, as shown in

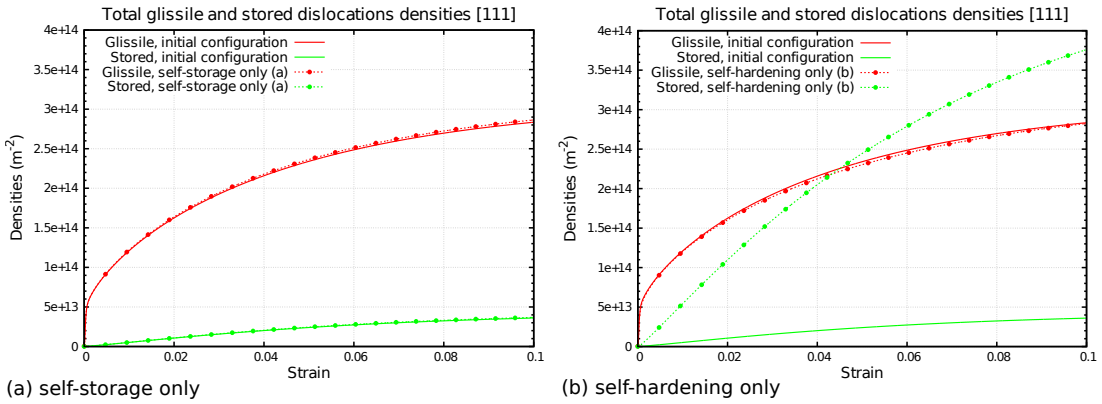


Figure 6.11: Total glissile and stored dislocations densities in all slip systems for (a) self-storage only and (b) self-hardening only compared to that of the initial configuration for the [111]-oriented crystal.

figure 6.7(a).

For simulation (b), the results show that the attainment of the same level of stress, when the material is solely hardened by the self-interactions, requires a significantly larger amount of transformations from glissile to stored dislocations. As a result, as shown in figure 6.11(b), the model now predicts a total amount of stored dislocations much larger than when full hardening is accounted for. In case (b), it is interesting to notice that the total amount of glissile dislocations on all systems is quasi equivalent to that when full latent-hardening is simulated. This type of approach is somehow similar to models where the plastic shearing rate is expressed as an exponential thermally activated factor (e.g. equation (6.27)) pre-multiplied by a reference shearing rate $\dot{\gamma}_0^s$ [33, 45].

The model presented here is therefore able to replicate features of other simpler validated models by simple choice of parameters, and more generally, poses the question of the quantification of densities from an experimental standpoint.

Figure 6.12 shows the evolution of the mean dislocation velocities for the [111]-oriented crystal. At the onset of dislocation glide, the dislocation velocities computed from the DDD procedure, denoted DDD velocities, range from 20–100 m/s. These

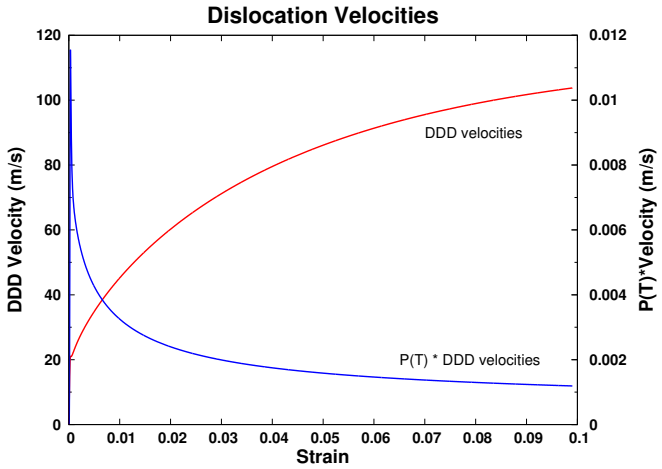


Figure 6.12: Comparison between averaged DDD dislocation velocities $\vec{\boldsymbol{v}}^{\alpha}(\theta)$ and thermally activated velocities $\vec{\boldsymbol{V}}^{\alpha}(\theta)$ entering the model (see equation (6.26)).

values are consistent with those measured experimentally [192, 231] for comparable level of stresses. The DDD velocities represent the speed at which dislocations are gliding when they are not impeded or pinned by other dislocations in the crystal. In thermally activated dislocation glide, only a fraction of the dislocations are moving at full speed. To determine this fraction, DDD velocities are multiplied by a temperature-dependent probability $P^{\alpha}(T)$ that represents the probability that thermal fluctuations are sufficient to unpin dislocations (see equation (6.26)). The product is referred to as the *model velocity*. Fig. 11 shows that the model velocity decreases and varies between 0.01 and 0.002 m/s. This indicates that most of the glissile segments are impeded by local obstacles. As strain increases, the dislocation density increases, causing more dislocations to become pinned and the model velocity to decrease. Fewer dislocations are actually moving. Because fewer are required to accommodate most of the deformation, their velocity is much higher than the average value shown in figure 6.12. In other words, the plastic deformation is being carried by fewer and fewer dislocations as straining increases, a result consistent with DDD simulations in [248].

In this model, dislocation density based evolution laws and DDD simulation have

been coupled to address two separate underlying mechanisms of strain-hardening, namely latent-storage and latent-hardening. First, latent-storage described by the transformation-interaction matrix $H_{end}^{\alpha\alpha'}(\theta,\theta')$ introduced in equation (6.33) (or $P_{end}^{\alpha\alpha'}$ in its simplified form in equation (6.49)) pertains to the transformation rate of glissile into stored dislocations according to the nature of the interactions between intersecting dislocations, and is directly informed by DDD simulation data. To the author's knowledge, except for the model suggested by Roters et al. [206], transformations such as those described in equation (6.40) have not been explicitly accounted for before in those terms. Second, latent-hardening, which describes the hardening of slip systems that are not active during plastic flow, is treated in a classical fashion via the use of an individual CRSS for each slip systems (see equation (6.30)). This formulation was first presented by Teodosiu et al. [228] as an extension of Taylor law to account for the effects of different forest densities. It introduces a multi-system strength-interactions matrix $a^{\alpha\alpha'}$ [90], which in this work is informed by DDD results.

Note that in order to obtain results for the four different orientations tested here, the two parameters of the model – namely the trapping coefficient k_1 and the recovery coefficient k_2 – were fitted for the perfectly aligned [111]-oriented crystal. Simulations of the [001], [112] and [123]-oriented crystals were then performed using the same values for these two parameters. Alternatively, in the work of Kubin et al. [132] and Devincre and Kubin [70], where the storage rate is expressed as:

$$\frac{d\rho}{d\gamma} = \frac{1}{bL_f} \quad \text{with} \quad L_f = \frac{K_{hkl}}{\sqrt{\sum_u a^{su}\rho^u}} \tag{6.53}$$

where L_f is the mean free path. This implies an orientation dependence of strain hardening, such that the dimensionless mean free path coefficient K_{hkl}, which is dependent of the number of active slip systems, needs to be computed using DDD simulations

for each deformation path, although in practice many authors use an orientation-independent coefficient in their models. In the model proposed here, however, determining the orientation-independent trapping parameter k_1 does not require massive DDD simulations involving numerous dislocations, but a simple fitting procedure.

In summary, the Hybrid Model proposes a new approach for integrating DDD information into larger length scale dislocation density based constitutive models. This allows for (1) the explicit integration of junctions formation and (2) a significant reduction in computational time thanks to the use of a small set of virtual dislocation loops. In order to simulate a representative volume describing a single crystal, classic DDD simulations usually treat up to several thousands of dislocation segments. In this model, the evolution of only 24 virtual loops has to be calculated, with each loop being representative of one of the polarized slip systems presented in table 6.1.

Finally, the overarching goal of the Hybrid Model is to predict single crystal and polycrystal response under complex loading conditions. Achieving this necessarily requires inclusion of polar dislocation densities. In metallic polycrystals, generation of polar dislocations is required in order to accommodate strain gradients arising from strain and curvature incompatibilities. The framework proposed here can be extended to account for such effects. Following the same reasoning as that taken in the above, it is likely that as a result of transformation of mobile non-polar segments, polar segments could be generated at a rate that can be estimated by direct computation of the incompatibility arising from the transformed lines due to junction formation and to forward immobilizations. One could therefore suggest a complementary source mechanism for the generation of polar dislocations coupled with the kinematically based rate evolution laws, by virtue of which the polar dislocation density tensor evolves at a rate equal to that of the curl of the plastic strain tensor.

6.8 Conclusion

In this section, a dislocation density based constitutive model referred to as the Hybrid Model is introduced. The model proposes a decomposition of dislocation densities into glissile and stored as well as polar and non-polar densities. The present formulation introduces highly coupled and separate rate laws for glissile (with unconstrained motion) and stored (sessile or constrained motion) dislocations, the latter including different junction types. The rate laws of the (non-polar) glissile dislocations make use of a virtual dislocation loop representation for dislocations of all orientations belonging to the same system. The interconnected motion of these dislocations is numerically calculated via the DDD method. Rate laws for stored dislocations are also developed such that several model parameters can be directly extracted from DDD simulations, such as outcomes of paired dislocation interactions and dislocation junction yield surfaces. As a first application, the model demonstrated its capability to capture the crystal orientation dependence in the uniaxial tensile response of single crystal aluminum with one small set of material parameters. The model reveals quantitatively the relationship between dislocation storage rates, junction formation, strain hardening, and slip activity. Prediction of the material behavior under more complex strain paths entails further refining the Hybrid Model such as to incorporate junction remobilization mechanisms and as to account for polar dislocation densities. In doing so the model shall be applicable to well-known strain path change tests, such as the Bauschinger test and cross tests.

CHAPTER VII

CONCLUSIONS

The main goal of the present research was to contribute to scale transitioning by introducing novel numerical tools and techniques to investigate and obtain a better understanding of dislocation interactions with lattice defects in metals, such as to address the questions of dislocation-dislocation, dislocation-twin, and dislocation-precipitates interactions. For such purpose, a careful and rigorous methodology was used to (1) develop a novel powerful discrete dislocation dynamics approach that addresses the limitations of current simulations and extends their range of applicability, (2) use it to extract critical parameters and laws that will be used into higher-scale models, and (3) develop a new constitutive framework that delineates novel routes allowing for the connection between discrete dislocation dynamics simulations and dislocation density based models.

The first challenge of this work was to build a robust, efficient and trustworthy discrete dislocation dynamics tool capable of treating any kind of crystalline structure and incorporating the treatment of junctions' formation in an explicit and physically relevant manner. This first step was particularly important as this tool was further intended to serve as a basic framework for all other models developed and all studies undertaken in this thesis. This was achieved by implementing state-of-the-art numerical techniques and by devising an appropriate parallel computing strategy.

The DDD tool was first employed to quantify dislocation-dislocation interaction coefficients for HCP magnesium for the first time. In doing so, the asymmetry in the hardening matrix and the strong stability of basal / pyramidal interactions were notably revealed. As such parameters have never been calculated thus far, generic

coefficients obtained through a fitting procedure were used in constitutive models. Therefore, the impact of their precise estimation on the macroscopic response of polycrystalline aggregate was assessed, and it was found that their values have a significant impact on the predictions of dislocation densities and plastic anisotropy response. The DDD tool was further used to demonstrate the validity of the FZ slip-hardening law – for which an extension for HCP crystals was proposed –, while it was shown that the FZ law consistently overestimates the multi-hardening predictions.

In order to study the interaction between dislocations and twin boundaries, a transmission scheme was then implemented in the DDD framework to model dislocation reactions upon crossing of a coherent interface. By studying the collective effect of transmission events, a rapid multiplication of dislocations due to dissociations of perfect dislocations at the TB was revealed. Further, it was observed that, in addition to providing a mechanism for twin growth, partial interfacial dislocations generated in this process act as stress riser agents in the twin while their profusion rapidly transforms the interface into an incoherent boundary.

The next challenge was to devise a novel approach that addresses the limitations of current DDD simulations, especially regarding the treatment of anisotropic and heterogeneous elasticity, so as to enable the study of plasticity in heterogeneous structures. To address both these limitations at the same time, a novel computationally efficient DDD approach based on an eigenstrain formalism and on the use of fast Fourier transforms was devised and implemented: the DDD-FFT. As a first approach, the method was used to examine the interaction between dislocations and second-phase particles. From a more general perspective, this approach offers new perspectives to DDD simulations. Particularly, it is expected that it could be successfully employed to investigate the effect of the shape and orientation of precipitates on dislocation propagation with a level of details never achieved before, so as to extract hardening trends that will be incorporated into constitutive laws, thereby guiding

microstructural design for fabricating alloys with higher strength. Furthermore, the heterogeneous DDD-FFT method developed in this thesis can virtually be employed to treat an infinite number of phases simultaneously, thereby paving the way towards performing full-scale DDD simulations in polycrystalline materials that include the effects of anisotropic elasticity.

Using the knowledge gained from fine-scale applications, a new strain hardening constitutive framework informed by DDD was then developed. In order to account for the evolution of glissile dislocations that is generally overlooked in current approaches, the core idea of the model relied in introducing a direct coupling with a DDD approach so as to precisely quantify the motion due to glide and dislocation interactions leading to glissile to stored transformations. The model was first applied to simulate tension in aluminium and demonstrated its capability to capture the orientation dependence response while highlighting the prominent role of glissile dislocations. Furthermore, the model could be ideally extended to incorporate a stochastic approach so as to account for the field distributions observed at lower scales, for which the full-field DDD-FFT method would provide a powerful tool for their quantification.

Finally, the work done in this thesis paves the way towards bridging the gap between fine and macro scales by providing new tools that allow for an exchange of information between the scales. In that sense, the present work opens up new potential areas of research. Particularly, the expertise earned and the novel approaches developed in this thesis are expected to guide the development of new predictive constitutive laws that are in direct connection with the physics of the dislocations.

APPENDIX A

STRESS FIELD OF A DISLOCATION IN AN ISOTROPIC MEDIUM

In this appendix, the formulations used for evaluating the stress field produced by dislocations in an elastically isotropic infinite medium are presented in detail. First, the general singular formulation introduced for a closed dislocation loop is recalled and details on its derivation are provided. Then, the non-singular formulation proposed by Cai et al. [35] is presented and the derivation of a fully analytical coordinate-independent form is detailed.

A.1 General formulation

As derived in Section 2.2.2, the stress field induced by a static dislocation loop in an infinite medium is given by the general line integral (2.14) first obtained by Mura [180, 181]:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \oint_L e_{lnh} C_{pqmn} G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_m dx'_h \tag{A.1}$$

where $\vec{\boldsymbol{x}}$ is the field point at which the stress is evaluated, C_{ijkl} is the elasticity tensor of the medium, $\vec{\boldsymbol{b}}$ is the Burgers vector of the dislocation, and where the integral of the first derivative of the Green's function $G_{kp,q}$ is carried out along the closed dislocation line L spanned with coordinate $\vec{\boldsymbol{x'}}$. For an elastically isotropic infinite medium, an analytical expression of the static Green's function is given by (e.g. see refs [99, 72, 181]):

$$G_{kp}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) = \frac{1}{8\pi\mu} \left[\delta_{kp} R_{,qq} - \frac{1}{2(1-\nu)} R_{,kp} \right] \tag{A.2}$$

where μ and ν are the shear modulus and the Poisson's ratio of the medium, respectively, δ_{ij} is the Kronecker symbol, and $R = \|\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}\|$ is the norm of the radius vector linking the field point $\vec{\boldsymbol{x}}$ and the source point $\vec{\boldsymbol{x'}}$. If one denotes the radius vector as $\vec{\boldsymbol{R}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}$ with coordinates $R_i = x_i - x'_i$, its norm R is explicitly given as:

$$R = \sqrt{\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{R}}} = \sqrt{R_i R_i} = \sqrt{(x_i - x'_i)(x_i - x'_i)} \tag{A.3}$$

In equation (A.2), $R_{,ij} = \partial_i \partial_j R = \frac{\partial^2}{\partial_i \partial_j} R$. Taking the first derivative of the Green's function yields:

$$G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) = \frac{1}{8\pi\mu} \left[\delta_{kp} R_{,mmq} - \frac{1}{2(1-\nu)} R_{,kpq} \right] \tag{A.4}$$

In isotropic elasticity, the stiffness tensor C_{ijkl} is conveniently written as:

$$C_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu \left(\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk} \right) \tag{A.5}$$

Combining equations (A.1), (A.4) and (A.5) yields the line integral isotropic expression (2.17) for evaluating the stress field of a dislocation:

$$\begin{aligned} \sigma_{ij}(\vec{\boldsymbol{x}}) = \frac{\mu b_n}{8\pi} \oint_L \Big[& R_{,mpp} \left(e_{jmn} dx'_i + e_{imn} dx'_j \right) \\ & + \frac{2}{1-\nu} e_{kmn} \left(R_{,ijm} - \delta_{ij} R_{,ppm} \right) dx'_k \Big] \end{aligned} \tag{A.6}$$

where the derivatives of the radius norm R are given by:

$$\begin{aligned} R_{,i} &= \frac{R_i}{R} \\ R_{,ij} &= \left(\delta_{ij} - \frac{R_i}{R} \frac{R_j}{R} \right) / R \\ R_{,ijk} &= \left[3 \frac{R_i}{R} \frac{R_j}{R} \frac{R_k}{R} - \left(\delta_{ij} \frac{R_k}{R} + \delta_{jk} \frac{R_i}{R} + \delta_{ki} \frac{R_j}{R} \right) \right] / R^2 \end{aligned} \tag{A.7}$$

The terms in (A.7) are obtained by successively differentiating the norm of the radius

vector given in (A.3) with respect to the field coordinates x_i. Thus, the explicit expressions of $R_{,i}$, $R_{,ij}$ and $R_{,ijk}$ are obtained as:

$$R_{,i} = \frac{\partial R}{\partial x_i} = \frac{\partial}{\partial x_i}\sqrt{R_j R_j} = \frac{2R_i}{2\sqrt{R_j R_j}} = \frac{R_i}{R} \tag{A.8}$$

$$\begin{aligned} R_{,ij} &= \frac{\partial R_{,i}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{R_i}{R}\right) \\ &= R_i \frac{\partial}{\partial x_j}\left(\frac{1}{R}\right) + \frac{1}{R}\frac{\partial R_i}{\partial x_j} \\ &= R_i \left(-\frac{1}{R^2}\frac{R_j}{R}\right) + \frac{1}{R}\delta_{ij} \\ &= \frac{1}{R}\left(\delta_{ij} - \frac{R_i R_j}{R^2}\right) \end{aligned} \tag{A.9}$$

$$\begin{aligned} R_{,ijk} &= \frac{\partial R_{,ij}}{\partial x_k} = \frac{\partial}{\partial x_k}\left(\frac{1}{R}\delta_{ij}\right) - \frac{\partial}{\partial x_k}\left(\frac{R_i R_j}{R^3}\right) \\ &= \delta_{ij}\left(-\frac{R_k}{R^3}\right) - R_i R_j \frac{\partial}{\partial x_k}\left(\frac{1}{R^3}\right) - \frac{1}{R^3}\frac{\partial}{\partial x_k}\left(R_i R_j\right) \\ &= -\delta_{ij}\frac{R_k}{R^3} - R_i R_j \left(\frac{-3}{R^4}\frac{R_k}{R}\right) - \frac{1}{R^3}\left(R_i \delta_{jk} + R_j \delta_{ik}\right) \\ &= \frac{3R_i R_j R_k}{R^5} - \delta_{ij}\frac{R_k}{R^3} - \delta_{jk}\frac{R_i}{R^3} - \delta_{ik}\frac{R_j}{R^3} \end{aligned} \tag{A.10}$$

Notice that the expression of $R_{,ijk}$ in (A.10) presents a symmetry between all indices, such that $R_{,ijk} = R_{,jki} = R_{,kji}$. Further, the terms $R_{,mpp}$ and $R_{,ppm}$ in equation (A.6) involve an implicit summation on index p, such that one can write:

$$R_{,mpp} = R_{,m11} + R_{,m22} + R_{,m33} \tag{A.11}$$

where the expression of $R_{,m11}$ is given from equation (A.10) as:

$$\begin{aligned} R_{,m11} &= \frac{3R_m R_1 R_1}{R^5} - \delta_{m1}\frac{R_1}{R^3} - \delta_{11}\frac{R_m}{R^3} - \delta_{m1}\frac{R_1}{R^3} \\ &= \frac{3R_m}{R^5}R_1^2 - \frac{R_m}{R^3} - 2\delta_{m1}\frac{R_1}{R^3} \end{aligned} \tag{A.12}$$

and that of $R_{,m22}$ and $R_{,m33}$ are similarly obtained by replacing index 1 by indices 2 and 3, respectively, such that $R_{,mpp}$ is expressed as:

$$\begin{aligned} R_{,mpp} &= R_{,m11} + R_{,m22} + R_{,m33} \\ &= \frac{3R_m}{R^5}(\underbrace{R_1^2 + R_2^2 + R_3^2}_{R^2}) - 3\frac{R_m}{R^3} - \frac{2}{R^3}\left(\delta_{m1}R_1 + \delta_{m2}R_2 + \delta_{m3}R_3\right) \end{aligned} \tag{A.13}$$

and since the first two terms cancel out, $R_{,mpp}$ finally writes:

$$R_{,mpp} = R_{,ppm} = -\frac{2R_m}{R^3} \tag{A.14}$$

With this, equation (A.6) can be evaluated at every field point $\vec{\boldsymbol{x}}$ in the medium. However, when $\vec{\boldsymbol{x}} = \vec{\boldsymbol{x'}}$ the expression of the stress field becomes singular. In other words, the stress field is not defined on the dislocation line itself when using this approach. To address this issue, Cai and co-workers [35] recently developed a non-singular approach that is presented in Appendix A.2.

Note further that expression (A.6) is given for a dislocation loop, i.e. when the integral is performed over a closed contour L. As detailed in Section 2.2.2, this results from the eigenstrain definition of the dislocation line that corresponds to the boundary L of the arbitrary defect surface $\vec{\boldsymbol{S}}$. However, since a dislocation loop can be regarded as a succession of parametric segments, equation (A.6) can be used to evaluate the stress on finite segments. Nevertheless, difficulties arise when evaluating the stress field along terminating segments, i.e. for segments whose at least one end finishes within the crystal. This is because the stress field for such segment cannot be determined uniquely [80, 14].

A.2 Non-singular formulation

The expression of the stress field induced by a dislocation given in equations (2.17) and (A.6) presents a singularity when evaluated on the dislocation line, i.e. for

$\vec{\boldsymbol{x}} = \vec{\boldsymbol{x'}}$. This is because the Green's function and its derivative become unbounded as $R \to 0$. Mathematically, this results from the description of the discontinuity in the displacement field $[\vec{\boldsymbol{u}}] = \vec{\boldsymbol{b}}$ across defect surface $\vec{\boldsymbol{S}}$ whose boundary defines the dislocation line in the eigenstrain-based approach: specifically, the dislocation core (i.e. the Burgers vector) is represented by a Dirac delta function that vanishes everywhere expect on the dislocation line. Obviously, such core singularity does not exist in real materials, and, despite taking complex forms in this region, the stress field remains finite.

Numerically, different approaches have been adopted to remove the stress singularity in the dislocation core. The most straightforward one consists in introducing a cut-off radius beyond which the stress field is assumed to be constant. However, this approach remains unsatisfactory since it corresponds to a crude approximation that is physically questionable.

To address this issue in a more rigorous way, Cai and co-workers recently proposed a consistent non-singular formulation whose main idea relies on introducing a spread of the Burgers vector, such as to remove the singularity on the core [35]. This approach is more consistent with the physics of dislocations and allows for a direct connection with molecular dynamics simulations: for instance, the distribution of the Burgers vector may be chosen such as to mimic arrangements of atoms in MD simulations. However, for the sake of simplicity, the authors first suggested to find an isotropic distribution of the Burgers vector such that the non-singular radius vector R_a expresses as:

$$R_a = \sqrt{R^2 + a^2} = \sqrt{R_i R_i + a^2} = \sqrt{(x_i - x'_i)(x_i - x'_i) + a^2} \tag{A.15}$$

where parameter a, denoting the core width, is introduced, and where R is the original radius vector of the classical theory whose expression is given in (A.3). Mathematically, R_a can be seen as the convolution between R and a particular isotropic

distribution of the Burgers vector. The obvious advantage of choosing the form for R_a given in (A.15) lies in the fact that its derivatives follow the expressions given in (A.7). Thus, one directly obtains:

$$\begin{aligned} R_{a,i} &= \frac{R_i}{R_a} \\ R_{a,ij} &= \left(\delta_{ij} - \frac{R_i}{R_a}\frac{R_j}{R_a}\right)/R_a \\ R_{a,ijk} &= \left[3\frac{R_i}{R_a}\frac{R_j}{R_a}\frac{R_k}{R_a} - \left(\delta_{ij}\frac{R_k}{R_a} + \delta_{jk}\frac{R_i}{R_a} + \delta_{ki}\frac{R_j}{R_a}\right)\right]/R_a^2 \end{aligned} \tag{A.16}$$

Note however that the non-singular stress field cannot be readily obtained from equation (A.6) by simply substituting R by R_a. Specifically, care must be taken when calculating $R_{a,mpp}$, that is given as:

$$\begin{aligned} R_{a,mpp} &= R_{a,m11} + R_{a,m22} + R_{a,m33} \\ &= \frac{3R_m}{R_a^5}(\underbrace{R_1^2 + R_2^2 + R_3^2}_{R^2 = R_a^2 - a^2}) - 3\frac{R_m}{R_a^3} - \frac{2}{R_a^3}\left(\delta_{m1}R_1 + \delta_{m2}R_2 + \delta_{m3}R_3\right) \\ &= -\left(\frac{3a^2}{R_a^5} + \frac{2}{R_a^3}\right)R_m \end{aligned} \tag{A.17}$$

in which a supplementary term involving a^2 appears with respect to $R_{,mpp}$. With this formulation, the non-singular stress field produced by a dislocation can be obtained at every point field $\vec{\boldsymbol{x}}$ by evaluating:

$$\begin{aligned} \sigma_{ij}(\vec{\boldsymbol{x}}) = \frac{\mu b_n}{8\pi}\oint_L \Bigg[& R_{a,mpp}\left(e_{jmn}dx'_i + e_{imn}dx'_j\right) \\ & + \frac{2}{1-\nu}e_{kmn}\left(R_{a,ijm} - \delta_{ij}R_{a,ppm}\right)dx'_k\Bigg] \end{aligned} \tag{A.18}$$

The absence of singularity facilitates the derivation of a general analytical formulation of expression (A.18). Thus, considering that dislocation loop L with Burgers vector $\vec{\boldsymbol{b}}$ is approximated by a prismatic loop composed of successive straight dislocation

segments, one has for each segment defined between end points $\vec{\boldsymbol{x}}_1$ and $\vec{\boldsymbol{x}}_2$:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = \frac{\mu b_n}{8\pi} \int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2} \Big[R_{a,mpp} \left(e_{jmn} dx'_i + e_{imn} dx'_j \right) + \frac{2}{1-\nu} e_{kmn} \left(R_{a,ijm} - \delta_{ij} R_{a,ppm} \right) dx'_k \Big] \tag{A.19}$$

A coordinate-independent analytical expression of (A.19) can be obtained as follows. As depicted in figure A.1, if one denotes by $\vec{\boldsymbol{x}}_0$ the orthogonal projection of field point $\vec{\boldsymbol{x}}$ on the dislocation line, the following segment parametric representation can be introduced:

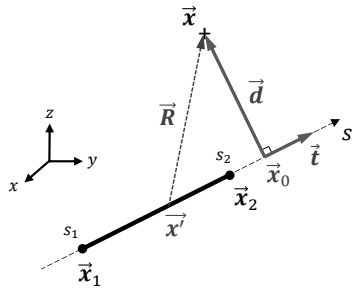


Figure A.1: Parametrization used to describe the straight dislocation segment defined between end points $\vec{\boldsymbol{x}}_1$ and $\vec{\boldsymbol{x}}_2$. Point $\vec{\boldsymbol{x}}_0$ is the orthogonal projection of field point $\vec{\boldsymbol{x}}$ on the dislocation line. With this representation, the unit tangent $\vec{\boldsymbol{t}}$ and vector $\vec{\boldsymbol{d}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}_0$ form an orthogonal basis, such that coordinate $\vec{\boldsymbol{x'}} = \vec{\boldsymbol{x}}_0 + s\vec{\boldsymbol{t}}$ spans segment 12 and the radius vector is defined by $\vec{\boldsymbol{R}} = \vec{\boldsymbol{d}} - s\vec{\boldsymbol{t}}$ with $s \in [s_1, s_2]$.

$$\vec{\boldsymbol{d}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}_0 \quad \text{with} \quad \vec{\boldsymbol{d}} \cdot \vec{\boldsymbol{t}} = 0 \tag{A.20}$$

$$\text{such that} \quad \vec{\boldsymbol{d}} = \vec{\boldsymbol{R}} - (\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{t}})\vec{\boldsymbol{t}} \tag{A.21}$$

where $\vec{\boldsymbol{d}}$ is the constant vector linking the field point $\vec{\boldsymbol{x}}$ to its projection $\vec{\boldsymbol{x}}_0$, such that $d = \|\vec{\boldsymbol{d}}\|$ is the distance from the field point to the segment, and $\vec{\boldsymbol{t}}$ is the unit tangent vector to the segment defined by:

$$\vec{\boldsymbol{t}} = \frac{\vec{\boldsymbol{x_2}} - \vec{\boldsymbol{x_1}}}{\|\vec{\boldsymbol{x_2}} - \vec{\boldsymbol{x_1}}\|} \tag{A.22}$$

With this, the dislocation segment $\vec{\boldsymbol{x'}}$ can be conveniently described with the following parametric representation:

$$\vec{\boldsymbol{x'}} = \vec{\boldsymbol{x}}_0 + s\vec{\boldsymbol{t}} \tag{A.23}$$

where the bounds for parameter s are given by:

$$s_1 = (\vec{\boldsymbol{x}}_1 - \vec{\boldsymbol{x}}_0) \cdot \vec{\boldsymbol{t}} \tag{A.24}$$

$$s_2 = (\vec{\boldsymbol{x}}_2 - \vec{\boldsymbol{x}}_0) \cdot \vec{\boldsymbol{t}} \tag{A.25}$$

With this setting, vectors $\vec{\boldsymbol{t}}$ and $\vec{\boldsymbol{d}}$ form an orthogonal basis such that the original radius vector $\vec{\boldsymbol{R}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}$ and its non-singular norm R_a can be expressed as:

$$\vec{\boldsymbol{R}} = \vec{\boldsymbol{d}} - s\vec{\boldsymbol{t}} \tag{A.26}$$

$$\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{R}} = (\vec{\boldsymbol{d}} - s\vec{\boldsymbol{t}}) \cdot (\vec{\boldsymbol{d}} - s\vec{\boldsymbol{t}}) = \vec{\boldsymbol{d}} \cdot \vec{\boldsymbol{d}} + s^2 \tag{A.27}$$

$$R_a = \sqrt{\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{R}} + a^2} = \sqrt{\vec{\boldsymbol{d}} \cdot \vec{\boldsymbol{d}} + s^2 + a^2} \tag{A.28}$$

such that R_a is solely a function of parameter s, both other terms being known constants associated with the segment. Thus, the first integral to be carried in expression (A.19) can be written as:

$$\begin{aligned} I_{im}(\vec{\boldsymbol{x}}) &= \int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2} R_{a,mpp} dx'_i = t_i \int_{s_1}^{s_2} R_{a,mpp} ds \\ &= -t_i \int_{s_1}^{s_2} \left(\frac{3a^2}{R_a^5} + \frac{2}{R_a^3} \right) R_m ds \end{aligned} \tag{A.29}$$

where the segment parametric representation (A.23) is used to perform the change of variable and $R_{a,mpp}$ is replaced by its expression in (A.17). R_m here denotes the m-th

component of the radius vector, such that one has:

$$\begin{aligned} R_m &= \vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{e}}_m = (\vec{\boldsymbol{d}} - s\vec{\boldsymbol{t}}) \cdot \vec{\boldsymbol{e}}_m \\ &= \underbrace{\vec{\boldsymbol{d}} \cdot \vec{\boldsymbol{e}}_m}_{d_m} - s \underbrace{\vec{\boldsymbol{t}} \cdot \vec{\boldsymbol{e}}_m}_{t_m} \end{aligned} \tag{A.30}$$

where d_m et t_m denote the m-th components of vectors $\vec{\boldsymbol{d}}$ and $\vec{\boldsymbol{t}}$, respectively. Integral (A.29) is therefore decomposed as:

$$I_{im}(\vec{\boldsymbol{x}}) = -t_i \left[d_m \int_{s_1}^{s_2} \left(\frac{3a^2}{R_a^5} + \frac{2}{R_a^3} \right) ds - t_m \int_{s_1}^{s_2} \left(\frac{3a^2}{R_a^5} + \frac{2}{R_a^3} \right) s ds \right] \tag{A.31}$$

Further, by defining J_{ij} to be the following line integral:

$$J_{ij} = \int_{s_1}^{s_2} \frac{s^i}{R_a^j} ds \tag{A.32}$$

expression (A.31) writes:

$$I_{im}(\vec{\boldsymbol{x}}) = -t_i \left[3a^2 d_m J_{05} + 2d_m J_{03} - 3a^2 t_m J_{15} - 2t_m J_{13} \right] \tag{A.33}$$

Similarly, the remaining integral to be evaluated in equation (A.19) writes:

$$\begin{aligned} K_{ijkm}(\vec{\boldsymbol{x}}) &= \int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2} R_{a,ijm} dx'_k = t_k \int_{s_1}^{s_2} R_{a,ijm} ds \\ &= t_k \int_{s_1}^{s_2} \left(\frac{3R_i R_j R_m}{R_a^5} - \delta_{ij} \frac{R_m}{R_a^3} - \delta_{jm} \frac{R_i}{R_a^3} - \delta_{im} \frac{R_j}{R_a^3} \right) ds \\ &= t_k \left[-(\delta_{ij} d_m + \delta_{jm} d_i + \delta_{im} d_j) J_{03} + (\delta_{ij} t_m + \delta_{jm} t_i + \delta_{im} t_j) J_{13} \right. \\ &\quad + 3(d_i d_j d_m) J_{05} - 3(d_i d_j t_m + d_i t_j d_m + t_i d_j d_m) J_{15} \\ &\quad \left. + 3(d_i t_j t_m + t_i d_j t_m + t_i t_j d_m) J_{25} - 3(t_i t_j t_m) J_{35} \right] \end{aligned} \tag{A.34}$$

Finally, inserting expressions (A.33) and (A.34) in (A.19) yields the following expression for the non-singular stress field:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = \frac{\mu b_n}{8\pi}\Bigg[e_{jmn}I_{im}(\vec{\boldsymbol{x}}) + e_{imn}I_{jm}(\vec{\boldsymbol{x}}) + \frac{2}{1-\nu}e_{kmn}\left(K_{ijkm}(\vec{\boldsymbol{x}}) - \delta_{ij}I_{km}(\vec{\boldsymbol{x}})\right)\Bigg] \tag{A.35}$$

where tensors $I_{im}(\vec{\boldsymbol{x}})$ and $K_{ijkm}(\vec{\boldsymbol{x}})$ depend on specific line integrals J_{ij} for which an analytical solution is given as follows:

$$\begin{aligned}
J_{03} &= \int_{s_1}^{s_2}\frac{1}{R_a^3}ds = \left.\frac{s}{(\vec{\boldsymbol{d}}\cdot\vec{\boldsymbol{d}}+a^2)R_a}\right|_{s_1}^{s_2}\\
J_{13} &= \int_{s_1}^{s_2}\frac{s}{R_a^3}ds = \left.-\frac{1}{R_a}\right|_{s_1}^{s_2}\\
J_{05} &= \int_{s_1}^{s_2}\frac{1}{R_a^5}ds = \left.\frac{2s^3}{3(\vec{\boldsymbol{d}}\cdot\vec{\boldsymbol{d}}+a^2)^2R_a^3}\right|_{s_1}^{s_2} + \left.\frac{s}{(\vec{\boldsymbol{d}}\cdot\vec{\boldsymbol{d}}+a^2)R_a^3}\right|_{s_1}^{s_2}\\
J_{15} &= \int_{s_1}^{s_2}\frac{s}{R_a^5}ds = \left.-\frac{1}{3R_a^3}\right|_{s_1}^{s_2}\\
J_{25} &= \int_{s_1}^{s_2}\frac{s^2}{R_a^5}ds = \left.\frac{s^3}{3(\vec{\boldsymbol{d}}\cdot\vec{\boldsymbol{d}}+a^2)R_a^3}\right|_{s_1}^{s_2}
\end{aligned} \tag{A.36}$$

Thus, equations (A.33) to (A.36) together provide a fully analytical solution for the stress field induced by a dislocation segment. Note besides that a vectorial form of this coordinate-independent expression might be convenient. Thus, equation (A.19) can be written as [35]:

$$\begin{aligned}
\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = &-\frac{\mu}{8\pi}\int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2}\left(\frac{2}{R_a^3}+\frac{3a^2}{R_a^5}\right)\left[(\vec{\boldsymbol{R}}\times\vec{\boldsymbol{b}})\otimes d\vec{\boldsymbol{x}}' + d\vec{\boldsymbol{x}}'\otimes(\vec{\boldsymbol{R}}\times\vec{\boldsymbol{b}})\right]\\
&+\frac{\mu}{4\pi(1-\nu)}\int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2}\left(\frac{1}{R_a^3}+\frac{3a^2}{R_a^5}\right)\left[(\vec{\boldsymbol{R}}\times\vec{\boldsymbol{b}})\cdot d\vec{\boldsymbol{x}}'\right]\boldsymbol{I}_2\\
&-\frac{\mu}{4\pi(1-\nu)}\int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2}\frac{1}{R_a^3}\left[(\vec{\boldsymbol{b}}\times d\vec{\boldsymbol{x}}')\otimes\vec{\boldsymbol{R}} + \vec{\boldsymbol{R}}\otimes(\vec{\boldsymbol{b}}\times d\vec{\boldsymbol{x}}')\right]\\
&+\frac{\mu}{4\pi(1-\nu)}\int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2}\frac{3}{R_a^3}\left[(\vec{\boldsymbol{R}}\times\vec{\boldsymbol{b}})\cdot d\vec{\boldsymbol{x}}'\right]\vec{\boldsymbol{R}}\otimes\vec{\boldsymbol{R}}
\end{aligned} \tag{A.37}$$

where $\boldsymbol{I}_2$ is the second-order identity tensor and operator $\otimes$ denotes the vector outer

product. Following the line parametrization introduced through equations (A.20) to (A.28), the tensorial form of (A.35) is given as a function of line integrals J_{ij} as [197]:

$$\begin{aligned}
\boldsymbol{\sigma}(\vec{\boldsymbol{x}}) = &\Big\{-\frac{\mu}{4\pi}\Big[(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\otimes\vec{\boldsymbol{t}}+\vec{\boldsymbol{t}}\otimes(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\Big] \\
&+\frac{\mu}{4\pi(1-\nu)}\Big[\Big\{\Big(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}}\Big)\cdot\vec{\boldsymbol{t}}\Big\}\boldsymbol{I}_2+(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\otimes\vec{\boldsymbol{d}}+\vec{\boldsymbol{d}}\otimes(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\Big]\Big\}\cdot J_{03} \\
&-\frac{\mu\nu}{4\pi(1-\nu)}\Big[(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\otimes\vec{\boldsymbol{t}}+\vec{\boldsymbol{t}}\otimes(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\Big]\cdot J_{13} \\
&+\Big\{-\frac{3\mu a^2}{8\pi}\Big[(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\otimes\vec{\boldsymbol{t}}+\vec{\boldsymbol{t}}\otimes(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\Big] \\
&+\frac{3\mu}{4\pi(1-\nu)}\Big[\Big\{(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\cdot\vec{\boldsymbol{t}}\Big\}\Big(a^2\boldsymbol{I}_2+\vec{\boldsymbol{d}}\otimes\vec{\boldsymbol{d}}\Big)\Big]\Big\}\cdot J_{05} \\
&+\Big\{\frac{3\mu a^2}{8\pi}\Big[(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\otimes\vec{\boldsymbol{t}}+\vec{\boldsymbol{t}}\otimes(\vec{\boldsymbol{t}}\times\vec{\boldsymbol{b}})\Big] \\
&-\frac{3\mu}{4\pi(1-\nu)}\Big[\Big\{(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\cdot\vec{\boldsymbol{t}}\Big\}\Big(\vec{\boldsymbol{t}}\otimes\vec{\boldsymbol{d}}+\vec{\boldsymbol{d}}\otimes\vec{\boldsymbol{t}}\Big)\Big]\Big\}\cdot J_{15} \\
&+\frac{3\mu}{4\pi(1-\nu)}\Big[\Big\{(\vec{\boldsymbol{d}}\times\vec{\boldsymbol{b}})\cdot\vec{\boldsymbol{t}}\Big\}(\vec{\boldsymbol{t}}\otimes\vec{\boldsymbol{t}})\Big]\cdot J_{25}
\end{aligned} \tag{A.38}$$

Thus, expressions (A.35), (A.37) or (A.38) provide an analytical non-singular formulation of the stress field of a dislocation segment within an infinite elastically isotropic medium. Practically, these expressions are directly used in regular DDD simulations to compute the internal stress state associated with the presence of the microstructure. However, as extensively discussed in Section 2.5, the computation of nodal velocities requires the calculation of segment forces. From equation (2.63), the force on segment 34 induced by dislocation segment 12 is expressed as:

$$\vec{\boldsymbol{f}}_{34}^{12} = l_{34}\int_0^1 (1-s)\Big[\Big(\boldsymbol{\sigma}^{12}\big((1-s)\vec{\boldsymbol{x}}_3+s\vec{\boldsymbol{x}}_4\big)\cdot\vec{\boldsymbol{b}}_{34}\Big)\times\vec{\boldsymbol{t}}_{34}\Big]\,ds \tag{A.39}$$

where $\vec{\boldsymbol{b}}_{34}$ and $\vec{\boldsymbol{t}}_{34}$ are the Burgers vector and the line direction of segment 34, respectively, and where $\boldsymbol{\sigma}^{12}(\vec{\boldsymbol{x}})$ denotes the stress field of dislocation 12 that is given by equations (A.35), (A.37) or (A.38). Thus, the calculation of segment forces requires an additional integration over segment 34 length. However, when stress field $\boldsymbol{\sigma}^{12}(\vec{\boldsymbol{x}})$

is non-singular and 34 is a straight segment, an analytical formulation of expression (A.39) can be obtained following a similar methodology as that described above. Full details and expressions for segment forces $\vec{\boldsymbol{f}}_{ij}^{kl}$ are provided in reference [5].

APPENDIX B

STRESS FIELD OF A DISLOCATION IN AN ANISOTROPIC MEDIUM

B.1 General anisotropic formulation

In this appendix, the numerical approaches to calculate the stress field of a dislocation loop in an anisotropic medium are presented. As derived in Section 2.2.2, the stress field induced by a static dislocation loop L at point $\vec{\boldsymbol{x}}$ in an infinite medium is given as a function of the first derivative of the Green's function $G_{ij,k}$ by the general line integral (2.14) first obtained by Mura [180, 181]:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl} \oint_L e_{lnh} C_{pqmn} G_{kp,q}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) b_m dx'_h \tag{B.1}$$

where e_{ijk} is the permutation tensor, $\vec{\boldsymbol{b}}$ is the Burgers vector and C_{ijkl} is the fourth-order elastic stiffness tensor expressed in the global frame $(\vec{\boldsymbol{e}}_1, \vec{\boldsymbol{e}}_2, \vec{\boldsymbol{e}}_3)$. In the case of an anisotropic elastic medium, the primary difficulty lies in the fact that no closed form expression of the Green's function exists. In general, it has been shown that its determination can be achieved by solving a sextic equation in terms of the Stroh eigenvalues following the method proposed by Ting and Lee [230], but the numerical evaluation of its derivatives cannot be directly obtained with this approach [146, 32]. Alternatively, the use of Fourier transforms techniques is probably the most straightforward approach to obtain an expression for G_{ij} [181]. Thus, as detailed in Section 3.3.1, the static Green's function for an infinite homogeneous linear elastic medium can be expressed from equation (3.26) as:

$$G_{ij}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) = \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} \widehat{G}_{ij}(\vec{\boldsymbol{\xi}}) e^{\mathrm{i}(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}) \cdot \vec{\boldsymbol{\xi}}} d\vec{\boldsymbol{\xi}} \tag{B.2}$$

where $\widehat{G}_{ij}(\vec{\boldsymbol{\xi}})$ is the static Green's function in the Fourier space that can be conveniently written from expression (3.21) as:

$$\widehat{G}_{ik}(\vec{\boldsymbol{\xi}}) = N_{ik}(\vec{\boldsymbol{\xi}}) D^{-1}(\vec{\boldsymbol{\xi}}), \quad \forall \vec{\boldsymbol{\xi}} \neq \vec{\boldsymbol{0}} \tag{B.3}$$

where $N_{ik}(\vec{\boldsymbol{\xi}})$ and $D^{-1}(\vec{\boldsymbol{\xi}})$ are the cofactor and the determinant of quantity $K_{ik} = C_{ijkl}\xi_l\xi_j$, respectively, and whose general expressions are given by relations (3.22) as:

$$\begin{aligned} N_{ij}(\vec{\boldsymbol{\xi}}) &= \frac{1}{2} e_{ikl} e_{jmn} K_{km}(\vec{\boldsymbol{\xi}}) K_{ln}(\vec{\boldsymbol{\xi}}) \\ D(\vec{\boldsymbol{\xi}}) &= e_{mnl} K_{m1}(\vec{\boldsymbol{\xi}}) K_{n2}(\vec{\boldsymbol{\xi}}) K_{l3}(\vec{\boldsymbol{\xi}}) \end{aligned} \tag{B.4}$$

where $\vec{\boldsymbol{\xi}}$ denotes the frequency coordinate in the Fourier space. By plugging the analytical expressions of $N_{ij}(\vec{\boldsymbol{\xi}})$ and $D^{-1}(\vec{\boldsymbol{\xi}})$ given in equations (3.40) for isotropic elasticity, the closed-form expression for G_{ij} given in equations (2.16) and (A.2) is directly retrieved. However, for general anisotropy, such direct expressions do not exist. Nevertheless, as proposed by Bacon and Barnett, an angular integral expression of the Green's function expressed in (B.2) can be obtained [14] as:

$$G_{kp}(\vec{\boldsymbol{R}}) = \frac{1}{4\pi^2 R} \int_0^{\pi} M^{-1}_{kp}(\vec{\boldsymbol{\xi}}) d\psi \tag{B.5}$$

where $\vec{\boldsymbol{R}} = \vec{\boldsymbol{x}} - \vec{\boldsymbol{x'}}$ is the radius vector of norm $R = \|\vec{\boldsymbol{R}}\|$, and where $M^{-1}_{kp}(\vec{\boldsymbol{\xi}}) = G_{kp}(\vec{\boldsymbol{\xi}})$ is used to conserve the original notation. Equation (B.5) is obtained by successively transforming the triple integration in (B.2) into a double integration on the unit sphere in the $\vec{\boldsymbol{\xi}}$-space, and into a line integral over the unit circle formed by the intersection of the unit sphere and the plane perpendicular to $\vec{\boldsymbol{R}}$. Full details of these integrations can be found in reference [181]. Consequently, by denoting $(\vec{\boldsymbol{e}}_{R1}, \vec{\boldsymbol{e}}_{R2})$

any arbitrary plane orthogonal to $\vec{\boldsymbol{R}}$, and $\vec{\boldsymbol{T}} = \vec{\boldsymbol{R}}/R$ the unit vector direction of $\vec{\boldsymbol{R}}$, $\vec{\boldsymbol{\xi}}(\psi)$ in equation (B.5) denotes the unit vector on plane $(\vec{\boldsymbol{e}}_{R1}, \vec{\boldsymbol{e}}_{R2})$ making an angle ψ with $\vec{\boldsymbol{e}}_{R1}$, and that satisfies $\vec{\boldsymbol{T}} \cdot \vec{\boldsymbol{\xi}} = 0$ by definition. With this, the first derivative of the anisotropic Green's function expressed in equation (B.1) is given by [17]:

$$G_{kp,q}(\vec{\boldsymbol{R}}) = \frac{1}{4\pi^2 R^2} \int_0^{\pi} \left(-T_q M_{kp}^{-1}(\vec{\boldsymbol{\xi}}) + \xi_q C_{jrnw} M_{kj}^{-1}(\vec{\boldsymbol{\xi}}) M_{np}^{-1}(\vec{\boldsymbol{\xi}})(\xi_r T_w + \xi_w T_r) \right) d\psi \tag{B.6}$$

that can be conveniently rewritten as:

$$G_{kp,q}(\vec{\boldsymbol{R}}) = \frac{g_{kpq}(\vec{\boldsymbol{T}})}{4\pi^2 R^2}$$

$$g_{kpq}(\vec{\boldsymbol{T}}) = \int_0^{\pi} \left(-T_q M_{kp}^{-1}(\vec{\boldsymbol{\xi}}) + \xi_q C_{jrnw} M_{kj}^{-1}(\vec{\boldsymbol{\xi}}) M_{np}^{-1}(\vec{\boldsymbol{\xi}})(\xi_r T_w + \xi_w T_r) \right) d\psi \tag{B.7}$$

where $g_{kpq}(\vec{\boldsymbol{T}})$ is the angular part of the Green's function derivative that is solely a function of the unit direction vector $\vec{\boldsymbol{T}} = \vec{\boldsymbol{R}}/R$.

No closed form solution exists for equation (B.7). As a result, the stress of a dislocation in an anisotropic medium is usually computed by successive numerical integrations of equations (B.7) and (B.1) [202, 108]. However, such integrations are extremely prohibitive in term of computational cost and render anisotropic regular DDD simulations impractical.

B.2 Spherical harmonics-based approach

Until very recently, the use of anisotropic elasticity to perform full-size DDD simulations was prohibited by the significant increase in computational time induced by the absence of analytical solution for the Green's function (B.7). To address this issue, Aubry and co-workers developed an approach based on spherical harmonics expansions as a way to numerically approximate equation (B.7) and decrease the computational cost of anisotropic calculations [12]. In this section, the method developed in reference [12] is presented.

The decomposition in spherical harmonics can be regarded as a numerical method to approximate a function on the unit sphere. Thus, the application of such technique to the calculation of the stress field of dislocations directly follows from the angular formulation of the derivative of the Green's function introduced by Barnett [17] in equation (B.7). In this expression, $g_{kpq}(\vec{\boldsymbol{T}})$ is solely a function of the unit vector $\vec{\boldsymbol{T}} = \vec{\boldsymbol{R}}/R$ whose union of all possible orientations spans the unit sphere. Consequently, if one denotes by (θ, φ) the coordinates of $\vec{\boldsymbol{T}}$ in the spherical coordinate system, one has $g_{kpq}(\vec{\boldsymbol{T}}) = g_{kpq}(\theta, \varphi)$, and the function g_{kpq} is suitable for a decomposition in spherical harmonics. Practically, it can be expanded in the following series:

$$g_{kpq}(\vec{\boldsymbol{T}}) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} g_{kpq}^{lm} Y_l^m(\vec{\boldsymbol{T}}) \tag{B.8}$$

which converges on the unit sphere, and where g_{kpq}^{lm} are the coefficients of expansion and $Y_l^m(\vec{\boldsymbol{T}})$ are the spherical harmonics. The main advantage of such decomposition lies in the fact that the expansion coefficients are independent of $\vec{\boldsymbol{T}}$ and express as:

$$g_{kpq}^{lm} = \int_0^{2\pi} \int_0^{\pi} g_{kpq}(\theta, \varphi) Y_l^{m*}(\theta, \varphi) \sin\theta d\theta d\varphi \tag{B.9}$$

where Y_l^{m*} denotes the complex conjugate of the spherical harmonic Y_l^m defined by:

$$\begin{aligned} Y_l^m(\theta, \varphi) &= (-1)^m M_l^m P_l^m(\cos\theta) e^{\mathrm{i}m\varphi} \\ \text{with} \quad M_l^m &= \sqrt{\frac{(2l+1)}{4\pi} \frac{(l-m)!}{(l+m)!}} \end{aligned} \tag{B.10}$$

where $(-1)^m$ is the Condon-Shortley phase factor and $P_l^m(\cos\theta)$ are the associated Legendre polynomials. For any $m \in [-l, l]$, the spherical harmonics Y_l^m on the unit sphere can be explicitly expressed in spherical coordinates as:

$$Y_l^m(\theta,\varphi) = (-1)^m M_l^{|m|} e^{\mathfrak{i}m\varphi} (\sin\theta)^{|m|} \sum_{k=0}^{\lfloor (l-|m|)/2 \rfloor} (-1)^k \frac{m!}{2l} \binom{l}{k} \binom{2l-2k}{l} \binom{l-2k}{m} (\cos\theta)^{l-|m|-2k} \tag{B.11}$$

Incidentally, the similitude between the graphical representation of the first spherical harmonics Y_l^m and the isocontours of the stress field of a dislocation is striking. In Cartesian coordinates, $Y_l^m(x,y,z)$ is obtained from (B.11) as:

$$Y_l^m(x,y,z) = (-1)^m M_l^{|m|} e^{\mathfrak{i}m\varphi} (x+\mathfrak{i}y)^m \sum_{k=0}^{\lfloor (l-|m|)/2 \rfloor} (-1)^k \frac{m!}{2l} \binom{l}{k} \binom{2l-2k}{l} \binom{l-2k}{m} z^{l-|m|-2k} \tag{B.12}$$

In equations (B.11) and (B.12), $\lfloor n \rfloor$ refers to the largest integer $\leq n$. Of particular interest, the expansion coefficients g_{kpq}^{lm} do not depend on the geometry of the dislocations but solely on the stiffness tensor C_{ijkl}, and are therefore identical for any dislocation segments. Consequently they can be conveniently computed once at the beginning of the simulation. Besides, equation (B.8) can be regarded as an interpolation of function $g_{kpq}(\vec{\boldsymbol{T}})$ making use of the Legendre interpolation polynomials $Y_l^m(\vec{\boldsymbol{T}})$ that form a complete set of orthonormal functions, and of the constant interpolation coefficients g_{kpq}^{lm}. Note that through their definition in equation (B.9), the expansion coefficients are analogous to Fourier coefficients. Following expression (B.12), when $\vec{\boldsymbol{T}}$ is expressed in the global frame $(\vec{\boldsymbol{e}}_1, \vec{\boldsymbol{e}}_2, \vec{\boldsymbol{e}}_3)$, $g_{kpq}(\vec{\boldsymbol{T}})$ in equation (B.8) writes:

$$g_{kpq}(\vec{\boldsymbol{T}}) = \sum_{l=0}^{\infty} \sum_{m=0}^{l} 2\mathfrak{R}\left(\left(\vec{\boldsymbol{T}} \cdot \vec{\boldsymbol{e}}_1 + \mathfrak{i}(\vec{\boldsymbol{T}} \cdot \vec{\boldsymbol{e}}_2) \right)^m g_{kpq}^{lm} \right) \sum_{s=0}^{\lfloor (l-m)/2 \rfloor} \frac{4\pi^2}{2} Q_s^{lm} (\vec{\boldsymbol{T}} \cdot \vec{\boldsymbol{e}}_3)^{l-m-2s} \tag{B.13}$$

where $\mathfrak{R}(x)$ denotes the real part of complex number x and scalar quantity Q_s^{lm} is defined as:

$$Q_s^{l0} = \frac{(-1)^s}{4\pi^2}\sqrt{\frac{(2l+1)}{4\pi}}\frac{1}{2^l}\binom{l}{s}\binom{2l-2s}{l} \quad \text{for } m=0$$

$$Q_s^{lm} = 2\frac{(-1)^{m+s}}{4\pi^2}\sqrt{\frac{(2l+1)}{4\pi}\frac{(l-m)!}{(l+m)!}}\frac{m!}{2^l}\binom{l}{s}\binom{2l-2s}{l}\binom{l-2s}{m} \quad \text{for } m>0 \quad \text{(B.14)}$$

Further, by substituting $\vec{\boldsymbol{T}} = \vec{\boldsymbol{R}}/R$ in equation (B.13) and by denoting $\vec{\boldsymbol{e}}_{12} = \vec{\boldsymbol{e}}_1 + \mathfrak{i}\vec{\boldsymbol{e}}_2$, the expansion of the Green's function derivative $G_{kp,q}(\vec{\boldsymbol{R}})$ in equation (B.7) in spherical harmonics series reads:

$$G_{kp,q}(\vec{\boldsymbol{R}}) = \sum_{l=0}^{\infty}\sum_{m=0}^{l}\sum_{s=0}^{\lfloor (l-m)/2 \rfloor} \Re\left(Q_s^{lm} g_{kpq}^{lm} \frac{(\vec{\boldsymbol{R}}\cdot\vec{\boldsymbol{e}}_{12})^m (\vec{\boldsymbol{R}}\cdot\vec{\boldsymbol{e}}_3)^{l-2s-m}}{R^{l-2s+2}}\right) \quad \text{(B.15)}$$

Note that the expression of the Green's function derivative in (B.15) contains even powers of $1/R$ which may be removed as the Green's function derivative is solely a function of odd powers of $1/R$. One can further notice that the terms that are functions of R solely depend on indices $l-2s$ and m. Therefore, by introducing the change of index $l = 2q+1$, expression (B.15) can be simplified as:

$$G_{kp,q}(\vec{\boldsymbol{R}}) = \sum_{q=0}^{\infty}\sum_{m=0}^{2q+1} \Re\left(S_{kpg}^{qm} \frac{(\vec{\boldsymbol{R}}\cdot\vec{\boldsymbol{e}}_{12})^m (\vec{\boldsymbol{R}}\cdot\vec{\boldsymbol{e}}_3)^{2q+1-m}}{R^{2q+3}}\right) \quad \text{(B.16)}$$

where only odd powers of $1/R$ remain and where the terms Q_s^{lm} and the expansion coefficients g_{kpq}^{lm} are conveniently lumped into S_{kpg}^{qm} as:

$$S_{kpg}^{qm} = \sum_{i=0}^{\infty} Q_i^{2i+1+2q,m} g_{kpg}^{2i+1+2q,m} \quad \text{(B.17)}$$

In practice, the summation over q in (B.16) is truncated to the order of the expansion q_{max}, whose choice is conditioned by the level of precision to be achieved, the computational cost and the ratio of anisotropy A (e.g. see equation (3.63)) of the medium [12]. For instance, the solution obtained for $G_{kp,q}(\vec{\boldsymbol{R}})$ for isotropic elasticity would be exact for $q_{max} = 1$, while it may require $q_{max} = 10$ for large ratios of anisotropy

($A > 7$). Combining equations (B.1) and (B.16), the stress field of a dislocation segment defined between vertices $\vec{\boldsymbol{x}}_1$ and $\vec{\boldsymbol{x}}_2$ and with Burgers vector $\vec{\boldsymbol{b}}$ in an anisotropic medium can be approximated by:

$$\sigma_{ij}(\vec{\boldsymbol{x}}) = C_{ijkl}e_{lnh}C_{pgmn}b_m \sum_{q=0}^{q_{max}} \sum_{m=0}^{2q+1} \Re \left(S_{kpg}^{qm} \int_{\vec{\boldsymbol{x}}_1}^{\vec{\boldsymbol{x}}_2} \frac{(\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{e}}_{12})^m (\vec{\boldsymbol{R}} \cdot \vec{\boldsymbol{e}}_3)^{2q+1-m}}{R^{2q+3}} dx'_h \right)$$

$$\text{with} \quad S_{kpg}^{qm} = \sum_{i=0}^{q_{max}-q} Q_i^{2i+1+2q,m} g_{kpg}^{2i+1+2q,m} \tag{B.18}$$

Since S_{kpg}^{qm} is independent of the geometry of the dislocations, it can be precomputed at the beginning of the simulation. Conveniently, the line integral from $\vec{\boldsymbol{x}}_1$ to $\vec{\boldsymbol{x}}_2$ can be computed analytically using recurrences relations when dealing with straight segments so as to obtain a computationally efficient approximation of dislocation stress fields and segment forces [12].

APPENDIX C

FINITE ELEMENT FORMULATION FOR NODAL DISLOCATION MOTION

C.1 Garlekin variational method for dislocation motion

The linear equation of motion (EOM) (2.49) expresses the general relation between dislocation force and velocity from thermodynamical considerations. Using a variational approach, a weak formulation can be obtained to solve for dislocation motion when the dislocation network is discretized into segments. From equation (2.49), the EOM relating the force $\vec{\boldsymbol{f}}$ and the velocity $\vec{\boldsymbol{v}}$ at each position along a closed dislocation loop can be expressed as [97]:

$$\oint \delta\vec{\boldsymbol{x}} \cdot (\vec{\boldsymbol{f}} - \boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}}) \, |d\vec{\boldsymbol{x}}| = 0 \tag{C.1}$$

where $\delta\vec{\boldsymbol{x}}$ is a virtual displacement of the dislocation line and the integration is performed along the entire dislocation line. Equation (C.1) holds when inertial effects are neglected. Knowing the force $\vec{\boldsymbol{f}}$ exerted at each point along the dislocation line and the drag matrix $\boldsymbol{\mathcal{B}}$, corresponding to the viscous resistance to dislocation motion, the velocity $\vec{\boldsymbol{v}}$ at each point $\vec{\boldsymbol{x}}$ along the dislocation line can be determined from equation (C.1). Using a discrete nodal representation, as detailed in Section 2.3, whereby each dislocation line is discretized into N_s segments ij delimited by nodes i and j at positions $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$, equation (C.1) rewrites:

$$\sum_{ij}^{N_s} \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} \delta\vec{\boldsymbol{x}}_{ij} \cdot \Big(\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij}) - \boldsymbol{\mathcal{B}}(\theta_{ij}(\vec{\boldsymbol{x}}_{ij}))\vec{\boldsymbol{v}}(\vec{\boldsymbol{x}}_{ij})\Big) \, |d\vec{\boldsymbol{x}}| = 0 \tag{C.2}$$

where $\vec{\boldsymbol{x}}$ is the vector that spans segment ij taking positions $\vec{\boldsymbol{x}}_{ij}(\vec{\boldsymbol{x}})$. Here, subscript

ij is used to specify that integral (C.2) is performed from node i to node j along the dislocation line connecting these two nodes. Therefore, throughout the rest of this section, note that subscript ij does not stand for the components of the vectors/matrices such that no implicit summation using Einstein's convention is meant. $\boldsymbol{\mathcal{B}}(\theta_{ij}(\vec{\boldsymbol{x}}_{ij}))$ is the drag matrix associated with the angle $\theta_{ij}(\vec{\boldsymbol{x}}_{ij})$ that makes dislocation segment ij at point $\vec{\boldsymbol{x}}_{ij}$ with its Burgers vector $\vec{\boldsymbol{b}}_{ij}$. Describing each dislocation segment ij as a parametric line spanned with coordinate $\vec{\boldsymbol{x}}$ $(\vec{\boldsymbol{x}}_i \leq \vec{\boldsymbol{x}} \leq \vec{\boldsymbol{x}}_j)$, the position $\vec{\boldsymbol{x}}_{ij}(\vec{\boldsymbol{x}})$ at coordinate $\vec{\boldsymbol{x}}$ along the segment can be obtained from the interpolation between N_d nodes as defined in (2.28) by:

$$\vec{\boldsymbol{x}}_{ij}(\vec{\boldsymbol{x}}) = \sum_{a}^{N_d} N_a(\vec{\boldsymbol{x}}) \vec{\boldsymbol{x}}_{ij}^{\,a} \tag{C.3}$$

where $N_a(\vec{\boldsymbol{x}})$ and $\vec{\boldsymbol{x}}_{ij}^{\,a}$ denote the interpolation function and the coordinate of the a-th node of segment ij, respectively. (Note that the use of generalized coordinates in (2.28) has been replaced by nodal coordinates in (C.3) for the sake of simplicity). For straight segments, the linear nodal interpolation is recovered by using the setting introduced by relations (2.33). Similarly to expression (C.3), one can write the virtual displacement $\delta\vec{\boldsymbol{x}}_{ij}$ and the velocity $\vec{\boldsymbol{v}}(\vec{\boldsymbol{x}}_{ij})$ as:

$$\delta\vec{\boldsymbol{x}}_{ij}(\vec{\boldsymbol{x}}) = \sum_{a}^{N_d} N_a(\vec{\boldsymbol{x}}) \delta\vec{\boldsymbol{x}}_{ij}^{\,a} \tag{C.4}$$

$$\vec{\boldsymbol{v}}(\vec{\boldsymbol{x}}_{ij}(\vec{\boldsymbol{x}})) = \sum_{a}^{N_d} N_a(\vec{\boldsymbol{x}}) \vec{\boldsymbol{v}}_{ij}^{\,a} \tag{C.5}$$

where $\vec{\boldsymbol{v}}_{ij}^{\,a}$ denotes the unknown velocity of node a belonging to segment ij. Using such parameterization and the expressions given in equations (C.3) to (C.5), equation (C.2) can be written as:

$$\sum_{ij}^{N_s} \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} \left(\sum_{a}^{N_d} N_a(\vec{\boldsymbol{x}}) \delta\vec{\boldsymbol{x}}_{ij}^{\,a} \right) \cdot \left(\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij}) - \boldsymbol{\mathcal{B}}(\theta_{ij}(\vec{\boldsymbol{x}}_{ij})) \sum_{b}^{N_d} N_b(\vec{\boldsymbol{x}}) \vec{\boldsymbol{v}}_{ij}^{\,b} \right) |d\vec{\boldsymbol{x}}| = 0 \tag{C.6}$$

From equation (C.6), one can express the following nodal quantities:

$$\vec{\boldsymbol{f}}_{ij}^{a} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_a(\vec{\boldsymbol{x}})\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij})\,|d\vec{\boldsymbol{x}}| \tag{C.7}$$

$$\boldsymbol{b}_{ij}^{ab} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_a(\vec{\boldsymbol{x}})N_b(\vec{\boldsymbol{x}})\boldsymbol{\mathcal{B}}(\theta_{ij}(\vec{\boldsymbol{x}}_{ij}))\,|d\vec{\boldsymbol{x}}| \tag{C.8}$$

where $\vec{\boldsymbol{f}}_{ij}^{a}$ denotes the nodal force on segment ij acting at node a and $\boldsymbol{b}_{ij}^{ab}$ is the drag matrix of segment ij associated with nodes a and b. With expressions (C.7) and (C.8), system (C.6) rewrites:

$$\sum_{ij}^{N_s}\left[\sum_{a}^{N_d}\delta\vec{\boldsymbol{x}}_{ij}^{a}\cdot\left(\vec{\boldsymbol{f}}_{ij}^{a}-\sum_{b}^{N_d}\boldsymbol{b}_{ij}^{ab}\vec{\boldsymbol{v}}_{ij}^{b}\right)\right]=0 \tag{C.9}$$

The discretization of continuous dislocation lines induces that dislocation segments share some common nodes when connected to each other. As a result, the two quantities in equation (C.9) can be assembled as done in a classical finite element approach:

$$\sum_{ij}^{N_s}\sum_{a}^{N_d}\delta\vec{\boldsymbol{x}}_{ij}^{a}\cdot\vec{\boldsymbol{f}}_{ij}^{a}=\sum_{k}^{N_t}\delta\vec{\boldsymbol{X}}_k\vec{\boldsymbol{F}}_k \tag{C.10}$$

$$\sum_{ij}^{N_s}\sum_{a}^{N_d}\sum_{b}^{N_d}\delta\vec{\boldsymbol{x}}_{ij}^{a}\cdot(\boldsymbol{b}_{ij}^{ab}\vec{\boldsymbol{v}}_{ij}^{b})=\sum_{k}^{N_t}\sum_{l}^{N_t}\delta\vec{\boldsymbol{X}}_k\boldsymbol{B}_{kl}\vec{\boldsymbol{V}}_l \tag{C.11}$$

where $N_t \leq N_s \times N_d$ is the total number of nodes used in the description of the entire dislocation line. Therefore system (C.9) reduces to:

$$\sum_{k}^{N_t}\delta\vec{\boldsymbol{X}}_k\cdot\left(\vec{\boldsymbol{F}}_k-\sum_{l}^{N_t}\boldsymbol{B}_{kl}\vec{\boldsymbol{V}}_l\right)=0 \tag{C.12}$$

Since equation (C.12) must hold for any arbitrary virtual line displacement $\delta\vec{\boldsymbol{X}}_k$, the system of equations finally writes:

$$\vec{\boldsymbol{F}}_k = \sum_{l}^{N_t} \boldsymbol{B}_{kl} \vec{\boldsymbol{V}}_l \quad \forall \text{ node } k, \quad \text{i.e.} \quad \boldsymbol{F} = \boldsymbol{B}\boldsymbol{V} \tag{C.13}$$

In equation (C.13), $\vec{\boldsymbol{F}}_k$ and $\vec{\boldsymbol{V}}_k$ respectively denote the force and velocity at node k, and $\boldsymbol{B}_{kl}$ denotes the drag matrix pertaining to nodes k and l. Therefore, expression (C.13) corresponds to the full system of equation of size N_t necessary to describe the motion of a dislocation line discretized into N_s segments and governed by the linear mobility law (2.49). Note however that $\boldsymbol{B}_{ij}$ is only defined along dislocation segments, i.e. when nodes i and j are connected. Consequently, system (C.13) is generally extremely sparse. With that in mind, the N_t nodal forces $\vec{\boldsymbol{F}}_i$ can be expressed as:

$$\vec{\boldsymbol{F}}_i = \sum_{j} \boldsymbol{B}_{ij} \vec{\boldsymbol{V}}_j \quad \forall \text{ node } i, \ \forall \text{ node } j \text{ connected to node } i, \text{ including } i = j \tag{C.14}$$

With equation (C.14), the velocity of each dislocation node can be determined from the forces acting at each node. The assembly and the computation of nodal forces are presented in Appendix C.2 while the construction of the resistivity matrix $\boldsymbol{B}$ is detailed in Appendix C.3.

C.1.1 Equation of motion accounting for inertial effects

At high strain rates, inertial effects on dislocation motion become important and need to be accounted for in order to obtain accurate predictions [248]. (Notice that relativistic effects are usually neglected in DDD simulations.) To directly account for inertial effects, equation (C.1) may be replaced by:

$$\oint \delta\vec{\boldsymbol{x}} \cdot (\vec{\boldsymbol{f}} - \boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}} - m\vec{\boldsymbol{a}}) \, |d\vec{\boldsymbol{x}}| = 0 \tag{C.15}$$

where $\vec{\boldsymbol{a}}$ denotes the acceleration and m is the effective dislocation mass per unit length. Taking advantage of time discretization used in the DDD framework (see

Section 2.1.1), the acceleration associated with the dislocation line can be approximated using a backward finite difference scheme, such that:

$$\vec{\boldsymbol{a}} = \frac{d\vec{\boldsymbol{v}}}{dt} = \frac{\vec{\boldsymbol{v}} - \vec{\boldsymbol{v}}_{prev}}{dt} \tag{C.16}$$

where $\vec{\boldsymbol{v}}_{prev}$ designates the velocity at previous time step, $\vec{\boldsymbol{v}}$ is the unknown velocity to be solved for from equation (C.15), and dt denotes the current time step increment of the DDD simulation. Inserting expression (C.16) into equation (C.15) leads to the following EOM:

$$\oint \delta\vec{\boldsymbol{x}} \cdot \left(\vec{\boldsymbol{f}} - \boldsymbol{\mathcal{B}}\vec{\boldsymbol{v}} - \frac{m}{dt}(\vec{\boldsymbol{v}} - \vec{\boldsymbol{v}}_{prev}) \right) |d\vec{\boldsymbol{x}}| = 0 \tag{C.17}$$

Rearranging the terms in (C.17) yields the alternative EOM for high strain rates:

$$\oint \delta\vec{\boldsymbol{x}} \cdot \left(\vec{\boldsymbol{f}} + \frac{m}{dt}\vec{\boldsymbol{v}}_{prev} - \left(\boldsymbol{\mathcal{B}} + \frac{m}{dt}\boldsymbol{I}_3 \right) \vec{\boldsymbol{v}} \right) |d\vec{\boldsymbol{x}}| = 0 \tag{C.18}$$

$$\oint \delta\vec{\boldsymbol{x}} \cdot (\vec{\boldsymbol{f}}^* - \boldsymbol{\mathcal{B}}^*\vec{\boldsymbol{v}}) \, |d\vec{\boldsymbol{x}}| = 0 \tag{C.19}$$

In cases where inertial effects are to be accounted for, equation (C.1) can be substituted by equation (C.19), for which the corresponding inertial forces $\vec{\boldsymbol{f}}^*$ and inertial $\boldsymbol{\mathcal{B}}^*$ drag matrices are expressed with respect to the actual forces $\vec{\boldsymbol{f}}$ and drag matrices $\boldsymbol{\mathcal{B}}$ as follows:

$$\begin{aligned} \vec{\boldsymbol{f}}^* &= \vec{\boldsymbol{f}} + \frac{m}{dt}\vec{\boldsymbol{v}}_{prev} \\ \boldsymbol{\mathcal{B}}^* &= \boldsymbol{\mathcal{B}} + \frac{m}{dt}\boldsymbol{I}_2 \end{aligned} \tag{C.20}$$

In equation (C.20), $\boldsymbol{I}_2$ denotes the 3×3 identity matrix and m is the effective mass per unit length associated with the dislocation line. An expression of the latter was derived by Hirth et al. [110]. Considering pure screw and pure edge dislocations,

their effective mass, respectively denoted m_s and m_e, can be expressed as:

$$
\begin{aligned}
m_s &= \frac{W_0}{v^2}\left(-\gamma^{-1}+\gamma^{-3}\right) \\
m_e &= \frac{W_0 C^2}{v^4}\left(-8\gamma_l - 20\gamma_l^{-1} + 4\gamma_l^{-3} + 7\gamma + 25\gamma^{-1} - 11\gamma^{-3} + 3\gamma^{-5}\right)
\end{aligned}
\tag{C.21}
$$

where γ_l and γ are two quantities related the longitudinal C_l and transversal C sound velocities, respectively, such that:

$$
\begin{aligned}
\gamma_l &= \left(1 - v^2/C_l^2\right)^{1/2} \\
\gamma &= \left(1 - v^2/C^2\right)^{1/2}
\end{aligned}
\tag{C.22}
$$

where v is the magnitude of the velocity of the dislocation line. To avoid equation (C.19) from being non-linear, the velocity in equation (C.22) can be taken as that of the previous step $\vec{\boldsymbol{v}}_{prev}$. In equations (C.21), W_0 is the rest energy factor of dislocation lines, expressed as:

$$
W_0 = \frac{\mu b^2}{4\pi}\ln\left(\frac{R}{r_0}\right) \tag{C.23}
$$

where R and r_0 are the inner and outer cut-off radii in the integration of energy terms, generally chosen such that $W_0 = \mu b^2$. With that, the effective mass of screw dislocation at zero velocity expresses as $\mu b^2/C^2$, and the rest energy of an edge dislocation is $\mu b^2/C^2(1 + C^4/C_l^4)$. As a simple approximation, the effective mass m of a mixed dislocation line can be taken as a simple interpolation from the effective masses of pure edge and screw dislocations given in equations (C.21), such that:

$$
m = m_e \sin^2\theta + m_s \cos^2\theta \tag{C.24}
$$

where θ is the angle between the dislocation line tangent and its Burgers vector. Finally, it follows from equations (C.20) that the nodal forces and resistivity matrices

when accounting for inertia are expressed from (C.7) and (C.8) as:

$$\vec{\boldsymbol{f}}_{ij}^{a} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_a(\vec{\boldsymbol{x}}) \left[\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij}) + \frac{m}{dt} \vec{\boldsymbol{v}}_{prev}(\vec{\boldsymbol{x}}_{ij}) \right] |d\vec{\boldsymbol{x}}| \tag{C.25}$$

$$\boldsymbol{b}_{ij}^{ab} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_a(\vec{\boldsymbol{x}}) N_b(\vec{\boldsymbol{x}}) \left[\boldsymbol{\mathcal{B}}(\theta_{ij}(\vec{\boldsymbol{x}}_{ij})) + \frac{m}{dt} \boldsymbol{I}_3 \right] |d\vec{\boldsymbol{x}}| \tag{C.26}$$

Finally, assembly of equations (C.25) and (C.26) with the finite element approach to obtain system (C.14) for the nodal velocities at high-strain rates is performed with equations (C.10) and (C.11).

C.2 Nodal forces and segment elastic interactions

C.2.1 Assembly of nodal forces

Expression (C.14) defining the system of equations to solve for the nodal velocities $\vec{\boldsymbol{V}}_j$ requires the evaluation and the assembly of the nodal forces $\vec{\boldsymbol{F}}_i$. For each dislocation node, the latter is assembled according to the classical finite element method and is given from (C.10) as:

$$\vec{\boldsymbol{F}}_i = \sum_j \vec{\boldsymbol{f}}_{ij}^{i} \tag{C.27}$$

where $\vec{\boldsymbol{f}}_{ij}^{i}$ is the contribution at node i of the force on segment ij, and the summation is performed over every node j connected to node i. Expressions of $\vec{\boldsymbol{f}}_{ij}^{i}$ are given from equation (C.7) when inertial effects are neglected, and from (C.25) when these are accounted for. However, for the rest of this section, for the sake of simplicity, the expression of the forces will be given for the case when inertial effects are disregarded.

As defined in equation (C.7), $\vec{\boldsymbol{f}}_{ij}^{a}$ corresponds to the force on segment ij acting at node a, where index a takes the values corresponding to the segment end nodes, i.e. i and j. When using the linear interpolation between the end nodes defined by relations (2.28) and (2.33) such as to consider dislocation segments as straight lines,

the nodal force (C.7) expresses as:

$$\vec{\boldsymbol{f}}_{ij}^{a} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_a(\vec{\boldsymbol{x}})\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij})\,|d\vec{\boldsymbol{x}}| = l_{ij}\int_0^1 N_a(s)\vec{\boldsymbol{f}}(\vec{\boldsymbol{x}}_{ij}(s))\,ds \tag{C.28}$$

$$\text{with } |d\vec{\boldsymbol{x}}| = \|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\|ds = l_{ij}ds \tag{C.29}$$

where $l_{ij} = \|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\|$ denotes the length of segment ij delimited by end nodes i and j at positions $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$, respectively, and where $s \in [0,1]$ is the curvilinear abscissa used for the parametric representation of the segment. Recall from (2.33) that the linear interpolation function at first node i is given by $N_i(s) = 1 - s$ and that at second node j is given by $N_j(s) = s$. For the sake of simplicity, when superscript a takes the value of the first node i, exponent i will be omitted such that $\vec{\boldsymbol{f}}_{ij}^{\,i} = \vec{\boldsymbol{f}}_{ij}$. Similarly, by virtue of the symmetry induced by the linear interpolation, when a refers to the second node j, we will denote $\vec{\boldsymbol{f}}_{ij}^{\,j} = \vec{\boldsymbol{f}}_{ji}$. With this notation, equation (C.27) rewrites:

$$\vec{\boldsymbol{F}}_i = \sum_j \vec{\boldsymbol{f}}_{ij} \tag{C.30}$$

and the force contribution $\vec{\boldsymbol{f}}_{ij}$ at node i of segment ij can be interchangeably computed from (C.28) as:

$$\begin{aligned} \vec{\boldsymbol{f}}_{ij} = \vec{\boldsymbol{f}}_{ij}^{\,i} &= l_{ij}\int_0^1 (1-s)\vec{\boldsymbol{f}}\,(\vec{\boldsymbol{x}}_{ij}(s))\,ds \\ \text{or} \quad \vec{\boldsymbol{f}}_{ij} = \vec{\boldsymbol{f}}_{ji}^{\,i} &= l_{ji}\int_0^1 s\vec{\boldsymbol{f}}\,(\vec{\boldsymbol{x}}_{ji}(s))\,ds \end{aligned} \tag{C.31}$$

As explained in Section 2.4, in the regular nodal DDD approach, the force $\vec{\boldsymbol{f}}$ in equations (C.31) acting on a dislocation segment encompasses two stress contributions, internal and external, when defined according to equation (2.47). The internal stress state arises from the elastic stress field induced by the presence of the other dislocations while the external contribution pertains to the imposed loading. Therefore,

the Peach-Kohler force per unit length (see equation (2.46)) at any point $\vec{\boldsymbol{x}}_{ij}$ along dislocation segment ij is generally written as:

$$\vec{\boldsymbol{f}}^{pk}(\vec{\boldsymbol{x}}_{ij}) = \left(\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij}\right) \times \vec{\boldsymbol{t}}_{ij} \tag{C.32}$$

where $\vec{\boldsymbol{b}}_{ij}$ and $\vec{\boldsymbol{t}}_{ij}$ are the Burgers vector and the unit tangent to segment ij, respectively. Here, the total stress tensor $\boldsymbol{\sigma}$ encompasses both the internal $\boldsymbol{\sigma}^{int}$ and the external $\boldsymbol{\sigma}^{ext}$ contributions. As detailed in Section 2.4.2, both these contributions are generally superimposed in the regular approach to define the effective total stress $\boldsymbol{\sigma}$ driving dislocation motion:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{int} + \boldsymbol{\sigma}^{ext} \tag{2.41 repeated}$$

Note here that an alternative approach to define the total stress without invoking the superposition principle is discussed in Chapter 3. In practice, to account for the lattice friction stress, the effective force $\vec{\boldsymbol{f}}_{ij}$ integrated along dislocation segment ij in equations (C.31) is obtained as:

$$\vec{\boldsymbol{f}}_{ij} = \begin{cases} \vec{\boldsymbol{f}}_{ij}^{pk} - \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \cdot \vec{\text{dir}}\left(\vec{\boldsymbol{f}}_{ij}^{pk}\right) & \text{if } \left|\vec{\boldsymbol{f}}_{ij}^{pk}\right| > \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \\ 0 & \text{if } \left|\vec{\boldsymbol{f}}_{ij}^{pk}\right| \leq \left|\vec{\boldsymbol{f}}_{ij}^{fric}\right| \end{cases} \tag{C.33}$$

where $\vec{\boldsymbol{f}}_{ij}^{fric}$ is the force arising from the lattice friction resistance (here function $\vec{\text{dir}}(\vec{\boldsymbol{v}})$ denotes the unit vector pointing in the direction of vector $\vec{\boldsymbol{v}}$ such that the friction force is opposed to dislocation movement) and $\vec{\boldsymbol{f}}_{ij}^{pk}$ is the Peach-Koehler force obtained from integration of expression (C.32) along the segment length. Here it is interesting to notice that by combining equations (C.32) and (2.41), the Peach-Kohler force is obtained as the sum of two contributions:

$$\vec{\boldsymbol{f}}_{ij}^{pk} = \vec{\boldsymbol{f}}_{ij}^{int} + \vec{\boldsymbol{f}}_{ij}^{ext} \tag{C.34}$$

where $\vec{\boldsymbol{f}}_{ij}^{ext}$ corresponds to the external applied force whose calculation is detailed in Appendix C.2.4, and $\vec{\boldsymbol{f}}_{ij}^{int}$ corresponds to the force arising from the internal stress fields induced by all dislocations in the medium. Following the framework introduced in Section 2.4.3, the internal stresses $\boldsymbol{\sigma}^{int}$ at each point of the simulation volume in regular DDD codes are calculated by virtue of the superposition principle, whereby the contributions of the elastic stress field of each dislocation segment computed in an infinite medium are added. Therefore, the force $\vec{\boldsymbol{f}}_{ij}^{int}$ on segment ij resulting from the internal stresses can be expressed as:

$$\vec{\boldsymbol{f}}_{ij}^{int} = \sum_{kl} \vec{\boldsymbol{f}}_{ij}^{kl} \tag{C.35}$$

where $\vec{\boldsymbol{f}}_{ij}^{kl}$ denotes the force acting at node i induced by the stress field of dislocation segment kl on segment ij. Note that the sum is performed over all dislocation segments kl present in the volume, including segment ij itself. As a matter of fact, $\vec{\boldsymbol{f}}_{ij}^{ij} = \vec{\boldsymbol{f}}_{ij}^{s}$ corresponds to the self-force of segment ij, i.e. the force of the segment on itself. To avoid singularities, the self-force is usually computed separately (see Appendix C.2.3), such that equation (C.35) is decomposed as:

$$\vec{\boldsymbol{f}}_{ij}^{int} = \vec{\boldsymbol{f}}_{ij}^{s} + \sum_{kl \neq ij} \vec{\boldsymbol{f}}_{ij}^{kl} \tag{C.36}$$

where the summation over the segment-segment elastic interaction forces $\vec{\boldsymbol{f}}_{ij}^{kl}$ is performed on every segment kl different from ij. From equation (C.36) it clearly appears that the computation of the force on each dislocation segment requires the evaluation of the stress field induced by all other dislocations present in the volume. Therefore, the computation of all nodal forces is a $\mathcal{O}(N_{seg}^2)$ process where N_{seg} is the total number of dislocation segments in the simulation volume, which can become extremely computationally expensive. Thus, special care needs to be taken so as to implement efficient methods for the computation of the segment elastic forces $\vec{\boldsymbol{f}}_{ij}^{kl}$. While

segment-segment elastic interactions forces $\vec{\boldsymbol{f}}_{ij}^{kl}$ are detailed in Section 2.5.3, approximations to improve the efficiency of their computation are presented in Section 2.9.2.

C.2.2 Segment-segment elastic interactions

See Section 2.5.3.

C.2.3 Self force

The self-force $\vec{\boldsymbol{f}}_{ij}^{s}$ corresponds to the force exerted by a segment ij on itself. The general expression of the self-force can be readily deduced from equation (2.64) as:

$$\vec{\boldsymbol{f}}_{ij}^{s} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\boldsymbol{\sigma}^{ij}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| \tag{C.37}$$

where $\boldsymbol{\sigma}^{ij}$ denotes the stress field of dislocation segment ij. Therefore, the evaluation of the self-force involves the integration of the stress field produced by the segment along itself, hence its name. The precise evaluation of the self-force is a very important aspect of DDD simulations as it directly affects the bowing of dislocations, and hence plays a major role in the activation of dislocation glide. However, in the general expression given from the classical theory in equation (2.14), the stress field of a dislocation is singular in its core, i.e. is not defined along the dislocation segment itself, such that contribution (C.37) cannot be directly evaluated. To circumvent this, different expressions of the self-force based on the dislocation local curvature have been proposed [71, 87, 98, 97]. In this case, the self-force is referred to as the line tension, corresponding to the restoring force induced by the increase in energy associated with the increase in line length of a bowing segment (e.g. Frank-Read source). However, the accuracy of curvature-based approaches are closely tied to the parametric line representation, and are inconsistent with the use of straight segments.

In the case of isotropic elasticity, the non-singular approach developed by Cai

and coworkers [35] (see Appendix A.2) allows for an analytical expression of the self-force (C.37). The latter is directly derived from the general closed-form solution for computing the forces between two parallel segments, and is simplified due to the fact that both segments are identical in this specific case. As a result, its expression simplifies to the following for isotropic elasticity [5]:

$$\vec{\boldsymbol{f}}_{ij}^{s} = -\frac{\mu}{4\pi}\left[\vec{\boldsymbol{t}}_{ij} \times \left(\vec{\boldsymbol{t}}_{ij} \times \vec{\boldsymbol{b}}_{ij}\right)\right]\left(\vec{\boldsymbol{t}}_{ij} \cdot \vec{\boldsymbol{b}}_{ij}\right) \left[\frac{\nu}{1-\nu}\left(\ln\left[\frac{l_a + l_{ij}}{a}\right] - 2\frac{l_a + a}{l_{ij}}\right) - \frac{\left(l_a - a\right)^2}{2 l_a l_{ij}}\right] \quad \text{(C.38)}$$

where μ and ν are the shear modulus and Poisson's ratio, a is the dislocation core width and $l_a = \sqrt{l_{ij}^2 + a^2}$. In this formulation, the core width radius a is a key parameter that governs the stiffness of dislocation bowing. Thus, the choice of a is usually calibrated such as to obtain a correct activation stress for dislocation sources. Furthermore, it clearly appears from equation (C.38) that the contribution of the self-force of segment ij on node j can be directly computed as $\vec{\boldsymbol{f}}_{ji}^{s} = -\vec{\boldsymbol{f}}_{ij}^{s}$.

When dealing with anisotropic elasticity, the self-force can be efficiently computed using the Willis-Steeds-Lothe formalism and a cut-off radius to remove core singularities [257, 264].

C.2.4 Applied force

The applied force $\vec{\boldsymbol{f}}_{ij}^{ext}$ corresponds to the contribution of the external stress to the Peach-Kohler force, as defined in equations (C.32) and (2.41). As introduced in Section 2.4.4, the external stress $\boldsymbol{\sigma}^{ext}$ results from the imposed loading and boundary conditions, and the different methods that can be used for its evaluation are presented in Section 2.8.2. The general expression for the applied force $\vec{\boldsymbol{f}}_{ij}^{ext}$ on segment ij can be readily obtained from the Peach-Kohler force expression as:

$$\vec{\boldsymbol{f}}_{ij}^{ext} = \int_{\vec{\boldsymbol{x}}_i}^{\vec{\boldsymbol{x}}_j} N_i(\vec{\boldsymbol{x}}_{ij}) \left[\left(\boldsymbol{\sigma}^{ext}(\vec{\boldsymbol{x}}_{ij}) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] |d\vec{\boldsymbol{x}}| \tag{C.39}$$

When using the linear interpolation defined in (2.33) such as to consider straight dislocation segments, expression (C.39) reduces to:

$$\vec{\boldsymbol{f}}_{ij}^{ext} = \int_0^1 (1-s) \left[\left(\boldsymbol{\sigma}^{ext}(\vec{\boldsymbol{x}}_{ij}(s)) \cdot \vec{\boldsymbol{b}}_{ij} \right) \times \vec{\boldsymbol{t}}_{ij} \right] ds \tag{C.40}$$

where $\boldsymbol{\sigma}^{ext}$ is integrated along dislocation segment ij with Burgers vector $\vec{\boldsymbol{b}}_{ij}$ and unit tangent $\vec{\boldsymbol{t}}_{ij}$. In the case where the external stress is uniform over the simulation volume (see Section 2.8.2), the integration in (C.40) reduces to the following analytical form:

$$\vec{\boldsymbol{f}}_{ij}^{ext} = \frac{1}{2} l_{ij} \left(\left[\boldsymbol{\sigma}^{ext} \cdot \vec{\boldsymbol{b}}_{ij} \right] \times \vec{\boldsymbol{t}}_{ij} \right) \tag{C.41}$$

C.3 Determination of nodal velocities

C.3.1 Assembly of the resistivity matrix

The determination of nodal velocities of dislocation segments at each time step is achieved by solving the system of equations defined in (C.14):

$$\vec{\boldsymbol{F}}_i = \sum_j \boldsymbol{B}_{ij} \vec{\boldsymbol{V}}_j \quad \forall \text{ node } i, \ \forall \text{ node } j \text{ connected to node } i, \text{ and for } i = j$$

(C.14 repeated)

where $\vec{\boldsymbol{V}}_j$ is the velocity of dislocation node j, $\vec{\boldsymbol{F}}_i$ is the force at node i, and $\boldsymbol{B}_{ij}$ is the resistivity matrix associated with the viscous drag of segment ij. Once the dislocation nodal forces $\vec{\boldsymbol{F}}_i$ have been obtained through the procedure detailed in Appendix C.2, matrix $\boldsymbol{B}_{ij}$ is assembled using a classical finite element approach. From expression (C.11) one obtains:

$$\boldsymbol{B}_{ij} = \boldsymbol{b}_{ij}^{ij} \quad \text{if } i \neq j$$
$$\boldsymbol{B}_{ii} = \sum_{k} \boldsymbol{b}_{ik}^{ii} \quad \text{if } i = j,\ \forall \text{ node } k \neq i \text{ connected to node } i \tag{C.42}$$

where $\boldsymbol{b}_{ij}^{ab}$ is the nodal quantity defined in equation (C.8). When straight lines are used to describe dislocation segments between connected nodes by using the linear interpolation defined in (2.33), $\boldsymbol{b}_{ij}^{ab}$ can be written as:

$$\boldsymbol{b}_{ij}^{ab} = l_{ij} \int_0^1 N_a(s) N_b(s) \boldsymbol{\mathcal{B}}(\theta_{ij}(s)) ds \tag{C.43}$$
$$\text{with } |d\vec{\boldsymbol{x}}| = \|\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i\| ds = l_{ij} ds \tag{C.44}$$

Note that in expression (C.43), the drag matrix $\boldsymbol{\mathcal{B}}(\theta_{ij})$ is now solely a function of the orientation $\theta_{ij} = \arccos\left(\frac{\vec{\boldsymbol{b}}_{ij}}{\|\vec{\boldsymbol{b}}_{ij}\|} \cdot \vec{\boldsymbol{t}}_{ij}\right)$ of the dislocation segment ij, which is constant throughout the entire segment when using a linear interpolation. With this, sub-matrices $\boldsymbol{b}_{ij}^{ab}$ pertaining to the drag resistance of segment ij can be evaluated for $a = i, j$ and $b = i, j$ from equation (C.43) as:

$$\begin{aligned}
\boldsymbol{b}_{ij}^{ii} &= l_{ij} \boldsymbol{\mathcal{B}}(\theta_{ij}) \int_0^1 (1-s)(1-s) ds = \frac{l_{ij}}{3} \boldsymbol{\mathcal{B}}(\theta_{ij}) \\
\boldsymbol{b}_{ij}^{ij} &= l_{ij} \boldsymbol{\mathcal{B}}(\theta_{ij}) \int_0^1 (1-s) s ds = \frac{l_{ij}}{6} \boldsymbol{\mathcal{B}}(\theta_{ij}) \\
\boldsymbol{b}_{ij}^{ji} &= l_{ij} \boldsymbol{\mathcal{B}}(\theta_{ij}) \int_0^1 s(1-s) ds = \frac{l_{ij}}{6} \boldsymbol{\mathcal{B}}(\theta_{ij}) \\
\boldsymbol{b}_{ij}^{jj} &= l_{ij} \boldsymbol{\mathcal{B}}(\theta_{ij}) \int_0^1 s^2 ds = \frac{l_{ij}}{3} \boldsymbol{\mathcal{B}}(\theta_{ij})
\end{aligned} \tag{C.45}$$

Assuming a simple mobility law whereby the drag resistance exerted on a dislocation line continuously vary with its character, a simple interpolation from edge to screw properties can be used, such that:

$$\boldsymbol{\mathcal{B}}(\theta_{ij}) = \left(\mathcal{B}_{ij}^{e} \sin^2 \theta_{ij} + \mathcal{B}_{ij}^{s} \cos^2 \theta_{ij}\right) \boldsymbol{I}_2 \tag{C.46}$$

where $\mathcal{B}_{ij}^{e}$ and $\mathcal{B}_{ij}^{s}$ are the edge and screw mobilities of segment ij, and $\boldsymbol{I}_2$ is the second-order identity tensor. As mentioned in Section 2.5, the mobility coefficients account for the phonon interactions with moving dislocations. Their values depend on the material properties and on the temperature, and can be measured experimentally or estimated via atomistic simulations.

Using equations (C.42) to (C.45), the velocity $\vec{\boldsymbol{V}}_j$ of each node j can be determined by solving the system of equations given in (C.14). Separating the case $i = j$ from that of $i \neq j$, equation (C.14) can be decomposed as:

$$\vec{\boldsymbol{F}}_i = \boldsymbol{B}_{ii}\vec{\boldsymbol{V}}_i + \sum_{j \neq i} \boldsymbol{B}_{ij}\vec{\boldsymbol{V}}_j \tag{C.47}$$

From the definition of $\boldsymbol{B}_{ij}$ given in (C.42), expression (C.47) reads:

$$\vec{\boldsymbol{F}}_i = \sum_{j \neq i} \boldsymbol{b}_{ij}^{ii}\vec{\boldsymbol{V}}_i + \sum_{j \neq i} \boldsymbol{b}_{ij}^{ij}\vec{\boldsymbol{V}}_j \tag{C.48}$$

where sub-matrices $\boldsymbol{b}_{ij}^{ii}$ and $\boldsymbol{b}_{ij}^{ii}$ are obtained from (C.45) as:

$$\begin{aligned} \boldsymbol{b}_{ij}^{ii} &= \frac{l_{ij}}{3}\boldsymbol{\mathcal{B}}(\theta_{ij}) \quad \text{for } a = b = i \\ \boldsymbol{b}_{ij}^{ij} &= \frac{l_{ij}}{6}\boldsymbol{\mathcal{B}}(\theta_{ij}) \quad \text{for } a \neq b \end{aligned} \tag{C.49}$$

Finally, inserting (C.49) into (C.48) yields the following system of equations:

$$\vec{\boldsymbol{F}}_i = \sum_{j \neq i} \frac{l_{ij}}{6}\boldsymbol{\mathcal{B}}(\theta_{ij})(2\vec{\boldsymbol{V}}_i + \vec{\boldsymbol{V}}_j) \tag{C.50}$$

System of equations (C.50) describe the motion of groups of connected dislocation nodes. Thus, for a group of N connected nodes, system (C.50) is of size $3N$. However, because all nodes are not individually connected to all other nodes, such system is generally extremely sparse. For example, in the case of a dislocation group comprising no junction node (i.e. for a dislocation line where each node, expect for the end nodes,

is connected to two nodes), system (C.50) results in a banded matrix of width 9. For such reason, special algorithms dedicated to sparse systems have been implemented to efficiently compute nodal velocities. Furthermore, a local velocity approximation is presented in Section 2.9.2.

C.3.2 Glide velocity

As presented in Section 2.2.4, dislocation lines are physically constrained to glide on their slip plane(s). As a result, each nodal velocity $\vec{\boldsymbol{V}}_i$ computed in equation (C.50) must be projected on the glide plane(s) on which the dislocation node lies, i.e. on the glide plane(s) associated with each connection of node i. Therefore, for each node i connected to nodes j, the nodal velocity $\vec{\boldsymbol{V}}_i$ must satisfy:

$$\vec{\boldsymbol{V}}_i \cdot \vec{\boldsymbol{n}}_{ij}^k = 0 \quad ; \quad \begin{array}{l} \forall \text{ node } j \text{ connected to node } i \\ \forall \text{ plane } k \text{ associated with segment } ij \end{array} \tag{C.51}$$

where $\vec{\boldsymbol{n}}_{ij}^k$ designates the unit normal associated with the k-th glide plane of segment ij. Note that the glide plane of any dislocation segment that is not a sessile junction is defined by its Burgers vector and its line direction.

C.3.3 Junction case

The motion of nodes belonging to junction segments requires a specific treatment. As mentioned in Appendix C.3.2, the velocity of a dislocation node needs to be projected into the glide plane(s) of all segments connected to the node. Thus, in the case of a sessile junction node, lying at the intersection of two planes, the direction of the velocity becomes collinear to the line formed by the intersection of both planes, such that the node can only move along the direction of the junction. Therefore, the resulting motion is artificial in the sense that the movement is purely tangential to the line direction, leading to either an increase or a decrease in the junction length. Physically, the growth of a junction occurs via the zipping process, which is solely

governed by segment-segment collisions and whose implementation is detailed in Section 2.6. Therefore, to remain consistent with the collision procedure and adopt a numerical framework closely based on the physics of dislocations, only the shrinking of junction segments is allowed by motion of dislocation nodes. This can be enforced by ensuring that the velocity of each junction node i is pointing in the same direction as junction segment ij oriented from node i to node j, i.e. each node i connected to node j for which segment ij is a junction segment must verify:

$$\vec{\boldsymbol{V}}_i \cdot \vec{\boldsymbol{t}}_{ij} = 0 \quad ; \quad \forall \text{ junction segment } ij \tag{C.52}$$

where $\vec{\boldsymbol{t}}_{ij}$ denotes the line direction (or unit tangent) of junction segment ij, i.e. the line defined by the intersection of the slip planes of the two dislocations that formed the junction. Note here that the use of equation (C.52) applies to simple sessile junctions only, i.e. to junctions that result from the intersection between two dislocations belonging to two different slip systems, and for which the Burgers vector is not comprised in one of the existing slip planes of the crystal. If the junction is glissile, i.e. if its Burgers vector lies in one of the slip planes of the crystal, the dislocation node is treated as a regular dislocation node and projected on its unique slip plane defined by its Burgers vector and its line direction. In the case of more complex junction segments resulting from the intersection of more than two dislocation segments, two cases may appear: (1) the motion of the junction node is null if the junction is sessile, since the intersection of three planes reduces to a single point, or (2) the nodes are treated as glissile nodes in the case of glissile triple junctions as evidenced in [31].

C.4 Time integration

System of equations (C.14) allows for the determination of nodal velocities $\vec{\boldsymbol{V}}_i$ for each dislocation node i. The conversion of the velocities into nodal displacements

defining dislocation motion therefore requires time integration. The choice of an appropriate time integrator is crucial in DDD, especially as it controls the time step size versus the numerical stability [5, 220]. Thus, the choice of a stable time integrator may allow for the use of large time steps improving the computational efficiency of the simulations. On the other hand, the choice of a poor time integrator may impose the use of small time steps to avoid numerical instabilities in dislocations motion. The different types of time integrators are broken down into two classes, explicit and implicit. While explicit integrators are usually very simple to implement as they only require the evaluation of forces and positions from the previous time step, their stability is known for being very sensitive to the size of the time step. On the other hand, implicit integrators allows for larger time steps thanks to their more robust stability, but the use of an implicit formulation to determine dislocation motion requires the knowledge of positions and forces at the previous and next time step. Therefore, when implicit integrators are used, the determination of nodal velocities must be achieved by iterative implicit schemes, such as the Newton-Raphson method, thereby requiring significant additional computational work. Particularly, each iteration within an implicit scheme involves the evaluation of the nodal forces, which is known to be the most intensive computational process in DDD simulations (see Appendix C.2). Moreover, the use of very large time steps in DDD simulations is not always desired, and depends on the dislocation configuration. This is because when time steps are too large, interactions and collisions between dislocation segments may be missed, leading to non-physical behaviors. This last point is particularly important since dislocation-dislocation interactions are known to play a crucial role in strain hardening [227, 52, 136, 93] (see Chapter 4). As a result, the vast majority of current DDD codes are using explicit time integrators to date [97, 68, 254, 5, 69]. For this reason, explicit methods have been implemented.

The simplest explicit method consists in a backward Euler integration, where the

new position $\vec{\boldsymbol{x}}_i^{\,t+\Delta t}$ of each dislocation node i at time $t+\Delta t$ is updated from the position and the velocity at previous step as:

$$\vec{\boldsymbol{x}}_i^{\,t+\Delta t} = \vec{\boldsymbol{x}}_i^{\,t} + \vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right)\Delta t \tag{C.53}$$

where $\vec{\boldsymbol{V}}_i = \vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right)$ is the velocity at node i given as a function of the nodal positions $\vec{\boldsymbol{x}}_i^{\,t}$ at time t and Δt is the current time step increment. Although this method is the simplest to implement, it is also the less stable one. Thus, when the time step is large and the dislocation discretization is fine, i.e. when the nodal displacement $\vec{\boldsymbol{V}}_i\Delta t$ approaches the size of the segments to which it is connected, vibrations may appear, leading to non-physical multiplication of dislocations, that can further lead to the collapse of the simulation. To address this issue, enhanced explicit integration schemes have been proposed. Among them, the trapezoidal Euler integration relies on a predictor-corrector scheme in which the nodal position $\vec{\boldsymbol{x}}_i^{\,t+\Delta t}$ at each node i and time step $t+\Delta t$ is iteratively determined as:

$$\vec{\boldsymbol{x}}_i^{\,t+\Delta t}\big|_0 = \vec{\boldsymbol{x}}_i^{\,t} + \vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right)\Delta t \tag{C.54}$$

$$\vec{\boldsymbol{x}}_i^{\,t+\Delta t}\big|_{j+1} = \vec{\boldsymbol{x}}_i^{\,t} + \frac{\vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t+\Delta t}\big|_j\right) + \vec{\boldsymbol{V}}_i\left(\vec{\boldsymbol{x}}_i^{\,t}\right)}{2}\Delta t \tag{C.55}$$

where $\vec{\boldsymbol{x}}_i^{\,t+\Delta t}\big|_j$ denotes the j-th iterate of the corrector, and the iterative process is stopped when all nodes converge towards a new position, i.e. when the error between two subsequent iterations becomes smaller than a prescribed tolerance. Although this method yields a better numerical stability, and therefore theoretically allows for larger time steps or finer segment mesh sizes, the total cost of the evaluations of nodal forces required in the calculations of the correctors $\vec{\boldsymbol{x}}_i^{\,t+\Delta t}\big|_j$ at each iterate j needs to be balanced against the gain in time step increment it permits.

Note also that very recently, the development of a local time-step subcycling approach has demonstrated potential significant improvement of the global efficiency

for explicit and implicit integrators [220], but has not been implemented in this work.

APPENDIX D

NUMERICAL IMPLEMENTATION OF THE DDD SIMULATION TOOL

In this appendix, important details on the numerical implementation of the DDD simulation tool developed in the course of this thesis and presented in Chapter 2 are provided.

D.1 Topological changes of the dislocation network

During a DDD simulation, dislocation segments multiply and interact with one another as a result of their motion. Therefore, an adaptive meshing of dislocation segments must be implemented in order to (1) keep the dislocation network well discretized, such as to allow for a good description of dislocation line curvature and avoid numerical vibrations, and (2) permit the treatment of dislocation core reactions whose incorporation is fundamental to reproduce strain hardening behaviors. For these purposes, specific topological operations are implemented to serve as supporting functions of dislocation line operations inducing changes in the network topology.

D.1.1 Topological operations

In the present nodal code, a set of topological operations allowing to perform topological changes in the dislocation network is implemented. A topological change here refers to a modification of the dislocation network that induces a change in the segments connectivity. Such changes principally occur during dislocation intersections (e.g. junctions formation and annihilations) and allows for dynamical line remeshing (see Section 2.7). The two main topological operations implemented in the DDD code, namely the split of a segment and the merge of nodes, are presented below.

D.1.1.1 Split of dislocation segment

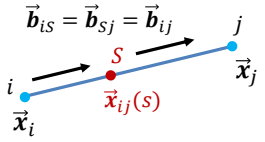


Figure D.1: Schematic of the split of a dislocation segment ij defined between end nodes at positions $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ and with Burgers vector $\vec{\boldsymbol{b}}_{ij}$. The original segment is split by inserting a new node S at coordinate $\vec{\boldsymbol{x}}_{ij}(s)$. In doing so, the properties of original segment ij (Burgers vector, Miller indices, ...) are transferred to new segments iS and Sj.

As depicted in figure D.1, the procedure to split a dislocation segment consists in inserting a new node at a given abscissa along the segment, in between the existing end nodes defining the segment. Through this procedure, the connectivity between the nodes is updated, and the properties of the original segment are transferred to the new segments, such that (1) the Burgers convention introduced in Section 2.3.1 is respected, and (2) the dislocation slip plane remains definite for each segment.

D.1.1.2 Merge of dislocation nodes

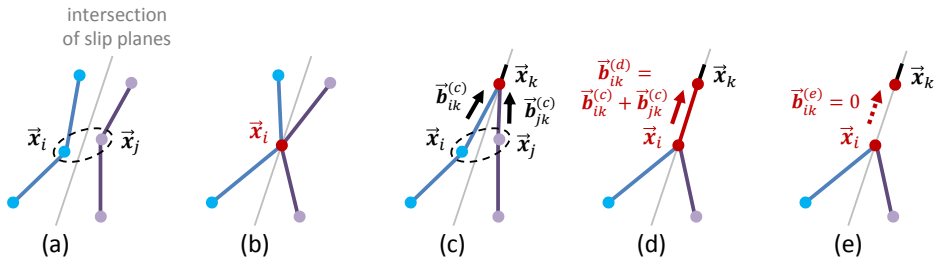


Figure D.2: Schematic of the merge of two dislocation nodes i and j. From stages (a) to (b), nodes i and j are merged into new node i who lies into the intersection of the slip planes of the initial nodes. In case (c), nodes i and j are connected to a common node k, such that the outcome of the merging procedure leads either (d) to the formation of a junction or (e) to an annihilation.

As illustrated in figure D.2, the procedure for merging two dislocation nodes consists in merging nodes i and j into the single node i. In this procedure, the new position of node i is determined by the union of the geometrical constraints applying to both nodes i and j. Thus, if nodes i and j belong to two different planes, the merged node will necessarily be located along the intersection of both planes, such as to conserve the properties of initial intersecting segments. However, the merge can also be performed between two coplanar nodes or two nodes belonging to the same dislocation line. When the merge has been performed, all connections of node j are transferred to node i, and node j is finally deleted.

Note however that a special case occurs when nodes i and j are connected to a common node k, as depicted in figure D.2(c). In that case, a new segment ik will be formed, whose resulting Burgers vector is equal to the sum of the Burgers vectors of the two initial segments ik and jk. If the resulting Burgers vector is null, it corresponds to an annihilation and new segment ik is deleted by removing the connection between nodes i and k (see figure D.2(e)).

D.1.2 Numerical implementation of junction formation

In this section, the numerical implementation of junction formation and annihilations presented in Section 2.6.1 is detailed. The numerical implementation of the dislocation segment-segment interactions requires the development of different steps, which are detailed in the following. First, a list of dislocation neighbor segments is created for each dislocation segment. Then, the minimum distance between a dislocation segment and each of its neighbors is computed, such that pairs of segments whose minimum distance d_{min} satisfies $d_{min} \leq d_{crit}$ are considered as intersecting. Using the topological procedures presented in Appendix D.1.1 enabling the split of dislocation segments and the merge of dislocation nodes, segment intersections are performed, and, finally, newly formed topology is checked for potential errors arising from the

above procedures. All the steps are described in detail below.

D.1.2.1 Determination of neighbor segments

Potential segment interactions only occur between close unconnected segments. For this reason, and for avoiding a time-consuming $\mathcal{O}(N_{seg}^2)$ distance calculation procedure involving all N_{seg} dislocation segments present in the simulation volume, a neighbor list is created to reduce the number of segment-segment distances to compute. To create the neighbor list, the simulation volume is partitioned into a certain number of boxes. The optimum box size is determined from the length of the longest segment of the simulation. With this partitioning, each dislocation segment lying in a box B will have as neighbors all segments lying in the 27 surrounding boxes (26 surrounding boxes and box B itself). Although d_{crit} is usually way smaller than the average segment length, the use of 27 boxes is required in order to properly handle the case where a segment lies closely to the boundary of the central box B.

D.1.2.2 Calculation of the minimum distance between two segments

Once the neighbor list has been created, the minimum distance between a segment and each of its neighbors is computed. Assuming straight segments, the shortest distance $d_{min}(ij, kl)$ between two segments ij and kl is found by minimizing the following expression:

$$d_{min}(ij,kl) = \min_{s,t} \|\vec{\boldsymbol{x}}_{ij}(s) - \vec{\boldsymbol{x}}_{kl}(t)\| \qquad \text{with} \qquad \begin{cases} 0 \leq s \leq 1 \\ 0 \leq t \leq 1 \\ i \neq j \neq k \neq l \end{cases} \tag{D.1}$$

From this expression, the curvilinear positions s and t corresponding to the positions for which the distance between the two segments reaches a minimum can also be determined. If the distance satisfies $d_{min}(ij, kl) \leq d_{crit}$, and the dislocation segments are moving towards each other, the intersection procedure is performed. Note here

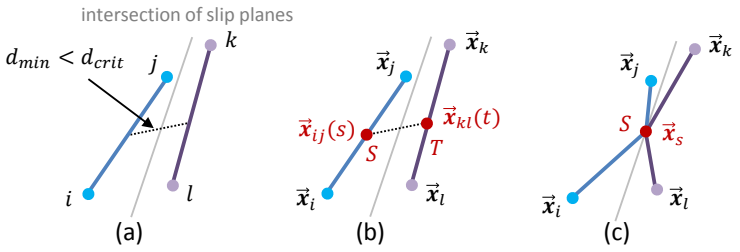


Figure D.3: Sequence of operations performed during the intersection procedure. (a) When two dislocation segments ij and kl are moving towards each other and are approaching at a distance lower than d_{crit}, the intersection procedure is as follows: (b) first, both segments are split via insertion of nodes S and T. Then, (c) nodes S and T are merged into single node S lying at the intersection of the glide planes of the initial segments. As a result, the initial dislocation segments become connected through a 4-connected junction node S.

that for the sake of consistency, the value of d_{crit} must be chosen such that (1) d_{crit} is smaller than the average dislocation length, and (2) d_{crit} is not too large with respect to dislocation nodal displacements.

D.1.2.3 Intersection procedure

The sequence of operations performed during the intersection procedure depicted in figure D.3 is detailed here. When two dislocation segments meet the intersection criterion, new nodes S and T are inserted along segments ij and kl at curvilinear positions s and t, respectively. Such is done by using the split procedure. Nodes S and T are then merged into a new single node S using the merge nodes procedure (see Appendix D.1.1). In order to respect the physics of dislocations, S is constrained to lie in the intersection of the glide plane(s) of initial segments ij and kl. Further, in order to avoid numerical instabilities induced by the creation of short segments, when position $\vec{\boldsymbol{x}}_{ij}(s)$ along segment ij (that corresponds to the minimum distance to segment kl) is very close to node i (respectively to node j), segment ij is not split, and node i (respectively j) is used in the merging procedure. As shown in figure 2.13 and discussed in Section 2.6.1, with this process, junction formation and annihilation

result from successive intersections of connected segments.

D.1.2.4 Verification of the new topology

As presented in the above, the numerical treatment of intersections induces topological changes in the discretized dislocation network. After interactions have been performed between all intersecting segments, the new dislocation topology is verified to ensure that potential errors that may have occurred are fixed or circumvented, such that the simulation can continue running properly. The verifications mainly consist in checking the consistency of the segment slip systems and the reciprocity of the nodal connectivity.

D.1.3 Numerical implementation of node dissociation

In this section, the numerical implementation of the dissociation procedure presented in Section 2.6.2 is detailed. From a topological perspective, the dissociation of a junction node is similar to the passage from the configuration depicted in figure 2.13(b) to that in figure 2.13(c), where the formation of the junction segment results from the sequence of operations illustrated in figure D.3 performed on two arms of the junction node. The two arms on which the split and merge occur depend on the dissociation configuration. Thus, for a 4-connected node i, the dissociation procedure yields three different possible outcomes, each of whose corresponds to a different choice of arms on which the split and merge operations are performed. When node i connected to nodes 1 to 4, i.e. $i \rightarrow \{1, 2, 3, 4\}$, is dissociated into connected junction nodes j and k, the three final possible configurations are as follows: (1) $j \rightarrow \{1, 2\}$ and $k \rightarrow \{3, 4\}$, (2) $j \rightarrow \{1, 3\}$ and $k \rightarrow \{2, 4\}$, and (3) $j \rightarrow \{1, 4\}$ and $k \rightarrow \{2, 3\}$. For each case, the new configuration is created and the nodal forces and velocities on newly created junction nodes j and k are calculated, such that the power dissipation of the configuration can be evaluated. Then, the initial 4-connected node configuration is restored and the dissipation of the next possibility

is evaluated. After evaluations of the dissipation of the three configurations, the one yielding the maximum power dissipation is recreated and kept in the simulation. Also, to ensure that junction segment jk would not vanish after its creation, only configurations in which velocities of nodes j and k are in opposite directions are considered, i.e. configurations corresponding to a zipping process. Further note that the length of junction segment jk, that is created as a result of the dissociation procedure of initial node i, is chosen to be slightly greater than the critical intersection radius d_{crit} to ensure that nodes j and k are not merged through the intersection procedure immediately after their creation (see Section 2.6.1). With this process, the Burgers vector of junction segment jk generally differs for each configuration. Of special interest, the dissociation procedure is able to treat unpinning events by ways of selecting configurations in which the junction Burgers vector is null.

D.2 Parallelization of the DDD simulation tool

In this section, details on the numerical implementation of the critical stages of the DDD cycle are presented. For a global overview of the parallel computing and load balancing strategies adopted in this work, the reader is referred to Section 2.9.3.

D.2.1 Short-range interactions calculation

Practically, the implementation of the current strategy based on an homogeneous distribution of total number of neighbors across the CPUs includes several steps. It also to be noticed that such distribution tacitly relies on the assumption that the computational cost of interaction forces is identical for any pair of segments. When using the non-singular analytical formulation for isotropic elasticity given in [5] and presented in Section 2.5.3, two cases are distinguished, collinear and non-collinear segments, whose costs differ. However, the occurrence of collinear cases is statistically limited compared to the general non-collinear case, such that, on average, an equivalent cost between all interactions can be safely considered. With that in

mind, the parallel distribution can be performed as follows. First, the number of neighbors must be determined for each segment. Here, the definition of short-range neighbors must remain consistent with the use of the Box Method (see Section 2.9.2.1) used in the current code. Thus, to avoid either disregarding or double counting segment interactions, the neighbors of each segment lying in box α are taken as the ensemble of segments lying in the union U_α composed of the 26 boxes surrounding box α and box α itself. Therefore, the number of neighbors $N_{nei} = N_\alpha$ of each segment belonging to box α can be efficiently determined using a bucket sort algorithm on the 3D spatial coordinates of the segments. For the sake of efficiency, all segments are sorted by box at the beginning of each time step so as to define both short-range and long-range interactions simultaneously. Once the number of neighbors is determined for each segment, a distribution of segments ensuring an optimal load balance can be achieved using the inexpensive best fit decreasing algorithm. With this process, the segments are first sorted in decreasing order by their number of neighbors, and then sequentially assigned in order to the CPU with the largest remaining space (i.e. the lowest total load). Here, the decreasing sort is used to ensure that the segments with the largest number of neighbors are not to be distributed once the bins are already filled. This sort is achieved using a $\mathcal{O}(N_{seg} \log N_{seg})$ heapsort algorithm. Note that in general, a simple selection sort algorithm of complexity $\mathcal{O}(N_{seg}^2)$ will lead the overall scalability to dramatically drop as N_{seg} increases.

Once dislocation segments have been distributed on each CPU, short-range interactions can be computed locally and nodal velocities and displacements can be determined (see Section 2.5) thanks to the nodal velocities approximation (see Section 2.9.2.2). The new nodal positions are then updated on all CPUs via point-to-point communications. At this stage, most of the computational work of the DDD cycle has been performed, as the calculation of interaction forces usually accounts for more than 90% of the total cost per simulation step. However, as stated by Amdahl's

law, the maximum theoretical speed-up that can be achieved with parallel computing quickly saturates when all portions of a program are not parallelized, i.e. when sequential operations remain [3]. In other words, the scalability will fall all the more quickly than the fraction of sequential operations is large when the number of processors increases. Therefore, other important stages such as long-range calculations, the dissociation procedure and the treatment of dislocation intersections (see Section 2.6 and Appendix D.1), require a parallel implementation as well.

D.2.2 Long-range calculation

When using the Box Method (see Section 2.9.2.1), the most straightforward strategy consists in equally distributing the N_{box} boxes across the CPUs so as to locally compute the total stress arising from long-range dislocations at the center of each box. Naturally, this approach will provide a good scalability when dislocation segments are well distributed within the volume. However, as for short-range interactions, the optimal strategy consists in distributing the boxes according to the total number of long-range interactions associated with each of them.

D.2.3 Dissociation procedure

For the dissociation procedure, the 4-connected nodes are first distributed among the processors (ensuring that pairs of 4-connected nodes connected to each other remain on a single CPU so as to avoid treating topological conflicts between CPUs) and the determination of the optimal configuration is determined locally (see Section 2.6.2 and Appendix D.1.3). Note that this determination involves the computation of several interaction forces and is in general identical in terms of computational cost for every 4-connected node. Every new configuration is then transferred back to all other CPUs.

D.2.4 Dislocation intersections

In the case of dislocation intersection, the picture is slightly different. As detailed in Section 2.6.1, the treatment of dislocation collisions involve two main steps, namely the determination of potential pairs of intersecting segments and the application of the resulting topological changes. Clearly, the last step does not involve any calculation, but solely invokes topological operations to modify the dislocation data (see Appendix D.1.1). In this case, the reduction in the sequential fraction of operations of the code must be balanced with the cost of the communication induced by the parallelization, which is not accounted for in Amdahl's model [3]. Practically, following the numerical intersection procedure detailed in Appendix D.1.2, it appears that only the parallelization of the determination of neighbor segments and of the calculation of the minimum distance between pairs of neighbor segments will contribute to an increase in the scalability of the code. Here, the determination of neighbor segments can be achieved using the same strategy as that described earlier to distinguish between short and long-range interactions. However, the metric used to determine the size of the boxes is not the same in both cases. In the first case, the choice is dictated by the validity of the long-range approximation based on the $1/R$ physical decay of the elastic stress fields. In the case of intersections, the size of the box is typically chosen to be smaller since neighbor segments in the context of intersections only pertain to pairs of segments whose distance is potentially smaller than the critical radius $d_{crit} \leq l_{avg}$, where l_{avg} is the average length of dislocation segments. Using a bucket sort algorithm, dislocation neighbors can be determined and distributed across the CPUs. Then, the computation of the minimum distance between pairs of neighbors can be performed locally. Further, potential collisions are transferred to all CPUs and the relatively inexpensive topological changes are finally performed sequentially.

APPENDIX E

FINITE ELEMENT FORMULATION FOR STRESS-STRAIN CALCULATION

E.1 Principle of virtual work

In continuum mechanics, the mechanical state at each point of a body V subjected to displacements and traction boundaries conditions can be determined by considering the following fundamental boundary value problem:

$$\sigma_{ij} = C_{ijkl}\epsilon^e_{kl} \tag{E.1}$$

$$\sigma_{ij,j} = 0 \tag{E.2}$$

$$u_i = u^*_i \text{ on } S_u \tag{E.3}$$

$$\sigma_{ij}n_j = t^*_i \text{ on } S_t \tag{E.4}$$

where σ_{ij} denotes the components of the stress tensor, ϵ^e_{kl} is the elastic strain, and C_{ijkl} is the fourth-order elastic stiffness tensor. In equations (E.3) and (E.4), $\vec{\boldsymbol{u}}^*$ and $\vec{\boldsymbol{t}}^*$ denote the applied displacements and tractions boundary conditions imposed on volume external surfaces S_u and S_t with normal $\vec{\boldsymbol{n}}$, respectively. Under the small strain framework, the total strain $\boldsymbol{\epsilon}$ is expressed as the symmetric part of the gradient of the displacement:

$$\epsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \tag{E.5}$$

Since the mechanical equilibrium $\sigma_{ij,j} = 0$ (here the body forces are neglected) must hold at every material point in volume V, the following relation must hold for the entire medium, for any differentiable suitable trial function $\vec{\boldsymbol{\psi}}$ that satisfies the boundary

conditions:

$$\int_V \sigma_{ij,j}\psi_i dV = 0 \tag{E.6}$$

Performing an integration by parts on (E.6) and using the divergence theorem, one can derive the weak formulation associated with the boundary value problem (E.1)–(E.4):

$$\begin{aligned}\int_V \sigma_{ij,j}\psi_i dV &= \int_V \left(\sigma_{ij}\psi_i\right)_{,j} dV - \int_V \sigma_{ij}\psi_{i,j} dV = 0 \\ &= \int_S \sigma_{ij}\psi_i n_j dS - \int_V \sigma_{ij}\psi_{i,j} dV = 0\end{aligned} \tag{E.7}$$

where $S = \partial V$ denotes the outer surface enclosing volume V, with normal $\vec{\boldsymbol{n}}$. Since relation (E.7) must hold for any suitable vector field $\vec{\boldsymbol{\psi}}$, one can denote $\delta\vec{\boldsymbol{u}}$ any admissible virtual displacement field in the body and use $\vec{\boldsymbol{\psi}} = \delta\vec{\boldsymbol{u}}$ as a trial function. With that, and using the definition of the traction vector $t_i = \sigma_{ij}n_j$, equation (E.7) can be written as:

$$\int_V \sigma_{ij}\delta u_{i,j} dV - \int_S t_i \delta u_i dS = 0 \tag{E.8}$$

Equation (E.8) is generally known as the Principle of Virtual Work (PVW). Essentially, the PVW states that the stresses and tractions are in equilibrium in the medium if the internal virtual work (left side of (E.8)) equates the external work (right side of (E.8)) done when a virtual displacement field is applied. The weak form (E.8) of the fundamental boundary value problem (E.1)–(E.4) is the starting point of the Finite Element Method (FEM) formulation of the classical approach in continuum mechanics. Depending on the ingredients of the constitutive law, it can be used to solve for linear elastic problems, or be extended to inelastic formulations, such as to account for elasto-plastic or elasto-viscoplastic behaviors.

E.2 Linear elasticity

This section is dedicated to the derivation of the mechanical FEM formulation in the case of linear elastic behaviors. This formulation is for instance used in DDD simulations to solve for the correction problem when using the superposition approach (see Section 2.4.2). In a linear elastic medium, the constitutive law simply writes:

$$\sigma_{ij} = C_{ijkl}\epsilon^e_{kl} = C_{ijkl}\epsilon_{kl} = C_{ijkl}u_{k,l} \tag{E.9}$$

where the elastic strain $\boldsymbol{\epsilon}^e$ is equal to the total strain $\boldsymbol{\epsilon}$ such that the strain definition in (E.5) can be directly used. Replacing the stress in the weak form (E.8) by its definition in (E.9) for linear elasticity yields:

$$\int_V C_{ijkl}u_{k,l}\delta u_{i,j}dV - \int_S t_i\delta u_i dS = 0 \tag{E.10}$$

Expression (E.10) describes a linear partial differential equation on displacements $\vec{\boldsymbol{u}}$, that can be numerically solved for using the FEM technique. Introducing a spatial discretization whereby medium V is partitioned using three-dimensional elements, the i-th component of the displacement field u_i at position $\vec{\boldsymbol{x}}$ anywhere in the volume can be interpolated from the displacement values u_i^a at mesh nodes as:

$$u_i(\vec{\boldsymbol{x}}) = \sum_{a=1}^{N_d} N^a(\vec{\boldsymbol{x}})u_i^a \tag{E.11}$$

where N^a is the shape function at node a and N_d is the number of nodes per element. More details on the type of elements used in the DDD code and their properties are given in Appendix E.5. Following the FEM framework, the spatial derivative of the displacement can be expressed as a function of the spatial derivative of the shape functions:

$$u_{i,j}(\vec{\boldsymbol{x}}) = \sum_{a=1}^{N_d} N^a_{,j}(\vec{\boldsymbol{x}}) u^a_i \tag{E.12}$$

where $N^a_{,j}$ designates the spatial derivative of the shape function N^a with respect to the j-th component. By plugging the discretization introduced in equations (E.11) and (E.12) into equation (E.10), one obtains:

$$\sum_{a=1}^{N_d} \left[\sum_{b=1}^{N_d} \int_V C_{ijkl} N^b_{,l} u^b_k N^a_{,j} dV - \int_S t_i N^a dS \right] \delta u^a_i = 0 \tag{E.13}$$

and since system of equations (E.13) must hold for any virtual displacement $\delta\vec{\boldsymbol{u}}$ satisfying the boundary conditions (E.3) and (E.4), it can be written as:

$$\sum_{a=1}^{N_d} \left[\sum_{b=1}^{N_d} \left(K^{ab}_{ik} u^b_k \right) - F^a_i \right] = 0 \tag{E.14}$$

where the nodal quantities K^{ab}_{ik} and F^a_i are defined as:

$$K^{ab}_{ik} = \int_V C_{ijkl} N^b_{,l} N^a_{,j} dV \tag{E.15}$$

$$F^a_i = \int_S t_i N^a dS \tag{E.16}$$

System (E.14) is assembled using the conventional FEM assembly procedure. With equation (E.14), displacements $\vec{\boldsymbol{u}}^{\,a}$ at each FE node a can be can solved for, leading to the determination of the strain and stress fields using constitutive equations (E.5) and (E.9). Evaluation of the integrals (E.15) and (E.16) at mesh nodes is usually done by performing a Gaussian integration over the elements/surfaces. In matrix form, system of equations (E.14) can be simply written:

$$\boldsymbol{K}\,[\boldsymbol{u}] = \boldsymbol{F} \tag{E.17}$$

where $\boldsymbol{K}$ is a square matrix of size $3N_t \times 3N_t$ where N_t is the total number of nodes

in the FEM mesh, and $\boldsymbol{F}$ and $[\boldsymbol{u}]$ are column vectors of length $3N_t$. Here vector $[\boldsymbol{u}]$ is populated with values u_i^a associated with the components of the nodal displacements, for $i = \{1, ..., 3\}$ and $a = \{1, ..., N_t\}$. Therefore, solving linear system (E.17) for $[\boldsymbol{u}]$ leads to the determination of all nodal displacements $\vec{\boldsymbol{u}}^{\,a}$. From there, the resulting strain and stress can be calculated at the mesh nodes using constitutive relations (E.5) and (E.9). Numerically, system (E.17) can be solved for using *direct methods* such as the LU decomposition or the QR factorization. However, although such methods are robust and allow to obtain the solution in a single pass, the computational cost is usually prohibitive. As a matter of fact, their computational complexity scales with $\mathcal{O}(N_t^3)$, which becomes impractical for large systems. As a result, iterative techniques such as the Jacobi method or the Gauss-Seidel method are usually preferred as their complexity drops to $\mathcal{O}(N_t^2)$. Further, the density of matrix $\boldsymbol{K}$ directly results from the mesh connectivity between elements. Consequently, system (E.17) is usually very sparse and dedicated sparse system solvers are generally used. Although from a general standpoint it is hard to evaluate the computational complexity of sparse solver (it generally depends on the sparsity of the system), the lowest complexity that can usually be achieved with standards FEM approaches is $\mathcal{O}(N_t^2)$.

Note that it is sometimes convenient to write the whole formulation using a matrix notation. In order to benefit from the symmetry of stress and strain tensors, the Voigt notation is generally used. With this, the elastic constitutive law (E.9) can be written for general anisotropy as:

$$\boldsymbol{\sigma} = \boldsymbol{C} : \boldsymbol{\epsilon}$$

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{12} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{13} & C_{23} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{14} & C_{24} & C_{34} & C_{44} & C_{45} & C_{46} \\ C_{15} & C_{25} & C_{35} & C_{45} & C_{55} & C_{56} \\ C_{16} & C_{26} & C_{36} & C_{46} & C_{56} & C_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{12} \\ 2\epsilon_{13} \\ 2\epsilon_{23} \end{Bmatrix} \tag{E.18}$$

where coefficients C_{IJ} with indices $I, J = \{1, ..., 6\}$ correspond to the Voigt counterparts of stiffness coefficients C_{ijkl} with indices $i, j, k, l = \{1, 2, 3\}$, and where the correspondence $ij \leftrightarrow I$ and $kl \leftrightarrow J$ defines the convention that is adopted. With this, the strain definition (E.5) writes:

$$\boldsymbol{\epsilon} = \frac{1}{2}\left[\nabla \vec{\boldsymbol{u}} + (\nabla \vec{\boldsymbol{u}})^T\right] = \boldsymbol{D}\vec{\boldsymbol{u}} \tag{E.19}$$

whose terms and their dimensions are explicitly given by:

$$\vec{\boldsymbol{u}} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} (3 \times 1); \quad \boldsymbol{\epsilon} = \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{12} \\ 2\epsilon_{13} \\ 2\epsilon_{23} \end{Bmatrix} (6 \times 1); \quad \boldsymbol{D} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} & 0 \\ \frac{\partial}{\partial x_3} & 0 & \frac{\partial}{\partial x_1} \\ 0 & \frac{\partial}{\partial x_3} & \frac{\partial}{\partial x_2} \end{bmatrix} (6 \times 3) \tag{E.20}$$

where $\vec{\boldsymbol{u}}$ is the displacement vector field, and $\boldsymbol{D}$ is the symmetrized gradient operator. When the medium is discretized into N_d-noded elements, the value of vector field $\vec{\boldsymbol{u}}$ at every material point can be interpolated from the mesh nodal displacement values as:

$$\vec{\boldsymbol{u}} = \boldsymbol{N}\left[\vec{\boldsymbol{u}}\right] \tag{E.21}$$

where $\boldsymbol{N}$ is the matrix of shape functions and $[\vec{\boldsymbol{u}}]$ denotes the nodal displacements vector. Essentially, for each element defined with N_d nodes, $[\vec{\boldsymbol{u}}]$ can be expressed as:

$$[\vec{\boldsymbol{u}}] = \left\{u_1^1\ u_2^1\ u_3^1\ u_1^2\ u_2^2\ u_3^2\ ...\ u_1^{N_d}\ u_2^{N_d}\ u_3^{N_d}\right\}^T (3N_d \times 1) \tag{E.22}$$

where u_i^a denotes the i-th component of nodal displacement vector $\vec{\boldsymbol{u}}^{\,a}$. The interpolation function matrix $\boldsymbol{N}$ in (E.21) is given by:

$$\boldsymbol{N} = \begin{bmatrix} N^1 & 0 & 0 & N^2 & 0 & 0 & ... & N^{N_d} & 0 & 0 \\ 0 & N^1 & 0 & 0 & N^2 & 0 & ... & 0 & N^{N_d} & 0 \\ 0 & 0 & N^1 & 0 & 0 & N^2 & ... & 0 & 0 & N^{N_d} \end{bmatrix} (3 \times 3N_d) \tag{E.23}$$

where the nodal interpolation functions N^a depend on the types of element and interpolation, and are provided in Appendix E.5. With definitions (E.22) and (E.23), it can be noticed that equation (E.21) is the matrix form of (E.11). Combining equations (E.19) and (E.21), the strain can be expressed as a function of the nodal displacements as:

$$\boldsymbol{\epsilon} = \boldsymbol{D}\vec{\boldsymbol{u}} = \boldsymbol{DN}\left[\vec{\boldsymbol{u}}\right] \equiv \boldsymbol{B}\left[\vec{\boldsymbol{u}}\right] \tag{E.24}$$

where $\boldsymbol{B} = \boldsymbol{DN}$ is the matrix whose components are literally given from relations (E.20) and (E.23) as:

$$\boldsymbol{B} = \boldsymbol{DN} = \begin{bmatrix} N^1_{,1} & 0 & 0 & N^2_{,1} & 0 & 0 & ... & N^{N_d}_{,1} & 0 & 0 \\ 0 & N^1_{,2} & 0 & 0 & N^2_{,2} & 0 & ... & 0 & N^{N_d}_{,2} & 0 \\ 0 & 0 & N^1_{,3} & 0 & 0 & N^2_{,3} & ... & 0 & 0 & N^{N_d}_{,3} \\ N^1_{,2} & N^1_{,1} & 0 & N^2_{,2} & N^2_{,1} & 0 & ... & N^{N_d}_{,2} & N^{N_d}_{,1} & 0 \\ N^1_{,3} & 0 & N^1_{,1} & N^2_{,3} & 0 & N^2_{,1} & ... & N^{N_d}_{,3} & 0 & N^{N_d}_{,1} \\ 0 & N^1_{,3} & N^1_{,2} & 0 & N^2_{,3} & N^2_{,2} & ... & 0 & N^{N_d}_{,3} & N^{N_d}_{,2} \end{bmatrix} (6 \times 3N_d) \tag{E.25}$$

where $N^a_{,j} = \partial N^a / \partial x_j$ denotes the derivative of N^a with respect to the j-th spatial coordinate. Therefore, the PVW expressed in (E.8) reads in matrix notation:

$$\left(\int_V \boldsymbol{B}^T \boldsymbol{\sigma} dV - \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS \right) \left[\delta \vec{\boldsymbol{u}}\right]^T = 0 \tag{E.26}$$

where $\boldsymbol{B}^T$ and $\boldsymbol{N}^T$ denotes the transpose of the $\boldsymbol{B}$ matrix and the shape functions matrix, respectively, and $\vec{\boldsymbol{t}}$ is the traction vector. Expression (E.26) is obtained by noticing that $\sigma_{ij}\delta u_{i,j} = \sigma_{ij}\delta\epsilon_{ij}$ thanks to the symmetry between indices i and j. Further, since it must hold for any admissible displacement field $\delta\vec{\boldsymbol{u}}$, the PVW

rewrites:

$$\int_V \boldsymbol{B}^T \boldsymbol{\sigma} dV - \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS = 0 \tag{E.27}$$

Finally, substituting $\boldsymbol{\sigma}$ by the constitutive expression in (E.2) and using the nodal stain expression in (E.24), one obtains:

$$\int_V \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{B} \left[\vec{\boldsymbol{u}}\right] dV - \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS = 0 \quad \Leftrightarrow \quad \boldsymbol{K}_{el} \left[\vec{\boldsymbol{u}}\right] = \boldsymbol{F}_{el} \tag{E.28}$$

where the matrix quantities associated with each element are given by:

$$\boldsymbol{K}_{el} = \int_V \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{B} dV \quad (3N_d \times 3N_d) \tag{E.29}$$

$$\boldsymbol{F}_{el} = \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS \quad (3N_d \times 1) \tag{E.30}$$

The global system in equations (E.14) and (E.17) is finally constructed by assembly of element quantities $\boldsymbol{K}_{el}$ and $\boldsymbol{F}_{el}$ following the classical FEM assembly procedure.

The FEM approach using linear elasticity is used in DDD simulations to account for specific boundary conditions, especially in the case of the presence of free surfaces (see Section 2.4). As depicted in figure 2.12, DDD simulations performed on confined volumes require the determination of the mechanical state associated with a correction problem. In this correction problem, the boundary conditions account for both the initial boundary conditions imposed on the original volume, and the virtual boundary conditions arising from the presence of dislocations within an infinite medium. With that, correction fields $\boldsymbol{\sigma}^{ext}$ and $\vec{\boldsymbol{u}}^{\,ext}$ are directly determined by solving the linear system of equation (E.17).

E.3 Linearized elasto-plastic framework

The use of linear elastic behavior laws in continuum mechanics remains a special case whose validity is limited to specific problems. In general, the finite element

method is used to solve for more sophisticated behaviors such as elasto-plastic or elasto-visco-plastic responses.

In the case of an elasto-plastic behavior, the constitutive law can be written as:

$$\begin{aligned}\sigma_{ij} &= C_{ijkl}\left(\epsilon_{kl} - \epsilon^p_{kl}\right) \\ &= \frac{1}{2}C_{ijkl}\left(u_{k,l} + u_{l,k}\right) - C_{ijkl}\epsilon^p_{kl} \\ &= C_{ijkl}u_{k,l} - C_{ijkl}\epsilon^p_{kl}\end{aligned} \tag{E.31}$$

where ϵ_{kl} and ϵ^p_{kl} are the total and plastic strain tensors, respectively, and where the small strain relation (E.5) is used. Physically, the stress is generally the driving force of the plastic activity, such that equation (E.31) describes a non-linear relation. In practice, such behavior may be linearized for the sake of simplicity. Therefore, in this section, the formulation for linear elasto-plastic behaviors is introduced, while the case of more complex non-linear behaviors will be detailed in Appendix E.4.

The simplest approach to linearize the elasto-plastic behavior in (E.31) is to consider that the plastic strain is a constant input of the FEM procedure. Under this assumption, the PVW in (E.8) expresses as:

$$\int_V C_{ijkl}u_{k,l}\delta u_{i,j}dV - \int_V C_{ijkl}\epsilon^p_{kl}\delta u_{i,j}dV - \int_S t_i\delta u_i dS = 0 \tag{E.32}$$

such that the weak form using the FEM discretization introduced in expressions (E.11) and (E.12) writes:

$$\sum_{a=1}^{N_d}\left[\sum_{b=1}^{N_d}\int_V C_{ijkl}N^b_{,l}u^b_k N^a_{,j}dV - \int_V C_{ijkl}\epsilon^p_{kl}N^a_{,j}dV - \int_S t_i N^a dS\right]\delta u^a_i = 0 \tag{E.33}$$

Following the same procedure as that described in Appendix E.2, the system of equation (E.33) to solve for the displacement field can be written as:

$$\sum_{a=1}^{N_d}\left[\sum_{b=1}^{N_d}\left(K_{ik}^{ab}u_k^b\right)-F_i^a\right]=0 \qquad \text{(E.14 repeated)}$$

where the nodal quantities K_{ik}^{ab} and F_i^a associated with the linearized elasto-plastic behavior are given by:

$$K_{ik}^{ab}=\int_V C_{ijkl}N_{,l}^b N_{,j}^a dV \tag{E.34}$$

$$F_i^a=\int_V C_{ijkl}\epsilon_{kl}^p N_{,j}^a dV+\int_S t_i N^a dS \tag{E.35}$$

Essentially, the resistivity matrix K_{ik}^{ab} is similar to that obtained in the linear elastic case, while the force vector F_i^a requires the computation of an additional volume integral related to the plastic strain field ϵ_{kl}^p. Therefore, when the plastic strain field is assumed to be a constant input at each time step of the FEM procedure, the latter can be directly accounted for in the formulation as a supplementary body force.

E.4 Non-linear formulation

Non-linear behaviors arise when the stress tensor $\boldsymbol{\sigma}$ can no longer be expressed as a linear function of the displacement $\vec{\boldsymbol{u}}$. Common non-linear examples include the cases of elasto-plastic and elasto-viscoplastic behaviors in which the plastic strain can no longer be considered as a constant input of the model. In visco-plastic behaviors for instance, the constitutive law is often written as:

$$\dot{\boldsymbol{\sigma}}=\boldsymbol{C}:(\dot{\boldsymbol{\epsilon}}-\dot{\boldsymbol{\epsilon}}^p(\boldsymbol{\sigma})) \tag{E.36}$$

where the plastic strain rate $\dot{\boldsymbol{\epsilon}}^p$ is a function of the stress state $\boldsymbol{\sigma}$. Such behavior is for instance found in the classical flow rule, in which the plastic strain rate is expressed as:

$$\dot{\boldsymbol{\epsilon}}^p=\overline{\lambda}\boldsymbol{S} \tag{E.37}$$

where $\overline{\lambda}$ is the plastic multiplier and $\boldsymbol{S}$ denotes the deviatoric stress tensor, and is also very common in crystal plasticity laws in which a power law is used to model the plastic slip such that the plastic strain rate is expressed as:

$$\dot{\boldsymbol{\epsilon}}^p = \sum_s \boldsymbol{M}^s \dot{\gamma}^s \equiv \sum_s \boldsymbol{M}^s \dot{\gamma}_0^s \left| \frac{\boldsymbol{M}^s : \boldsymbol{\sigma}}{\tau_c^s} \right|^n \tag{E.38}$$

where $\boldsymbol{M}^s$ is the symmetric Schmid tensor, τ_c^s denotes the CRSS, $\dot{\gamma}_0^s$ the reference plastic shear and n an exponent related to the rate sensitivity.

Thus, when considering non-linear behaviors, the stress $\boldsymbol{\sigma}(\vec{\boldsymbol{u}})$ becomes a non-linear function of the displacement $\vec{\boldsymbol{u}}$. As a result, the PVW in equation (E.8) is expressed in matrix form (see equation (E.27)) as:

$$\int_V \boldsymbol{B}^T \boldsymbol{\sigma}(\vec{\boldsymbol{u}}) dV - \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS = 0 \tag{E.39}$$

Since equation (E.39) cannot be be solved for directly, a practical solution consists in resorting to iterative methods to minimize the residual quantity $\boldsymbol{R}(\vec{\boldsymbol{u}})$ defined as:

$$\boldsymbol{R}(\vec{\boldsymbol{u}}) = \int_V \boldsymbol{B}^T \boldsymbol{\sigma}(\vec{\boldsymbol{u}}) dV - \int_S \boldsymbol{N}^T \vec{\boldsymbol{t}} dS \tag{E.40}$$

For such purpose, the Newton-Raphson method can be employed. With this method, a new solution for the displacement $\vec{\boldsymbol{u}}_{i+1}$ can be evaluated at each iteration $i+1$ as:

$$\vec{\boldsymbol{u}}_{i+1} = \vec{\boldsymbol{u}}_i - \left[\frac{\partial \boldsymbol{R}(\vec{\boldsymbol{u}}_i)}{\partial \vec{\boldsymbol{u}}} \right]^{-1} \cdot \boldsymbol{R}(\vec{\boldsymbol{u}}_i) \tag{E.41}$$

Further, by denoting $\Delta\vec{\boldsymbol{u}}_{i+1} = \vec{\boldsymbol{u}}_{i+1} - \vec{\boldsymbol{u}}_i$ the correction obtained at each iteration of the Newton-Raphson, equation (E.41) can be conveniently rewritten as:

$$\boldsymbol{K'}(\vec{\boldsymbol{u}}_i) \left[\Delta\vec{\boldsymbol{u}}_{i+1} \right] = -\boldsymbol{R}(\vec{\boldsymbol{u}}_i) \tag{E.42}$$

where the expression of quantity $\boldsymbol{K'}(\vec{\boldsymbol{u}}_i)$ is given from equation (E.40) as:

$$\boldsymbol{K'}(\vec{\boldsymbol{u}}_i) = \frac{\partial \boldsymbol{R}(\vec{\boldsymbol{u}}_i)}{\partial \vec{\boldsymbol{u}}} = \int_V \boldsymbol{B}^T \frac{\partial \boldsymbol{\sigma}(\vec{\boldsymbol{u}}_i)}{\partial \vec{\boldsymbol{u}}} dV = \int_V \boldsymbol{B}^T \frac{\partial \boldsymbol{\sigma}(\vec{\boldsymbol{u}}_i)}{\partial \boldsymbol{\epsilon}} \frac{\partial \boldsymbol{\epsilon}}{\partial \vec{\boldsymbol{u}}} dV \tag{E.43}$$

such that equation (E.43) can finally be written as:

$$\boldsymbol{K'}(\vec{\boldsymbol{u}}_i) = \int_V \boldsymbol{B}^T \boldsymbol{L}_i^{tan} \boldsymbol{B} dV \tag{E.44}$$

where $\boldsymbol{L}_i^{tan}$ is referred to as the tangent modulus, or material jacobian, and is defined as:

$$\boldsymbol{L}_i^{tan} = \left. \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\epsilon}} \right|_{\vec{\boldsymbol{u}}_i} \tag{E.45}$$

In this context, the Newton-Raphson procedure iterates until convergence is reached, at which step the final value displacement field $\vec{\boldsymbol{u}}_{conv}$ is obtained as:

$$\vec{\boldsymbol{u}}_{conv} = \vec{\boldsymbol{u}}_0 + \sum_{i=1}^{n} \Delta \vec{\boldsymbol{u}}_i \tag{E.46}$$

where $\vec{\boldsymbol{u}}_0$ denotes the initial guess for the displacement, and n is the total number of iterations to achieve convergence.

E.5 Element and shape functions

In this work, height-noded C3D8 elements are used. With this, any field variable X can be interpolated from its nodal values X^a from equation (E.11) as:

$$X(\vec{\boldsymbol{x}}) = \sum_{a=1}^{N_d=8} N^a(\vec{\boldsymbol{x}}) X^a \tag{E.47}$$

where $N_d = 8$ for C3D8 elements, and for which the shape functions $\{N^a(\vec{\boldsymbol{s}})\}_{a=1,8}$ of the curvilinear abscissa $\vec{\boldsymbol{s}}$ with spatial components $\{s_i\}_{i=1,3} \in [-1, 1]$ are given as:

$$
\begin{aligned}
N^1(\vec{\boldsymbol{s}}) &= 1/8(1-s_1)(1-s_2)(1-s_3) \\
N^2(\vec{\boldsymbol{s}}) &= 1/8(1+s_1)(1-s_2)(1-s_3) \\
N^3(\vec{\boldsymbol{s}}) &= 1/8(1+s_1)(1+s_2)(1-s_3) \\
N^4(\vec{\boldsymbol{s}}) &= 1/8(1-s_1)(1+s_2)(1-s_3) \\
N^5(\vec{\boldsymbol{s}}) &= 1/8(1-s_1)(1-s_2)(1+s_3) \\
N^6(\vec{\boldsymbol{s}}) &= 1/8(1+s_1)(1-s_2)(1+s_3) \\
N^7(\vec{\boldsymbol{s}}) &= 1/8(1+s_1)(1+s_2)(1+s_3) \\
N^8(\vec{\boldsymbol{s}}) &= 1/8(1-s_1)(1+s_2)(1+s_3)
\end{aligned} \tag{E.48}
$$

Spatial derivatives of the shape functions given in expressions (E.48) are obtained as:

$$
\begin{aligned}
N^1_{,1}(\vec{\boldsymbol{s}}) &= -1/8(1-s_2)(1-s_3) \\
N^1_{,2}(\vec{\boldsymbol{s}}) &= -1/8(1-s_1)(1-s_3) \\
N^1_{,3}(\vec{\boldsymbol{s}}) &= -1/8(1-s_1)(1-s_2) \\
&\ldots
\end{aligned} \tag{E.49}
$$

APPENDIX F

DISCRETE FOURIER TRANSFORMS AND THE FFT ALGORITHM

In this appendix, the mathematical concepts underlying FFT-based approaches are presented. First, the expansion of functions in Fourier series is recalled and the resulting continuous Fourier transform is derived. Then, the discrete Fourier transform for numerical application is introduced and the efficient Fast Fourier Transform algorithm for calculating the latter is exposed.

F.1 Fourier series

Let us consider a piecewise continuous function f over the one-dimensional Cartesian space $\mathbb{R}$. If function f is L-periodic such that $f(x) = f(x+L), \forall x \in \mathbb{R}$, then one can define for all $N \in \mathbb{N}$ its Fourier series expansion $S_N f(x)$, also referred to as Fourier projection, as:

$$S_N f(x) = a_0(f) + \sum_{k=1}^{N} \left(a_k(f) \cos \frac{k\pi x}{L} + b_k(f) \sin \frac{k\pi x}{L} \right) \tag{F.1}$$

where the trigonometric Fourier coefficients are defined by:

$$\begin{aligned} a_0(f) &= \frac{1}{L} \int_{[L]} f(x) dx \\ a_n(f) &= \frac{2}{L} \int_{[L]} f(x) \cos \frac{n\pi x}{L} dx \\ b_n(f) &= \frac{2}{L} \int_{[L]} f(x) \sin \frac{n\pi x}{L} dx \end{aligned} \tag{F.2}$$

where $[L]$ denotes every segment of length L. Further, using Euler's formula, it can

be easily shown that expression (F.1) can be equivalently written as:

$$S_N f(x) = \sum_{k=-N}^{N} \widehat{f}(k) e^{\mathrm{i}\frac{k\pi x}{L}} \tag{F.3}$$

with

$$\widehat{f}(k) = \frac{1}{L} \int_0^L f(x) e^{-\mathrm{i}\frac{k\pi x}{L}} dx \tag{F.4}$$

where the coefficients $\{\widehat{f}(k)\}_{-N \leq k \leq N}$ are called the spectral or Fourier coefficients of function f, and $\mathrm{i} = \sqrt{-1}$ is the complex number. One of the most remarkable properties of the Fourier series is that, when f is piecewise continuous, defined and integrable over $\mathbb{R}$, its expansion $S_N f(x)$ is convergent for all x when $N \to \infty$ and its sum is:

$$S_\infty f(x) = \sum_{k=-\infty}^{\infty} \widehat{f}(k) e^{\mathrm{i}\frac{k\pi x}{L}} = \frac{1}{2}\left(f(x^+) + f(x^-)\right) \tag{F.5}$$

In other words, such property, known as the Dirichlet's theorem, states that the infinite Fourier series $S_\infty f(x)$ is equal to $f(x)$ where f is continuous. Furthermore, one very interesting practical property of $S_N f(x)$ is its exponential convergence $|S_N f(x) - f(x)| \leq e^{-\eta \sqrt[\alpha]{N}}, x \in \mathbb{R}$, where the root exponent $\alpha > 1$ and factor η are tied to the global smoothness of f. Thus, $S_N f(x)$ enjoys the *spectral accuracy*, that is, benefits from the exponential decay rate of $S_N f(x) - f(x)$. Therefore, $S_N f(x)$ constitutes a good approximation of $f(x)$, and conversely, and more importantly, a good approximation of f can be practically reconstructed from a limited set of its Fourier coefficients $\{\widehat{f}(k)\}_{-N \leq k \leq N}$ according to relation (F.3).

F.2 Continuous Fourier transform

The case $L \to \infty$ is of particular interest since it no longer requires the periodicity of function f. In this case, the infinite sum in (F.5) tends to an integral such that the

function can be expressed as:

$$f(x) = \int_{-\infty}^{+\infty} \widehat{f}(\xi) e^{\mathrm{i}\xi x} d\xi \tag{F.6}$$

with

$$\widehat{f}(\xi) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} f(x) e^{-\mathrm{i}\xi x} dx \tag{F.7}$$

where $\xi = \frac{n\pi}{L}$. Expressions (F.6) is called the Fourier integral form of f and (F.7) is known as the continuous Fourier transform of f. However, note that depending on the domain of application, the definition of the Fourier transform and its inverse is not unique. Specifically, factors 2π or $\sqrt{\pi}$ may be removed or added, and the sign in the exponential may vary depending on the formulations. In the current work, the definition of the Fourier transform is given by:

$$\widehat{f}(\xi) = \int_{-\infty}^{+\infty} f(x) e^{-\mathrm{i}\xi x} dx \tag{F.8}$$

Practically, expression (F.8) can be regarded as a way to transform the original function f defined in the real space into its spectral/frequency representation defined in the Fourier space via the calculation of the Fourier coefficients. Then, the original function f can be retrieved from its spectral representation from the inverse Fourier transform that is defined in this work as:

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \widehat{f}(\xi) e^{\mathrm{i}\xi x} d\xi \tag{F.9}$$

F.3 Discrete Fourier transforms

Any numerical application involving Fourier transforms requires to work with discretized data. Let us therefore assume that the piecewise continuous L-periodic function f is sampled at N evenly spaced grid points along the x direction for $0 \leq x \leq L$.

If we denote $\delta = L/N$ the interval between two consecutive sample positions, then the sampled function values $\{f_j\}_{0 \leq i \leq N-1}$ are defined as:

$$f_j = f(x_j), \quad x_j = \delta j, \quad j = 0, ..., N-1 \tag{F.10}$$

Taking advantage of the spectral accuracy resulting from the exponential convergence of the Fourier series given in (F.1), let us assume that writing the Fourier series of a function f sampled at N grid points requires N Fourier coefficients values to obtain a good approximation. Thus, following the expression given in (F.1) in the case of continuous Fourier series, the approximated discrete function can be reconstructed from its Fourier coefficients $\{\widehat{f}_n\}_{0 \leq n \leq N-1}$, such that one can has:

$$\begin{aligned} f_j = f(x_j) &= \sum_{n=0}^{N-1} \widehat{f}_n e^{\mathrm{i} k_n x_j}, \quad j = 0, ..., N-1 \\ \text{with} \quad \widehat{f}_n &= \widehat{f}(k_n) = \frac{1}{L} \int_0^L f(x) e^{-\mathrm{i} k_n x} dx \end{aligned} \tag{F.11}$$

where k_n is the discrete wave number and x_j the discrete j-th position in the real space at which f_j is sampled. Values of the discrete wave numbers k_n should be chosen such that the Fourier coefficients satisfy the periodicity conditions and should be equally spaced. The following choice satisfies both these conditions:

$$k_n = \frac{2\pi n}{N\delta}, \quad n = 0, ..., N-1 \tag{F.12}$$

where $\omega = \frac{2\pi}{L} = \frac{2\pi}{N\delta}$ is the pulsation associated with a L-periodic function. Using the definitions given in (F.10) and (F.12), the discrete transformation expressed in (F.11) writes:

$$f(x_j) = f_j = \sum_{n=0}^{N-1} \widehat{f}_n e^{\mathrm{i} \frac{2\pi n}{N\delta} \delta j} = \sum_{n=0}^{N-1} \widehat{f}_n e^{\mathrm{i} \frac{2\pi j n}{N}}, \quad j = 0, ..., N-1 \tag{F.13}$$

In order to reconstruct the discrete function f, the N Fourier coefficients $\{\widehat{f}_n\}_{0\leq n\leq N-1}$ in (F.13) need to be computed. These coefficients define the spectral representation of function f in the frequency domain. Numerically, as opposed to their continuous formulation given in (F.11), the discrete Fourier coefficients $\widehat{f}_n$ are obtained from the sampled values $\{f_j\}_{0\leq j\leq N-1}$ through the discrete Fourier transform (DFT). Given that N sampled values f_i are known, it cannot be expected to obtain more than N independent output values $\widehat{f}_n$. More importantly, from the fundamental sampling theorem, there exists a special frequency $\xi_c = \frac{1}{2\delta}$ associated to any sampling interval δ. This frequency is called the Nyquist frequency and the spectral representation of f is said to be not aliased only within the frequency interval $[-\xi_c, \xi_c]$. Therefore, to avoid aliasing and ensure that the function can be entirely recovered without loss of information, the N discrete values $\widehat{f}_n$ are solely estimated in the range $[-\xi_c, \xi_c]$, i.e. at discrete frequencies:

$$\xi_n = \frac{n}{N\delta}, \quad n = -\frac{N}{2}, ..., \frac{N}{2} \tag{F.14}$$

for which the extremes values at $n = -N/2$ and $n = N/2$ corresponds to the Nyquist frequencies $-\xi_c$ and ξ_c, respectively. Note that equation (F.14) provides a set of $N+1$ frequencies, but since the extreme values produce identical spectral coefficients by periodicity, only N independent values are obtained. If we associate each frequency ξ_n with its wave number k_n given in (F.12), the discrete formulation in (F.11), i.e. the Discrete Fourier Transform (DFT) of f – to obtain the discrete Fourier coefficients $\{\widehat{f}_n\}_{0\leq n\leq N-1}$ – writes:

$$\begin{aligned}\widehat{f}_n = \widehat{f}(k_n) &= \frac{1}{L}\int_0^L f(x)e^{-\mathrm{i}k_n x}dx \\ &= \frac{1}{L}\sum_{j=0}^{N-1} f(x_j)e^{-\mathrm{i}k_n x_j}\delta = \frac{1}{N}\sum_{j=0}^{N-1} f_j e^{-\mathrm{i}\frac{2\pi jn}{N}}\end{aligned} \tag{F.15}$$

To summarize, the discrete Fourier coefficients $\{\widehat{f}_n\}_{0\leq n\leq N-1}$ of any discrete function f sampled at N evenly spaced grid points can be obtained from the sampled values $\{f_j\}_{0\leq j\leq N-1}$ through the direct DFT expressed in (F.15) as:

$$\widehat{f}_n = \frac{1}{N}\sum_{j=0}^{N-1} f_j e^{-\mathrm{i}\frac{2\pi jn}{N}}, \quad n = 0, ..., N-1 \tag{F.16}$$

where the Fourier coefficients $\{\widehat{f}_n\}_{0\leq n\leq N-1}$ define the discrete spectral representation of the discrete function f in the frequency domain. Conversely, from its N Fourier coefficients, the values $f_j = f(x_j)$ of any discrete function f can be accurately recovered at its N grid sampled positions x_j through the inverse DFT defined in expression (F.13) as:

$$f_j = \sum_{n=0}^{N-1} \widehat{f}_n e^{\mathrm{i}\frac{2\pi jn}{N}}, \quad j = 0, ..., N-1 \tag{F.17}$$

Naturally, by relating the DFT and the inverse DFT to the spectral accuracy of the Fourier series, it appears that increasing the number of sampling values N leads to an increase of resolution and accuracy in the discrete Fourier representation and consequently in the reconstruction from the discrete Fourier coefficients. Also, one can notice from expressions (F.16) and (F.17) that the complete determination of the discrete Fourier transforms in their fundamental form each initially requires $\mathcal{O}(N^2)$ calculations.

Fortunately, the Fourier transforms are not limited to one-dimensional functions, and can be easily extended to higher dimensions. For instance, the Fourier transform (and its inverse) of any complex function $f = f(\vec{\boldsymbol{x}})$ defined over the three-dimensional Cartesian space $\mathbb{R}^3$ with coordinates $\vec{\boldsymbol{x}} = \{x_i\}_{i=1,3}$ can be extended from expressions (F.8) and (F.9) so as to write:

$$\widehat{f}(\vec{\boldsymbol{\xi}}) = \iiint_{-\infty}^{+\infty} f(\vec{\boldsymbol{x}}) e^{-\mathrm{i}\vec{\boldsymbol{\xi}}\cdot\vec{\boldsymbol{x}}} d\vec{\boldsymbol{x}}$$

$$f(\vec{\boldsymbol{x}}) = \frac{1}{(2\pi)^3} \iiint_{-\infty}^{+\infty} \widehat{f}(\vec{\boldsymbol{\xi}}) e^{\mathrm{i}\vec{\boldsymbol{\xi}}\cdot\vec{\boldsymbol{x}}} d\vec{\boldsymbol{\xi}} \tag{F.18}$$

where $\vec{\boldsymbol{\xi}} = \{\xi_i\}_{i=1,3}$ denotes the frequency in the Fourier space and $\vec{\boldsymbol{\xi}} \cdot \vec{\boldsymbol{x}} = \xi_i x_i$ is the dot product between frequency and spatial coordinates $\vec{\boldsymbol{\xi}}$ and $\vec{\boldsymbol{x}}$ using indicial notation. Such three-dimensional definitions are those used in relations (3.17) when introducing spectral methods for continuum mechanics in Section 3.3. Essentially, the three-dimensional Fourier transforms expressed in equations (F.18) can be regarded as one-dimensional Fourier transforms sequentially taken on each dimension.

For any periodic numerical simulation performed in 3D such as in DDD-FFT simulations, the primary volume $V = L_1 \times L_2 \times L_3$ can be discretized into N_i voxels in each i-th spatial direction, leading to the determination of a 3D grid of $N = N_1 \times N_2 \times N_3$ points $\{\vec{\boldsymbol{x}}\}$ in the Cartesian real space. Following the definition in (F.18) and the DFT expression in (F.16), the direct 3D DFT of any L_i-periodic discrete function $f = f(x_1, x_2, x_3)$ evenly sampled in all spatial directions $i = \{1, ..., 3\}$ with intervals $\delta_i = L_i/N_i$ is computed as:

$$\widehat{f}_{npq} = \frac{1}{N_1 N_2 N_3} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} \sum_{k=0}^{N_3-1} f(\vec{\boldsymbol{x}}_{ijk}) e^{-\mathrm{i}(\vec{\boldsymbol{k}}_{npq}\cdot\vec{\boldsymbol{x}}_{ijk})}, \quad n, p, q = 0, ..., N_{1,2,3} - 1 \tag{F.19}$$

where $\vec{\boldsymbol{k}}_{npq} = \left(\frac{2\pi n}{N_1\delta_1}, \frac{2\pi p}{N_2\delta_2}, \frac{2\pi q}{N_3\delta_3}\right)$ is the 3D discrete wave number and $\vec{\boldsymbol{x}}_{ijk} = (\delta_1 i, \delta_2 j, \delta_3 k)$ is the spatial position at which function f is sampled. By denoting $f(\vec{\boldsymbol{x}}_{ijk}) = f_{ijk}$ and using the definitions of $\vec{\boldsymbol{k}}_{npq}$ and $\vec{\boldsymbol{x}}_{ijk}$, the 3D DFT of f can be written as:

$$\widehat{f}_{npq} = \frac{1}{N} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} \sum_{k=0}^{N_3-1} f_{ijk} e^{-\mathrm{i}2\pi\left(\frac{in}{N_1} + \frac{jp}{N_2} + \frac{kq}{N_3}\right)}, \quad n, p, q = 0, ..., N_{1,2,3} - 1 \tag{F.20}$$

and following equation (F.17), the inverse DFT of f is given as:

$$f_{ijk} = \sum_{n=0}^{N_1-1} \sum_{p=0}^{N_2-1} \sum_{q=0}^{N_3-1} \widehat{f}_{npq} e^{-\mathrm{i}2\pi\left(\frac{in}{N_1}+\frac{jp}{N_2}+\frac{kq}{N_3}\right)}, \quad i,j,k = 0,...,N_{1,2,3}-1 \tag{F.21}$$

Note that 3D DFTs given in equations (F.20) and (F.21) can be computed using the FFT algorithm. Practically, the transforms are obtained by sequentially applying the one-dimensional FFT algorithm to the three spatial dimensions of f.

F.4 FFT algorithm: Fast Fourier Transforms

As presented in the previous section, the determination of the DFT (or its inverse) of a function is basically a $\mathcal{O}(N^2)$ process according to their definitions in (F.16) and (F.17), or in (F.20) and (F.21) for their three-dimensional extensions. However, Fourier-based methods would probably not have received that much attention and being so widely employed if the computation of DFT would have practically required $\mathcal{O}(N^2)$ calculations. As a matter of fact, in 1965, James Cooley and John Tukey proposed a fast algorithm to compute the DFT of a function, taking advantage of the recursivity appearing between the different spectral coefficients – which was already spotted by Danielson and Lanczos in 1942 – and coupling it with a numerical scheme based on the divide-and-conquer technique [51]. Incidentally, it was later discovered that this algorithm has already been devised by Gauss in 1805. With this fast algorithm, commonly known as the Fast Fourier Transform (FFT) algorithm, the number of computations required to obtain the DFT (or its inverse) falls to $\mathcal{O}(N \log N)$. Obviously, for large values of N, the difference between $N \log N$ and N^2 rapidly becomes considerable, hence the phenomenal success of numerical Fourier-based methods. However, the only limitation is that, to enjoy such theoretical performances, the choice of N should be restricted to power of 2 (e.g. $N = 64, 128, 256, 512, ...$). If not the case, the performance of the algorithm drastically departs from the $\mathcal{O}(N \log N)$ complexity.

APPENDIX G

SUPPLEMENTARY MATERIAL FOR THE DDD-FFT APPROACH

G.1 Removal of the Gibbs oscillations

One of the main numerical difficulties associated with FFT-based spectral methods is the apparition of spurious oscillations in the computed solution fields. This phenomenon occurs in taking Fourier transforms of discontinuous fields, and is for instance illustrated in figure 3.5(b) in the context of this work. In the DDD-FFT approach, the undesirable Gibbs effect arises because of the discontinuities in the plastic strain field induced by dislocation motion – the glide of a dislocation segment produces a jump in the displacement field across its slip plane, see Section 2.2.

For instance, when a single dislocation loop is introduced in the simulation volume, the plastic strain in the volume presents a delta-function like distribution. Thus, for the configuration illustrated in figure 3.4(a), the value of the plastic strain along a y-line taken at height $z = L/2$ on slice (100) at position $x = L/2$, is $\mathbf{0}$ everywhere, except at position $y = L/2$ where its value is related to the area swept by the loop during its introduction in the simulation volume. Naturally, interpolating such delta distribution with a finite set of sinusoidal functions – such as done when computing the Fourier coefficients via the FFT algorithm – leads to the apparition of spurious oscillations in the areas surrounding the discontinuities, that further spread to the region of smoothness when computing the inverse discrete Fourier transforms. Consequently, oscillations arise in the resulting stress calculation leading to a global loss of accuracy.

In order to tackle this undesirable effect pertaining to spectral methods in general, several techniques have been proposed. In signal processing where the FFT is extensively used, low-pass filters on the discrete spectral signal are generally used in order to attenuate the numerical oscillations carried out by high frequencies. Following this, exponential filters have for instance been used in the spectral method developed in [75] to solve for the dislocation density transport equation. Alternatively, an approach a based on a *numerical spreading* was proposed in the work of Brenner et al. [26]. In their two-dimensional implementation of static field dislocation mechanics, the authors proposed to spread any dislocation density $\boldsymbol{\alpha}$ defined at a pixel across its neighboring pixels and studied the effect of different distribution schemes. The authors reported that spreading the core of a dislocation over 3×3 pixels using a triangular distribution was sufficient to strongly attenuate the oscillations produced from the Gibbs phenomenon. As another approach, the use of discrete gradient operators in the Fourier space has provided an efficient manner to attenuate the oscillations. In this approach, the continuous derivatives taken in the Fourier space are replaced by discrete operators derived from finite difference schemes. Thus, Berbenni and co-workers used discrete gradient operators based on the centered-scheme in their implementation of the two-dimensional static field g-disclinations mechanics [20]. Note also that very recently, Willot proposed a novel discrete operator based on a *rotational scheme* allowing to substantially accelerate the convergence of iterative spectral methods [258] while reducing the oscillations. Interestingly, in developing the DDD-FFT method, it appears that the efficiency of the different methods to attenuate the Gibbs oscillations cannot be assessed from a general perspective, but strongly depends on the formulation of the specific problem to which it is applied. Therefore, while the discrete gradient approach based on the centered-scheme is seen to perform very well when solving the Poisson equation in two dimensions, it is seen to completely fail when applied to the DDD-FFT method.

In this work, several numerical techniques to attenuate the Gibbs oscillations have been investigated. First, the implementation of the *numerical spreading* technique used in the homogeneous FFT formulation and whose effects are studied in Section 3.4.2 is detailed in Appendix G.1.1. Then, the discrete gradient operators approach and their results are discussed in Appendix G.1.2. Finally, the effects of the discrete gradients on the convergence of the heterogeneous iterative schemes are examined in Appendix G.2.

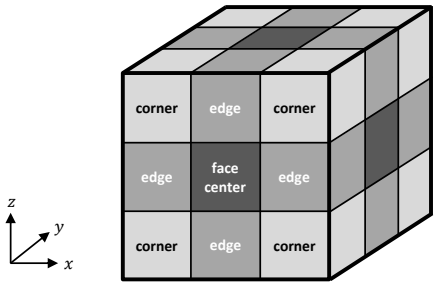
G.1.1 Numerical spreading

Following the work of Brenner and co-workers [26], a similar *numerical spreading* of the plastic strain with a three-dimensional triangular distribution over $3 \times 3 \times 3$ voxels has been implemented to the DDD-FFT code. Practically, the plastic shear computed at each grid point from the numerical regularization in equations (3.5) to (3.8) (for which an analytical alternative procedure is developed in Section 3.6) is distributed over the 27 neighboring voxels using a triangular distribution. As depicted in figure G.1(a), the 27 neighboring voxels of each grid point can be classified into four sets of voxels as a function of their distance to the center point, such that the ensemble comprises 1 center voxel, 6 face center voxels, 12 edge voxels and 8 corner voxels. Thus, a triangular distribution around the center point consists in affecting the following weights to the different sets: w to the center voxel, w/n to a face center voxel, w/n^2 to an edge voxel, and w/n^3 to a corner voxel, where n is the spread parameter of the distribution. To ensure that the plastic shear transferred to the mesh remains valid, the total weight W of the distribution must respect:

$$W = \sum_i n_i w_i = w + \frac{6w}{n} + \frac{12w}{n^2} + \frac{8w}{n^3} = 1 \tag{G.1}$$

where coefficients n_i and w_i denote the number of voxels and the weigth associated to each set of voxels i. To select the best distribution, the effect of the spreading for

(a)



(b)

value of n	value of $w = w_1$
$n = 1$	$w = 1/27 \approx 0.037$
$n = 2$	$w = 1/8 = 0.125$
$n = 3$	$w = 27/125 = 0.216$
$n = 4$	$w = 8/27 \approx 0.296$

Figure G.1: (a) Classification of the different sets of voxels in an ensemble of 27 neighbors surrounding the center voxel. (b) Values of the weight w for the different tested values of the spreading parameter n. Here $w = w_1$ refers to the weight of the center voxel. The weights of the other sets of voxels are calculated using the quantities given in equation (G.1).

different values of n has been tested. The different values of n and their resulting weight w are given in table G.1(b). Logically, the smaller n, the smoother the distribution. In contrast, the removal of the numerical spreading is theoretically recovered for $n \to \infty$, i.e. for $w \to 1$. From the results (not reported here), it is seen that for the value $n = 1$, the spread is too important in that it produces a smearing out of the dislocation core. As a result, the oscillations disappear, but the description of the stress field in the vicinity of the core departs from the analytical solution. On the opposite, choosing a value of $n = 4$ does not produce a sufficient spread so that the oscillations are not fully removed. However, it appears that the spread for $n = 2$ offers a satisfactory compromise: the description of the dislocation core remains accurate while the oscillations become imperceptible. Besides, despite the fact that a two-dimensional setting was used, the triangular distribution used in [26] was tacitly based on this choice of parameter. Note however that other types of distributions could be considered. Ideally, the best achievable distribution would be that which ensures consistency with the non-singular formulation of Cai et al. [35] that is used to compute local interactions, so as to prevent any double counting to occur [100]. Such approach was for instance suggested, although in a different context, in the work of

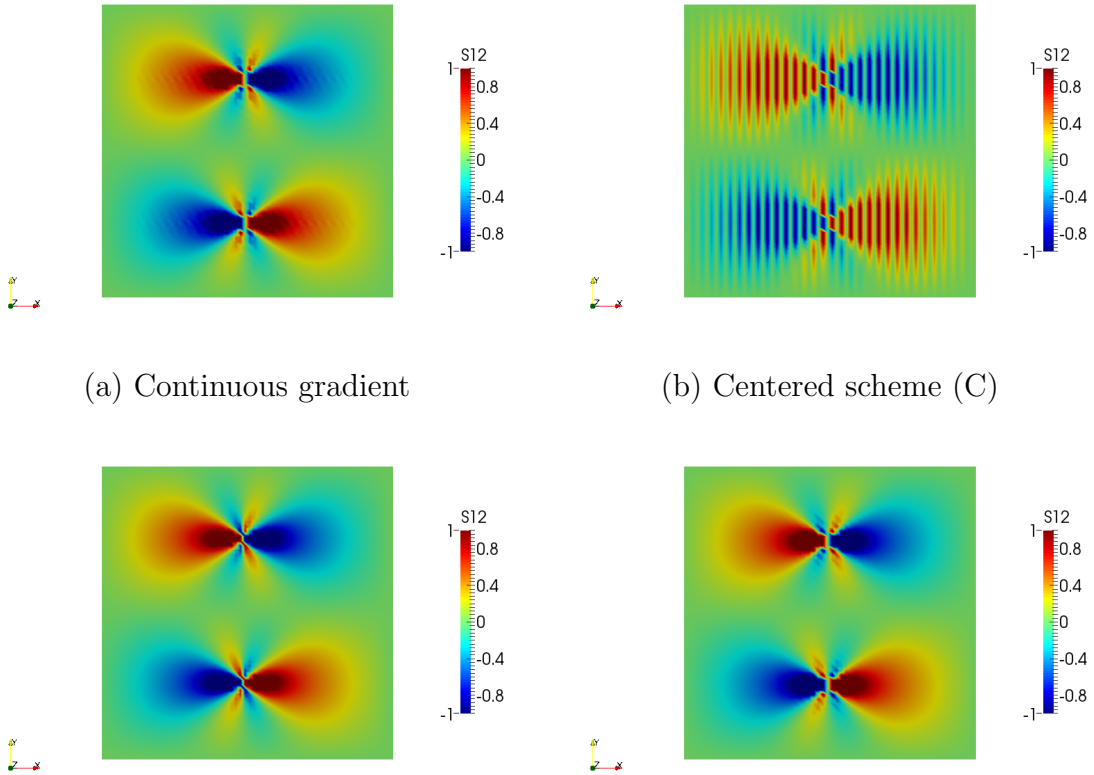
Liu and al. [155]. Nevertheless, for the sake of simplicity, a numerical spreading with parameter $n = 2$ is chosen in this work.

G.1.2 Discrete gradient operators

In this section, the effect of the discrete gradients operators presented in Section 3.5.3.2 on the resulting stress field of a static dislocation in an homogeneous medium is assessed. To this end, the setting used in Section 3.4.2.1 and depicted in figure 3.4 is reused and results are given for a (yz) slice at $x = L/2$ made of $N \times N$ pixels. The dislocation loop with unit Burgers $\vec{\boldsymbol{b}}$ in the y-direction is similarly introduced using a Volterra-like process in the middle of the simulation volume of size $V = 1 \times 1 \times 1$. The medium is chosen as elastically isotropic with Lamé constants $\lambda^1 = 0.4$ and $\mu^1 = 0.6$ and no load is prescribed, i.e. the average imposed strain is set to $\boldsymbol{E} = \boldsymbol{0}$.

Results for the σ_{23} stress component for a resolution of 64×64 pixels on the (yz) slice are presented in figure G.2. When using the continuous gradient operator (3.89), Gibbs oscillations appear. Here, the *numerical spreading* technique employed in Section 3.4.2.1 is not used in order to solely test for the effect of discrete gradient operators alone. As already explained in details, these oscillations result from the spectral representation of the discontinuous (delta-like) plastic strain field distribution produced by the dislocation loop. Interestingly, it is seen that the centered scheme (C) using discrete operator (3.83) leads to a disastrous propagation of these oscillations through the entire volume. Such observation seems all the more surprising that the centered scheme has been reported to be particularly efficient to remove oscillations in the context of static field dislocation mechanics [20]. Beyond any mathematical and numerical considerations, this result highlights the importance in selecting the appropriate strategy in accordance with the specific problem when dealing with spectral approaches.

Conversely, the backward (W) and rotational (R) discrete schemes allow for a



(a) Continuous gradient

(b) Centered scheme (C)

(c) Backward scheme (W)

(d) Rotational scheme (R)

Figure G.2: Effect of the different discrete gradient schemes on the σ_{23} stress component produced by a static prismatic dipole in an homogeneous isotropic elastic medium for a resolution of 64×64 pixels on the (yz) slice. (a) When using the continuous gradient operator, Gibbs oscillations are produced as a result of the delta function-like distribution of the plastic strain field. (b) With the centered scheme (C), the oscillations widely propagate through the whole domain. In (c) and (d), the oscillations are removed with schemes (W) and (R), but the description of the core is different for both schemes.

removal of all spurious fluctuations by producing smooth stress fields within the volume. However, a difference between both schemes is spotted in the vicinity of the dislocation core. This difference is highlighted in figure G.3 where the evolution of the stress is plotted along a y-line passing through the core of the dislocation. As shown, the rotational scheme (R) produces a clean profile in which the staircase evolution of the continuous operator is entirely smoothed out. Alternatively, the backward scheme (W) exhibits a slight loss of accuracy in the vicinity of the core as attested by the

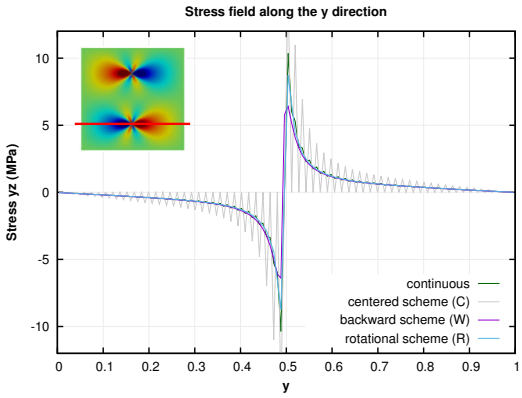
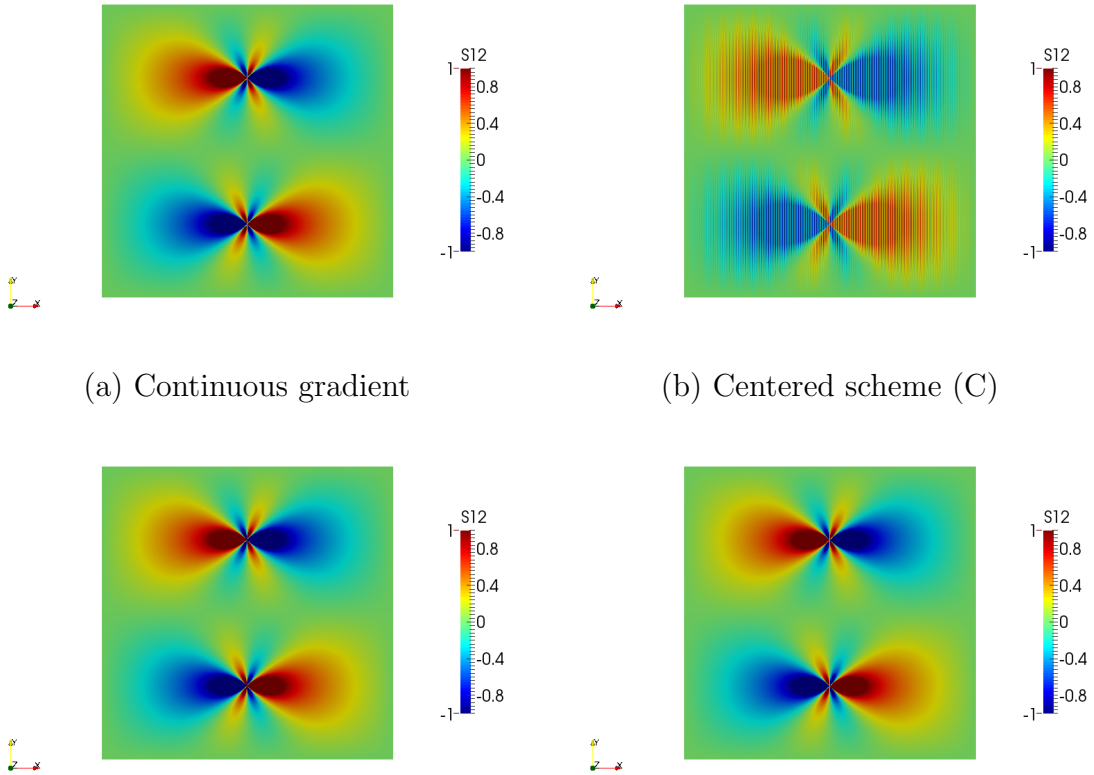


Figure G.3: Evolution of the σ_{23} stress component produced by a prismatic dislocation loop for a resolution of 128×128 pixels on the (yz) slice. The evolution is plotted along the red y-line passing through the core of the dislocation as shown in the inset. While the rotational scheme (R) produces a smooth and accurate profile in which all oscillations have been removed, the backward scheme (W) suffers from a slight loss of accuracy in the immediate vicinity of the core whose position has been slightly translated due to its unsymmetrical definition.

lower maximum magnitude that is obtained. Also, by looking closely at figure G.3, it appears that its unsymmetrical definition (see equation (3.86)) leads to a small translation of the actual position dislocation of the dislocation core.

In figure G.4, the results for the σ_{yz} stress component on the (yz) slice are reported for a finer resolution of 512×512 pixels. Globally, the conclusions are similar than that drawn for a resolution of 64×64 pixels, and the oscillations produced by the centered scheme (C) are still preventing the use of this approach. Furthermore, it becomes apparent than the Gibbs oscillations naturally fade with increasing resolution, and, consequently, it is expected that the different schemes will eventually converge as $N \to \infty$, where N is the number of pixels.

From these observations, it appears that the *numerical spreading* introduced in Section 3.4.2.1 and employed in the homogeneous DDD-FFT formulation can be replaced by the use of the discrete operators (W) or (R) defined in equations (3.85) and (3.87), respectively, or both can techniques be combined so as to simultaneously ensure that spurious oscillations are removed while preserving a mean to control the



(a) Continuous gradient

(b) Centered scheme (C)

(c) Backward scheme (W)

(d) Rotational scheme (R)

Figure G.4: Effect of the different discrete gradient schemes on the σ_{23} stress component produced by a static prismatic dipole in an homogeneous isotropic elastic medium for a resolution of 512×512 pixels on the (yz) slice. Conclusions are similar than that drawn for a resolution of 64×64 pixels, although the increase in resolution naturally induces an increase in accuracy and an attenuation of the Gibbs oscillations in (a).

width of the dislocation core via the selection of the spreading distribution.

G.2 Validation and comparison of the heterogeneous schemes

In this section, the accuracy and convergence rate of the different schemes presented in Section 3.5.3 are compared in the context of the DDD-FFT framework. To ease the comparison with existing results, the following results are given for a two-dimensional setting. Given that, to the author's knowledge, no analytical solution exits for an

elasto-plastic problem in heterogeneous elasticity, the heterogeneous DDD-FFT approach will be assessed here for elastic problems, i.e. when no dislocations are present in the medium. The presence and effects of dislocations in heterogeneous elastic media will be investigated in Section 5.2 to study the interactions between dislocations and phases or precipitates.

To validate the implementation of the heterogeneous DDD-FFT approach and that of the different iterative schemes presented in the previous section (for which full details are given in Section 3.6.3), the results produced by Willot in reference [258] where the different discrete gradient operators are compared are first reproduced. To this end, the setting depicted in figure G.5(a) is used: a square inclusion with elastic constants λ^2 and μ^2 is embedded at the top-right corner of a matrix with elastic constants $\lambda^1 = 0.4$ and $\mu^1 = 0.6$ and of size $L \times L$. In this configuration, the setting is linear elastic and no dislocation is present in the domain. To simulate the deformation, an average pure shear $E_{ij} = 1/2(\delta_{i1}\delta_{j2} + \delta_{i2}\delta_{j1})$ is imposed to the periodic medium. The medium size is set to $L = 1$ and the contrast between the inclusion and the matrix defined as

$$K = \frac{\lambda^2}{\lambda^1} = \frac{\mu^2}{\mu^1} \tag{G.2}$$

is first varied between values of $K = 1$ and $K = 10^3$, i.e. for stiff inclusions.

The case $K = 1$ corresponds to an elastic homogeneous medium. In this case, all schemes are converging to an uniform elastic response in a single step. Although not fundamental, this result ensures that the heterogeneous DDD-FFT formulation can be regarded as a general framework to which homogeneous elasticity is a special case. The analytical result of the σ_{yz} component for a contrast of $K = 10^3$ is provided in figure G.5(b). As no analytical solution for this specific problem has been found in the literature, the solution corresponds to that consistently obtained with the (W) and the (R) gradient operators at convergence for a precision $\epsilon^{tol} = 10^{-4}$, for which

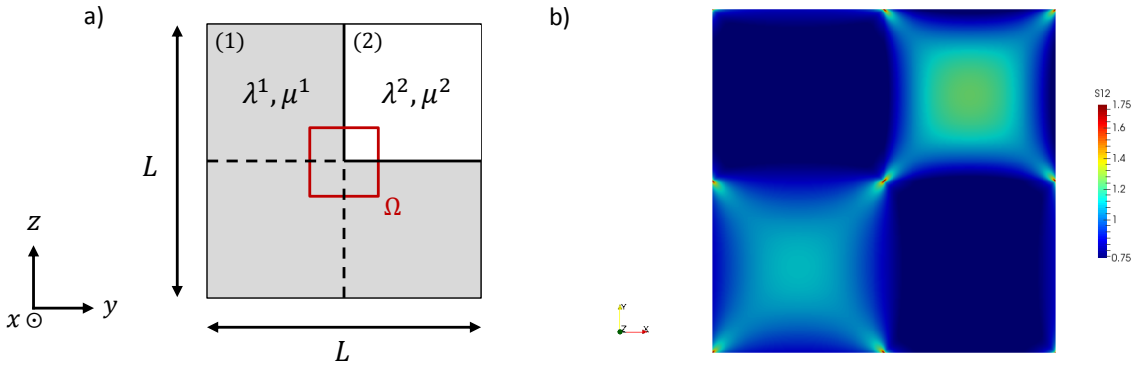
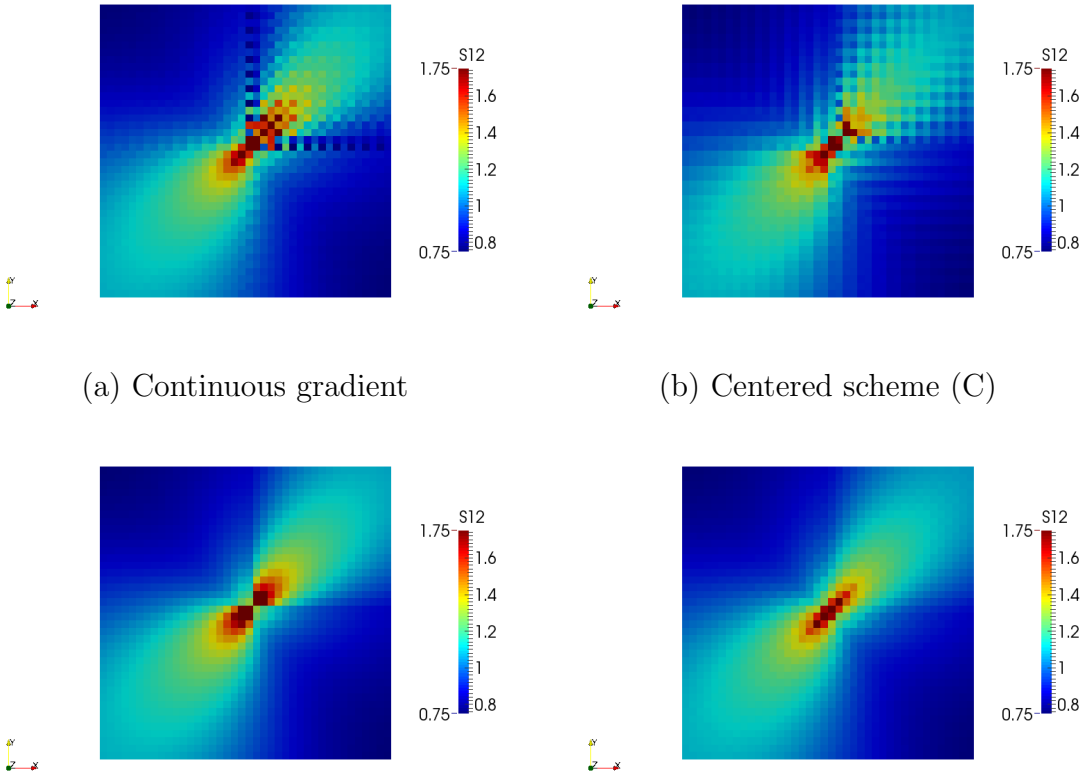


Figure G.5: Setting used for investigating the accuracy of local responses and the convergence rate associated with the different discrete gradient operations and iterative schemes. (a) A sharp white square inclusion (2) is embedded at the top-right corner of a gray matrix (1) of size $L \times L$. The red domain of size $\Omega = \frac{2}{25}L \times \frac{2}{25}L$ centered in the medium delineates the region on which an emphasis on the results will be placed. (b) Analytical solution for the σ_{yz} stress component over the entire medium for a contrast $K = 10^3$. This solution is given as the consistent convergent solution obtained with the (W) and (R) gradient operators for a 512×512 resolution.

the convergence is reached when:

$$\|\boldsymbol{\epsilon}_{i+1} - \boldsymbol{\epsilon}_i\| \leq \epsilon^{tol} \tag{G.3}$$

where $\boldsymbol{\epsilon}_{i+1}$ denotes the full strain vector of length $3N_{pix}^2$ composed of rows of the spatial components of $\boldsymbol{\epsilon}$ for each pixel at iteration $i+1$, where N_{pix} is the number of pixels in each direction. Further, the solution in figure G.5(b) is given for a resolution of 512×512 pixels and is in perfect agreement with that obtained in the work of Willot [258], thereby validating the current implementation. As depicted, and unsurprisingly, the most severe stress concentrations arise at the corners of the inclusion (recall that the medium is periodically repeated in the y and z directions). For this reason the analysis of the accuracy of the local responses of the different schemes is performed in the following for a subdomain $\Omega = \frac{2}{25}L \times \frac{2}{25}L$ centered around the lower-left corner of the inclusion, as delineated in figure G.5(a).



(a) Continuous gradient

(b) Centered scheme (C)

(c) Backward scheme (W)

(d) Rotational scheme (R)

Figure G.6: Effect of the different discrete gradient schemes on the σ_{yz} stress component produced by a square inclusion using isotropic elasticity for a contrast $K = 10^3$ and a resolution of 512×512 pixels. The reported results are obtained after convergence is reached for each scheme (i.e. not after the same number of iterations) and for a precision $\epsilon^{tol} = 10^{-4}$. The continuous gradient operator suffers from Gibbs oscillations while the centered scheme (C) exhibits a downgraded resolution. No oscillations are present when using discrete schemes (W) or (R). While the backward scheme (W) produces a sharp stress distribution at the corner, the latter is more diffuse with the rotational scheme (R).

The results on Ω for the continuous gradient and the three discrete gradient operators are reported in figure G.6 for a grid made of 512×512 pixels total and for the same contrast $K = 10^3$. The results reported in this figure are given after convergence has been reached for each different case. Therefore, they do not correspond to the results obtained after the same number of iterations, but those obtained when the precision defined in (G.3) is set to $\epsilon^{tol} = 10^{-4}$. In case of the continuous gradient

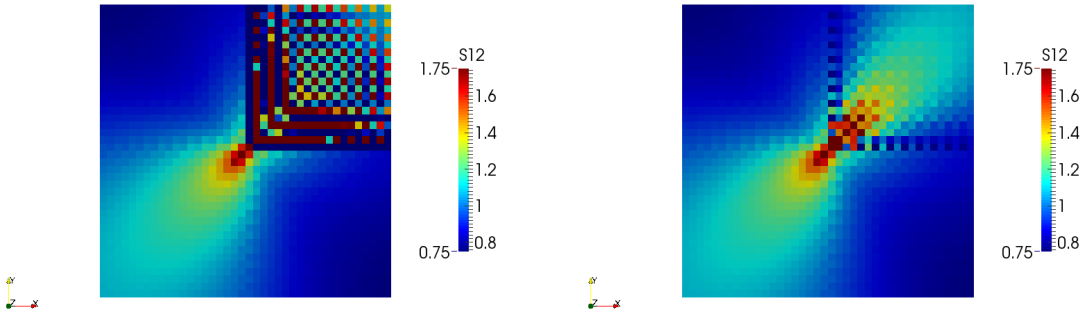
operator, Gibbs oscillations arise along the boundary of the inclusion and the stress state around the corner tip seems muddled. When using the centered scheme (C), it is not evident if the oscillations are removed as the resolution appears as blurred and downgraded outside of the vicinity of the inclusion tip. Conversely, the backward (W) and rotational (R) schemes provide smooth and clean stress fields. However, both schemes exhibit different stress distribution profiles at the corner tip: while the backward (W) method produces a sharp stress distribution with important gradients between neighboring pixels, the rotational scheme provides a narrower but smoother stress distribution.

Up until this point, the effect of the different gradient operators has been solely assessed in terms of accuracy in the local fields. However, their impact on the convergence rate of the implicit schemes has not been investigated. As presented earlier, the iterative methods considered in this work are the *basic* and the *accelerated* schemes, whose formulations are respectively given in equations (3.74) and (3.77), and the *conjugate-gradient* method formulated in (3.79). The number of iterations N required to achieve convergence with the different iterative and discrete gradient schemes for a precision $\epsilon^{tol} = 10^{-4}$ are reported in table G.1. The results are given for a resolution of 128×128 pixels and for contrasts of $K = 2, 10, 10^2,$ and 10^3. Clearly, the contrast is the most important factor on the number of iterations. Thus, for $K = 2$, six to eight iterations are expected for every scheme, and, apart from removing the oscillations, the effect of the iterative and discrete gradient operators on the convergence rate remains limited. However, as the contrast increases to large values, the efficiency of the discrete schemes becomes obvious as they substantially accelerate the convergence. Besides, the use of the *conjugate-gradient* scheme consistently allows for the fastest convergence, and the *accelerated* scheme shows superior efficiency with respect to the *basic* scheme.

Table G.1: Effect of the different iterative and gradient operators schemes on the number of iterations N to reach convergence for a precision $\epsilon^{tol} = 10^{-4}$ and for different contrasts K between the matrix and the upper-right inclusion. Values are reported for a resolution of 128×128 pixels.

Iterative scheme	Gradient operator	$K = 2$	$K = 10$	$K = 100$	$K = 1000$
basic	continuous	8	34	281	2303
accelerated	continuous	8	17	101	587
conjugate gradient	continuous	6	14	42	107
basic	centered (C)	8	30	88	112
accelerated	centered (C)	8	17	39	49
conjugate gradient	centered (C)	7	15	24	28
basic	backward (W)	8	31	107	147
accelerated	backward (W)	8	17	49	67
conjugate gradient	backward (W)	7	14	26	30
basic	rotational (R)	8	25	47	51
accelerated	rotational (R)	8	17	24	25
conjugate gradient	rotational (R)	6	9	10	10

For the sake of direct comparison, the resulting stress state for a contrast of $K = 10^3$ when using the continuous gradient with (a) the *basic* scheme and (b) the *accelerated* scheme is plotted in figure G.7 after 100 iterations. The difference between the convergence rates of both schemes distinctively appears: while the stress obtained with the *accelerated* scheme strongly resembles its state at convergence (see figure G.6(a)), that provided by the *basic* scheme, particularly in the inclusion domain, is far from providing a satisfactory approximation and exhibits very high gradients between neighboring pixels. In contrast, convergence is practically established at this stage with the *conjugate-gradient* method (107 iterations are required). In addition to its superior efficiency, the *conjugate-gradient* method is also shown to be nearly insensitive to the choice of the reference medium. Note that the biconjugate gradient stabilized method has been tested in this work, and, although it allows for a decrease in the number of iterations with respect to the standard conjugate gradient technique, it eventually leads to an overall increase in the computational cost (four FFTs need to be performed per iteration when only two are required in the standard conjugate



(a) basic scheme (b) accelerated scheme

Figure G.7: Effect of the different iterative schemes on the convergence of the σ_{yz} stress component produced by a square inclusion using isotropic elasticity for a resolution of 512×512 pixels. The results presented here are those obtained after 100 iterations when using the continuous gradient operator (3.89) with (a) the *basic* scheme (3.74) and (b) the *accelerated* scheme (3.77).

gradient). Further, it is interesting to notice that, although the convergence criterion (G.3) is directly linked to the size of the grid, the resolution does not have a great impact on the convergence rate. For instance, for a resolution of 512×512 pixels, the number of iterations to convergence is seen to increase by less than 10% on average with respect to a 128×128 grid, while the number of pixels is increased by a factor 16.

As another important factor, the precision ϵ^{tol} defined in equation (G.3) plays a crucial role in controlling the accuracy of the results and governs the number of iterations to reach convergence. For instance, the effect of the precision has been tested for a contrast $K = 10$ and a grid of 128×128 pixels, for which the results are reported in table G.2 for $\epsilon^{tol} = 10^{-4}$ and $\epsilon^{tol} = 10^{-10}$. As expected, a lower tolerance leads to a decrease in the convergence rate for all schemes, and, given that the schemes are mathematically proven to be convergent for elastic behaviors and finite contrasts [168], to a better accuracy in the response. Therefore, as with any iterative method, the efficiency of the heterogeneous approach resides in finding a compromise between the accuracy and the computational cost. Nevertheless, it must be noticed that from

Table G.2: Effect of the precision ϵ^{tol} on the number of iterations to convergence for a contrast $K = 10$ between the matrix and the upper-right inclusion and for a resolution of 128×128 pixels. The results are reported for two values $\epsilon^{tol} = 10^{-4}$ and $\epsilon^{tol} = 10^{-10}$ of the precision defined in (G.3).

Iterative scheme	Gradient operator	$\epsilon^{tol} = 10^{-4}$	$\epsilon^{tol} = 10^{-10}$
basic	continuous	34	103
accelerated	continuous	17	52
conjugate gradient	continuous	14	41
basic	centered (C)	30	82
accelerated	centered (C)	17	39
conjugate gradient	centered (C)	15	32
basic	backward (W)	31	86
accelerated	backward (W)	17	43
conjugate gradient	backward (W)	14	33
basic	rotational (R)	25	48
accelerated	rotational (R)	17	34
conjugate gradient	rotational (R)	9	17

a numerical standpoint, choosing a very low tolerance may lead the different schemes never to reach convergence. As a matter of fact, the number of iterations reported in this section are those obtained when using the FFTW library, but it has been observed that the use of others library or FFT algorithms produces different results. Specifically, convergence cannot be reached for low tolerances with certain libraries, as the convergence measure (G.3) starts fluctuating around an asymptotic value in some cases. For this reason, a conservative value of $\epsilon^{tol} = 10^{-4}$ is chosen for the DDD-FFT approach.

In the following, the case of a porous inclusion $K \leq 1$ is investigated. To this end, the classical problem of a hole in a plate depicted in figure G.8(a) is examined. Although a direct comparison with the analytical solution for this simple problem cannot be made – here it must be recalled that the 2D setting neither corresponds to a plane-stress or plane-strain situation and that PBC are to be accounted for –, the resulting stress profile still provides a qualitative appreciation of the validity of the method. In the present setting, the hole of diameter $d = L/4$ is placed at the center

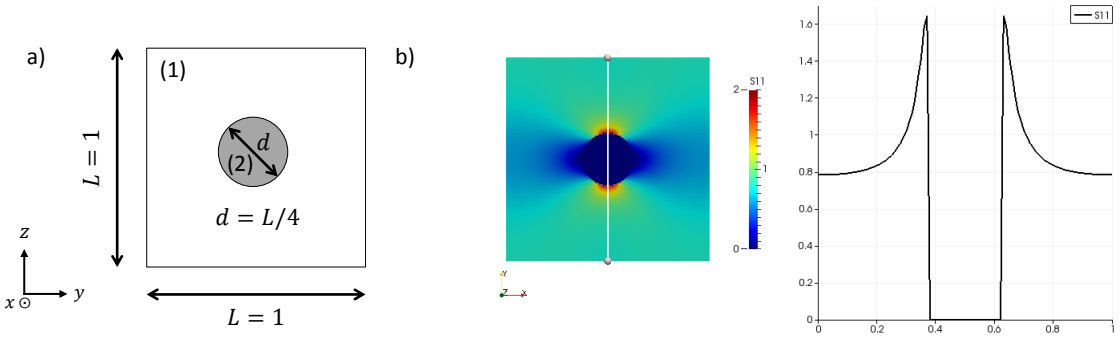


Figure G.8: Setting used for investigating the solution of a porous inclusion $K \leq 1$. (a) A circular inclusion (2) of diameter $d = L/4$ is embedded at the center of a rectangular plate (1) of side $L = 1$. (b) Results for the σ_{yy} stress component when the infinite plate is subjected to an uniaxial tension $E_{yy} = 0.5$ and for a contrast $K = 10^{-3}$. On the right, the σ_{yy} component is plotted along the vertical white z-line delineated on the left figure.

of the plate of size $L \times L$, and the contrast is set to $K = 10^{-3}$ to simulate the hole as a porous inclusion. The resulting σ_{yy} stress component obtained under an uniaxial average tension $E_{yy} = 0.5$ imposed on the plate is plotted in figure G.8(b). At a first glance, the resulting stress distribution is very similar to that associated with a hole in a plate. More precisely, taking a look at the evolution of the σ_{yy} along the vertical z-line passing through the middle at the plate reveals a perfect qualitative agreement with the analytical solution of a hole in a plate, with a maximum stress concentration at the edge of the inclusion, and where the stress vanishes inside the inclusion. Although the maximum stress concentration factor cannot be directly compared with that of the analytical solution for the reasons mentioned earlier – a value of approximately 2 is obtained here –, the excellent qualitative agreement validates the current elastic heterogeneous approach and implementation for porous inclusions and voids.

At this stage, the validity of the DDD-FFT approach to treat dislocations in homogeneous elasticity has been assessed (see Section 3.4.2) and the effect of the gradient operators on the resulting stress field has been investigated in Appendix G.1.2. Further, the validity of the solution of elastic problems in heterogeneous media has

been assessed and the impact of the discrete and iterative schemes on the convergence rate has been investigated in this section. Consequently, by virtue of the superposition principle – the plastic strain distribution is considered as a constant input of the elastic Lippmann-Schwinger problem in the heterogeneous DDD-FFT formulation, see equation (3.73) –, independent validations of the homogeneous elasto-plastic and heterogeneous elastic formulations demonstrate the validity of the heterogeneous elasto-plastic DDD-FFT method. Application of the heterogeneous DDD-FFT approach when accounting for the presence of dislocations in heterogeneous elastic media will be presented Section 5.2 when investigating the interactions between dislocations and second-phase particles.

G.3 Analytical regularization implementation

G.3.1 Intersection area calculation using Green's theorem

As described in Section 3.6.1, a new analytical regularization procedure is introduced in this work to distribute the plastic shear produced by the glide of a dislocation segment. As illustrated in figure 3.10, the new procedure requires the calculation of the area $dS_{ij}^{\vec{\boldsymbol{p}}}$ defined by the intersection between the quadrilateral sheared area produced by the glide of dislocation segment ij and the sphere $d\phi(\vec{\boldsymbol{p}})$ of radius $h/2$ centered at grid point $\vec{\boldsymbol{p}}$. From a purely geometrical perspective, the intersection between a sphere and a quadrilateral reduces to an in-plane intersection between a circle – resulting from the intersection between the sphere and the plane containing the quadrilateral – and the quadrilateral itself. As result, the contour of such intersection is exclusively composed of a succession of straight segments and arcs, as depicted in figure 3.10. Following that, $dS_{ij}^{\vec{\boldsymbol{p}}}$ can be analytically computed by line integration using Green's theorem that establishes the relation between a curvilinear integral carried out along a simple closed contour $\mathcal{C}$ and the double integral integral on the region $\mathcal{D}$ delimited par $\mathcal{C}$. Thus, if $\mathcal{C}^+$ denotes a positively oriented, piecewise smooth,

simple closed curve in plane (x, y) that delimits region $\mathcal{D}$, and if P and Q are functions of (x, y) that have continuous partial derivatives on $\mathcal{D}$, one has:

$$\oint_{\mathcal{C}^+} Pdx + Qdy = \iint_{\mathcal{D}} \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) dxdy \tag{G.4}$$

Of particular interest, this theorem allows for the area calculation $\mathcal{A}(\mathcal{D})$ of any bounded domain $\mathcal{D}$ defined by $\mathcal{C} = \partial\mathcal{D}$ by choosing P and Q that satisfy $\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} = 1$. For instance, the choice $P(x, y) = -y/2$ and $Q(x, y) = x/2$ verifies the latter condition, and leads to:

$$\mathcal{A}(\mathcal{D}) = \iint_{\mathcal{D}} dxdy = \frac{1}{2} \oint_{\mathcal{C}^+} (-ydx + xdy) \tag{G.5}$$

Consequently, surface $dS_{ij}^{\vec{\boldsymbol{p}}}$ in equation (3.6) can be analytically determined by:

$$dS_{ij}^{\vec{\boldsymbol{p}}} = \frac{1}{2} \oint_{\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}} (-ydx + xdy) \tag{G.6}$$

where $\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}$ denotes the closed contour defined by the intersection between the quadrilateral defined by the motion of segment ij and the sphere $d\phi(\vec{\boldsymbol{p}})$ centered in grid point $\vec{\boldsymbol{p}}$, and x and y are the coordinates spanning the contour in the two-dimensional frame defined in the dislocation glide plane. Notice that equation (G.6) generally holds for any types of dislocation motion since the swept area produced by the latter is expected to generate non-intersecting quadrilaterals, thereby generating *simple* closed contours $\mathcal{C}$. Here the notion of *simplicity* refers to the absence of self-intersection. However, in the case of purely rotational dislocation motion, self-intersecting swept areas may be generated: in this case, Green's theorem cannot be applied, but the resulting cross-quadrilateral can be decomposed into two connected triangles, whose intersection areas with sphere $d\phi(\vec{\boldsymbol{p}})$ can be independently computed using (G.6). Importantly, as stated in equation (G.4), the contour $\mathcal{C}_{ij}^{\vec{\boldsymbol{p}}}$ must be oriented, i.e. the

sequence of each portion of the contour delimiting the intersection area must be carefully determined and consistently oriented when travelling along the closed curve. Further, the area is obtained when the contour is positively oriented. This is because when the contour is negatively oriented, the resulting area is negative. Practically, to avoid dealing with the difficulty associated with contour orientation, the absolute value of equation (G.6) is used.

For a closed contour formed of n successive straight segments and arcs, expression (G.6) can be further decomposed as the summation of individual line integrals (see figure 3.10(b)):

$$dS_{ij}^{\vec{p}} = \left| \sum_{k=1}^{n} \frac{1}{2} \oint_{\mathcal{C}_k} (-ydx + xdy) \right| = \left| \sum_{k=1}^{n} I_{\mathcal{C}_k} \right| \tag{G.7}$$

where $\{\mathcal{C}_k\}_{k=1,n}$ denotes the piecewise continuous set of individual curves defining the entire contour $\mathcal{C}_{ij}^{\vec{p}}$. As mentioned earlier, the contour delimiting the intersection of a circle and a quadrilateral (or a triangle for self-intersecting swept areas) is necessarily formed of a set of straight segments and arcs. Therefore, $I_{\mathcal{C}_k}$ can be computed analytically. The easiest way to evaluate the line integral in (G.7) probably lies in rewriting equation (G.4) in vector field notation such that:

$$\int_{\mathcal{C}_k} Pdx + Qdy = \int_{\mathcal{C}_k} \vec{\boldsymbol{F}} \cdot d\vec{\boldsymbol{s}} \tag{G.8}$$

where $\vec{\boldsymbol{F}} = (P, Q)$ and $d\vec{\boldsymbol{s}} = (dx, dy)$ is the differential field, such that upon parametrization of line contour $\mathcal{C}_k$ using vector function $\vec{\boldsymbol{r}}(t)$ with parameter t ranging from t_0 to t_1, one obtains:

$$\int_{\mathcal{C}_k} \vec{\boldsymbol{F}} \cdot d\vec{\boldsymbol{s}} = \int_{t_0}^{t_1} \vec{\boldsymbol{F}}\left(\vec{\boldsymbol{r}}(t)\right) \cdot \vec{\boldsymbol{r}'}(t) dt \tag{G.9}$$

where vector function $\vec{\boldsymbol{r}'}(t)$ denotes the derivative of function $\vec{\boldsymbol{r}}(t)$ with respect to parameter t. Following this, it appears convenient to parametrize any straight segment

$\mathcal{C}_k$ defined between vertices (x_0, y_0) and (x_1, y_1) with the following vector function:

$$\begin{cases} \vec{\boldsymbol{r}}(t) = \big((x_1 - x_0)t + x_0, (y_1 - y_0)t + y_0\big) \\ \vec{\boldsymbol{r}'}(t) = \big(x_1 - x_0, y_1 - y_0\big) \end{cases} \qquad 0 \leq t \leq 1 \tag{G.10}$$

Consequently, the line integral $I_{\mathcal{C}_k}^{seg}$ along this segment can be analytically derived from equation (G.7) as:

$$\begin{aligned} I_{\mathcal{C}_k}^{seg} &= \frac{1}{2}\int_{\mathcal{C}_k} (-ydx + xdy) = \int_0^1 \vec{\boldsymbol{F}}\left(\vec{\boldsymbol{r}}(t)\right) \cdot \vec{\boldsymbol{r}'}(t)dt \\ &= \frac{1}{2}\int_0^1 \left(-(y_1 - y_0)t - y_0, (x_1 - x_0)t + x_0\right) \cdot (x_1 - x_0, y_1 - y_0)\, dt \\ &= \frac{1}{2}\int_0^1 [-(y_1 - y_0)(x_1 - x_0)t - y_0(x_1 - x_0) \\ &\qquad\qquad + (x_1 - x_0)(y_1 - y_0)t + x_0(y_1 - y_0)]\, dt \\ &= \frac{1}{2}\left(-y_0(x_1 - x_0) + x_0(y_1 - y_0)\right) \\ &= \frac{1}{2}\left(x_0 y_1 - y_0 x_1\right) \end{aligned} \tag{G.11}$$

where $\vec{\boldsymbol{F}} = (-y/2, x/2)$. Similarly, for any arc $\mathcal{C}_k$ defined as a portion of a circle of radius r centered in (x_c, y_c) and delimited by end vertices (x_0, y_0) at angle θ_0 and (x_1, y_1) at angle θ_1 (assuming $\theta_1 > \theta_0$), the following parametric representation can be used:

$$\begin{cases} \vec{\boldsymbol{r}}(t) = \big(r\cos t + x_c, r\sin t + y_c\big) \\ \vec{\boldsymbol{r}'}(t) = \big(-r\sin t, r\cos t\big) \end{cases} \qquad \theta_0 \leq t \leq \theta_1 \tag{G.12}$$

With this, the line integral $I_{\mathcal{C}_k}^{arc}$ along this arc can be analytically derived from equation (G.7) as:

$$
\begin{aligned}
I_{\mathcal{C}_k}^{arc} &= \frac{1}{2}\int_{\mathcal{C}_k}(-ydx+xdy) = \int_{\theta_0}^{\theta_1}\vec{\boldsymbol{F}}\left(\vec{\boldsymbol{r}}(t)\right)\cdot\vec{\boldsymbol{r}}'(t)dt \\
&= \frac{1}{2}\int_{\theta_0}^{\theta_1}(-r\sin t - y_c, r\cos t + x_c)\cdot(-r\sin t, r\cos t)\,dt \\
&= \frac{1}{2}\int_{\theta_0}^{\theta_1}\left[r^2\sin^2 t + y_c r\sin t + r^2\cos^2 t + x_c r\cos t\right]dt \\
&= \frac{1}{2}\left[r^2(\theta_1-\theta_0) + x_c r(\sin\theta_1 - \sin\theta_0) - y_c r(\cos\theta_1 - \cos\theta_0)\right] \\
&= \frac{1}{2}\left[r^2(\theta_1-\theta_0) + x_c(r\sin\theta_1 + y_c - r\sin\theta_0 - y_c) - y_c(r\cos\theta_1 + x_c - r\cos\theta_0 - x_c)\right] \\
&= \frac{1}{2}\left[r^2(\theta_1-\theta_0) + x_c(y_1-y_0) - y_c(x_1-x_0)\right] \qquad \text{(G.13)}
\end{aligned}
$$

Thus, expressions (G.11) and (G.13) provide fully analytical solutions to compute the intersection area $dS_{ij}^{\vec{\boldsymbol{p}}}$ with equation (G.7), provided the individual pieces $\{\mathcal{C}_k\}_{k=1,n}$ forming the contour $\mathcal{C}$ bounding region $dS_{ij}^{\vec{\boldsymbol{p}}}$ are determined. The numerical algorithm developed in this work for the determination of such contour is presented in Appendix G.3.2

G.3.2 Numerical construction of the contour of the intersection area

The full contour $\mathcal{C}$ of the intersection area $dS_{ij}^{\vec{\boldsymbol{p}}}$ described in Appendix G.3.1 is constructed by ordering and joining the quadrilateral vertices and the circle-quadrilateral intersection points through straight segments and arcs (see figure G.9(a)). Note that this contour can be directly obtained by using computational geometry libraries such as CGAL [42]. In the present work however, the following numerical algorithm is implemented.

Let us consider a dislocation segment ij defined between end nodes i and j moving from their initial positions $\vec{\boldsymbol{x}}_i$ and $\vec{\boldsymbol{x}}_j$ at velocities $\vec{\boldsymbol{v}}_i$ and $\vec{\boldsymbol{v}}_j$ during time dt, respectively, and gliding on plane $\mathcal{P}$ with unit normal $\vec{\boldsymbol{n}}$. First, the sheared area produced by the glide of the segment forms a quadrilateral $\mathcal{Q}$ defined by vertices $\{\vec{\boldsymbol{P}}_1, \vec{\boldsymbol{P}}_2, \vec{\boldsymbol{P}}_3, \vec{\boldsymbol{P}}_4\} = \{\vec{\boldsymbol{x}}_i, \vec{\boldsymbol{x}}_j, \vec{\boldsymbol{x}}_j + \vec{\boldsymbol{v}}_j dt, \vec{\boldsymbol{x}}_i + \vec{\boldsymbol{v}}_i dt\}$ (see figure 3.10(a)). Second, the

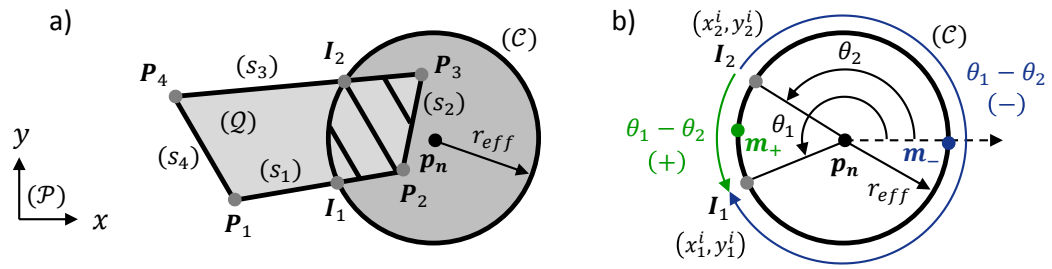


Figure G.9: Schematic of the intersection between quadrilateral $\mathcal{Q}$ defined by the sheared area produced by the glide of a dislocation segment on a plane $\mathcal{P}$ and circle $\mathcal{C}$ of radius r_{eff} resulting from the intersection between the sphere $d\phi(\vec{\boldsymbol{p}})$ and plane $\mathcal{P}$. (a) The intersection area $dS_{ij}^{\vec{\boldsymbol{p}}} = \text{area}(\mathcal{Q} \cap \mathcal{C})$ (shaded region) can be analytically calculated using Green's theorem (G.7) by following the oriented contour composed of the straight segments $\overline{\boldsymbol{I}_1\boldsymbol{P}_2}$, $\overline{\boldsymbol{P}_2\boldsymbol{P}_3}$ and $\overline{\boldsymbol{P}_3\boldsymbol{I}_2}$ and the arc $\widehat{\boldsymbol{I}_2\boldsymbol{I}_1}$ where $\{\vec{\boldsymbol{P}}_j\}_{j=1,4}$ denotes the vertices of $\mathcal{Q}$ and $\{\vec{\boldsymbol{I}}_k\}_{k=1,n}$ the n intersection between edges of $\mathcal{Q}$ and $\mathcal{C}$ (here $n = 2$). (b) Arc $\widehat{\boldsymbol{I}_2\boldsymbol{I}_1}$ can possibly be travelled along the positive $(+)$ or the negative $(-)$ direction. When $n = 2$, the direction along which it must be travelled can be determined as that whose middle point, respectively $\vec{\boldsymbol{m}}_+$ and $\vec{\boldsymbol{m}}_-$, lies within quadrilateral $\mathcal{Q}$. In the present case, $\vec{\boldsymbol{m}}_+ \in \mathcal{Q}$ and the arc should be travelled in the positive $(+)$ direction.

intersection between the sphere $d\phi(\vec{\boldsymbol{p}})$ and the glide plane $\mathcal{P}$ results in a circle $\mathcal{C}$ of radius $r_{eff} = \sqrt{h^2/4 - d^2}$ centered at $\vec{\boldsymbol{p}}_n$, where $\vec{\boldsymbol{p}}_n$ is the orthogonal projection of $\vec{\boldsymbol{p}}$ onto plane $\mathcal{P}$ such that $\vec{\boldsymbol{p}} - \vec{\boldsymbol{p}}_n = d\vec{\boldsymbol{n}}$. Therefore, as depicted in figure G.9, the intersection area $dS_{ij}^{\vec{\boldsymbol{p}}}$ (shaded region) is defined by the area of the region $\mathcal{Q} \cap \mathcal{C}$.

As mentioned earlier, the contour of $\mathcal{Q} \cap \mathcal{C}$ can be solely composed of straight segments and arcs. Numerically, its determination can be achieved using the method described hereafter. First notice that $(\mathcal{Q} \cap \mathcal{C}) \in \mathcal{P}$ such that the problem can be conveniently solved in the two-dimensional space. Let us select a frame $(\mathcal{O}, x, y)$ on plane $\mathcal{P}$ in which the coordinates of vertices $\{\vec{\boldsymbol{P}}_j\}_{j=1,4}$ are given by (x_j, y_j). The first step to determine the contour of $\mathcal{Q} \cap \mathcal{C}$ consists in sequentially finding the intersections points $\{\vec{\boldsymbol{I}}_k\}_{k=1,n}$ of coordinates (x_j^i, y_j^i) between the oriented edges $\{s_j\}_{j=1,4}$ of $\mathcal{Q}$ and the circle $\mathcal{C}$, where s_j denotes the segment defined between points $\vec{\boldsymbol{P}}_j$ and $\vec{\boldsymbol{P}}_{j+1}$. Since each edge s_j may either not intersect with circle $\mathcal{C}$, intersect in one point,

or intersect in two points, the total number of intersections n can take values of $0 \leq n \leq 8$. Region $\mathcal{Q} \cap \mathcal{C}$ is then delimited by sequentially joining the intersection points $\{\vec{\boldsymbol{I}}_k\}_{k=1,n}$ and the quadrilateral vertices $\{\vec{\boldsymbol{P}}_j\}_{j=1,4}$ through straight segments and arcs while travelling the entire contour in one given direction. With that, area $dS_{ij}^{\vec{\boldsymbol{p}}}$ is computed using expression (G.7), in which straight segments and arcs line integrals are calculated using equations (G.11) and (G.13), respectively.

The only remaining difficulty lies in the determination of the angle $\theta_j - \theta_i$ (see equation (G.13)) in the case of arc portions $\widehat{\boldsymbol{I}_i\boldsymbol{I}_j}$, as a circle can always be travelled along two different paths from coordinates (x_j^i, y_j^i) to (x_i^k, y_i^k), namely along the positive $(+)$ or negative $(-)$ direction, i.e. anti-clockwise or clockwise. In other words, contrary to a segment, the knowledge of the entry point on an arc supported by a circle does not determine its orientation. This difficulty can easily be tackled in the case where more than two intersections are detected (i.e. for $n > 2$): in that case, the arc defined between two intersection points should be travelled along the path that does not include any other intersection point $\vec{\boldsymbol{I}}_k$. If the only detected intersection points are those defining the arc, the middle point of each possible path can be tested to choose the path for which the middle point lies inside the quadrilateral $\mathcal{Q}$ (see figure G.9(b)). The latter condition can be checked using a Point-In-Polygon (PIP) algorithm based on the ray casting approach. Note that in the case where no intersection between the circle and the quadrilateral are found (i.e. for $n = 0$), either the circle is entirely comprised into the quadrilateral, or the quadrilateral is fully comprised into the circle, or both are well separated. Such configurations can be easily identified, especially via the use of bounding boxes and PIP techniques.

G.4 GPU-accelerated implementation

Traditionally, the use of graphics processing units (GPUs) was dedicated to numerical calculations related to computer graphics, as way to supplement the primary computational capacities of the CPUs. Primary applications include effect rendering in video and image processing. However, in the recent years, general-purpose computing on graphics processing units (GP-GPU) has developed out to scientific computing in various domains such as bioinformatics and molecular biology.

The main advantage of GPU computing is to benefit from the massively parallel architecture offered by the profusion of interconnected processing units in GPUs. Further, the development of dedicated interfaces and libraries for mathematical calculations such as CUDA provides a convenient solution to accelerate up to orders of magnitudes any sort of parallelizable computations traditionally handled by CPUs.

In this thesis, a GPU-accelerated version of the DDD-FFT code is developed. Although several stages of the DDD cycle might benefit from a GPU implementation, the focus is placed here on developing a GPU implementation of the FFT-based solver. This choice is dictated by the observation that, in the DDD-FFT approach, most of the computation time is dedicated to the evaluation of the stress state via the FFT-based approach. Other stages of the cycle such as local dislocation-dislocation calculations could as well benefit from a GPU implementation following the procedure developed in [83] – although their overall computational cost is expected to remain limited when the resolution is chosen adequately –, but, by lack of time, their implementation would not be considered in this work.

G.4.1 General principles and constraints

The general principle for GPU computing through the use of the CUDA framework is as follows. First, the fundamental processing flow describing the interaction between the host (CPU) and the device (GPU) can be summarized by the 4 following steps:

(1) Copy the main memory to the GPU memory

(2) CPU instructs the process to GPU

(3) GPU executes the process in parallel in each thread

(4) Copy the GPU results back to the main memory

The main difficulty of the GPU implementation lies in the development and design of the computational processes to be performed on the GPU, i.e. step (3) in the above. The efficiency of GPU-accelerated calculations essentially stems from a fully parallel program architecture, which practically involves the implementation of kernels. A kernel is the equivalent of a function on the CPU, and for the sake of simplicity, can be regarded as a set of instructions that will be executed on each thread. The total number of threads available defines the performance of the GPU device (it corresponds to the total number of logical cores) and corresponds to the maximum simultaneous parallel calculations the GPU can treat. Threads are further regrouped into blocks, such that a thread corresponds to the execution of a kernel with a given index.

The design and implementation of kernels are subjected to specific constraints. First, the kernels can only access the memory of the GPU. Therefore, when performing calculations on arrays originally defined and populated on the host CPU, these must be copied to the GPU memory in order to be accessible for the kernels. As CUDA features a fully shared memory, data will be accessible to every thread. Nonetheless, when working with large arrays, the sole cost associated with their copies can in certain cases exceed the total computational cost, and can therefore significantly impact the performance. Although it can be alleviated by using asynchronous memory transfers, the cost of the copies, i.e. steps (1) and (4), should not be overlooked in the evaluation of the overall speed up that can be achieved.

Further, the number of threads per block should be a multiple of the warp size (32 threads). Although each group can execute different branch of a program, the performance is not expected to be impacted as long as the load remains balanced

across each group. However, the use of divergent paths within a group significantly affects the performance, and should be avoided. For this reason, the usage of *if-else* statements (or any other divergent instructions) is usually prohibited in kernels, or should be coupled with templated kernels dedicated to specific groups. This induces strong algorithmic constraints so as to ensure that each specific branch of a program is designed to be executed by a minimum of a multiple of 32 parallel threads.

Finally, other parameters and procedures can greatly influence the performance of GPU computations (synchronization, double-precision, loop-unrolling, ...) but are not detailed here. For more information on GPU implementation using CUDA, the reader is referred to the CUDA documentation [186].

G.4.2 GPU-FFT-solver implementation

The development and implementation of a GPU-accelerated version of the DDD-FFT code developed in this thesis comprises two principal components. The first component pertains to the development of an interface between the original Fortran code and the CUDA kernels, and the second to the implementation of the kernels to perform the calculations associated with the FFT-based solver in a parallel fashion.

As mentioned in Chapter 2, the DDD code developed in this thesis and on which the DDD-FFT approach is inserted is written in Fortran 90. This host program, running on the CPU, will call the GPU kernels to execute the GPU-FFT-solver. Alternatively, the GPU kernels need to be programmed using CUDA. Practically, CUDA instructions are coded using CUDA C++, that can be regarded as an overloaded version of the C++ language featuring specific additional instructions, and that need to be compiled with the dedicated CUDA C++ compiler (*nvcc*). As a result, an interface between Fortran and C++ is developed to bind the host program to the GPU kernels.

The general algorithm describing the main stages of the homogeneous DDD-FFT

cycle is presented in figure 3.14. The use of a GPU-FFT-solver solely consists in performing the sequence of operations in step (iv) in figure 3.14 on the GPU device. In the heterogeneous formulation, the operations of step (iv) are replaced by those given in figures 3.15, 3.16 or 3.17 depending on the numerical scheme. Essentially for both the homogeneous and heterogeneous approaches, the computational operations to be performed involve 5 elementary steps, namely (a) the calculation of the polarization tensor, (b) the direct FFT, (c) the calculation of the strain in the Fourier space, (d) the inverse FFT, and (e) the calculation of the stress state. For the heterogeneous formulation, the main difference lies in that those 5 steps are repeatedly calculated in an outer iterative loop, so that both formulations can share the same implementation.

Devising a parallel strategy for the kernels of the GPU-FFT-solver is a straightforward task. This is because, whether in the real space or in the Fourier domain, the behaviors associated with each voxel are independent from a voxel to another. Thus, in the real space, the calculation of the polarization tensor (a) and that of the stress state (e) at a given voxel do not require the knowledge of the state at other voxels. Similarly, in the Fourier domain, the calculation of the Green's function at a given frequency is independent of any other frequencies. Mathematically, this results from the orthogonality of the Fourier decomposition, and the interactions between voxels solely pertain to the FFT calculations. As a result, the kernels associated with elementary steps (a), (c) and (e) can be conveniently implemented such as to process one voxel per thread. This way, it can be further ensured that these kernels do not contain any divergent path.

The computation of the FFTs (direct FFT of the polarization tensor (b) and inverse FFT of the strain tensor (d), see figure 3.14) is performed via the use of the CUDA native *cufft* library. The *cufft* library is a pre-built kernel that computes the complex FFT of any complex input data. As done in the CPU implementation, the computation of the FFT of a tensor field is achieved by independently computing the

FFT on each component, i.e. 6 FFTs for a second-order symmetric tensor field.

Following the above, the solution employed in this work to handle memory in the most efficient way and avoid costly unnecessary copies is to fragment and copy the real-valued array $\{\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ of size $6N_{vox}$ stored on the CPU into 6 real-valued arrays of size N_{vox} in the GPU memory, each of them storing a different tensor component. After the plastic strain array is copied to the GPU at each simulation step, the computation of the steps (a) to (e) is performed as indicated in the above. During these computations, intermediate tensor fields are sequentially stored in 6 complex-valued arrays of size N_{vox} allocated on the GPU. After step (e), the resulting stress distribution $\{\boldsymbol{\sigma}(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$ is transferred back to the CPU memory.

Apart from the plastic strain distribution $\{\boldsymbol{\epsilon}^p(\vec{\boldsymbol{x}}_d)\}_{d=1,N_{vox}}$, the calculations in the 5 elementary steps (a)–(e) require the knowledge of the elastic stiffness tensors $\boldsymbol{C}$ of every phase. Such data, independent of the microstructure evolution, can be copied to the GPU memory once at the beginning of each simulation.

In the case of the heterogeneous formulation, a supplementary convergence test needs to be performed to determine whether the computation of the 5 elementary steps needs to be repeated in a subsequent iteration to reach convergence (e.g. see figure 3.15). As defined in equation (3.81), the convergence criterion, that consists in evaluating the norm of the change in the strain tensor between two consecutive iterations, is efficiently computed using the *reduce* kernel of the CUDA *thrust* library. The same library is used to performed vector operations in the case of the *conjugate-gradient* scheme (see figure 3.17).

Although GPU implementations generally allow for substantial computational accelerations, their performance remains dictated by the specifications the GPU device(s) that are used. Further, as mentioned earlier, the cost of the memory copy is generally significant; therefore, it is expected that the GPU-FFT-solver will provide larger speed-up in the heterogeneous formulation, in which case the overall copy time

accounts for a smaller portion of the total calculations. Finally, it is to be noted that, extending Amdahl's law [3], the overall gain allowed by GPU-implementation is limited by the fraction of the program executed on GPU. Performance of the GPU-accelerated version of the DDD-FFT code is assessed in Section 3.7.2.

REFERENCES

[1] Agnew, S. R. and Duygulu, O., "Plastic anisotropy and the role of non-basal slip in magnesium alloy az31b," *International Journal of plasticity*, vol. 21, no. 6, pp. 1161–1193, 2005.

[2] Akarapu, S. and Zbib, H. M., "Line-integral solution for the stress and displacement fields of an arbitrary dislocation segment in isotropic bi-materials in 3d space," *Philosophical Magazine*, vol. 89, no. 25, pp. 2149–2166, 2009.

[3] Amdahl, G. M., "Validity of the single processor approach to achieving large scale computing capabilities," in *Proceedings of the April 18-20, 1967, spring joint computer conference*, pp. 483–485, ACM, 1967.

[4] Argon, A. S., "A statistical theory for easy glide ii," in *Physics of Strength and Plasticity*, p. 217–244, Cambrige, MA: MIT Press, 1969.

[5] Arsenlis, A., Cai, W., Tang, M., Rhee, M., Oppelstrup, T., Hommes, G., Pierce, T. G., and Bulatov, V. V., "Enabling strain hardening simulations with dislocation dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 15, no. 6, pp. 553–595, 2007.

[6] Arsenlis, A. and Parks, D. M., "Crystallographic aspects of geometrically-necessary and statistically-stored dislocation density," *Acta Materialia*, vol. 47, no. 5, pp. 1597–1611, 1999.

[7] Arsenlis, A. and Parks, D. M., "Modeling the evolution of crystallographic dislocation density in crystal plasticity," *Journal of the Mechanics and Physics of Solids*, vol. 50, no. 9, pp. 1979–2009, 2002.

[8] Arsenlis, A., Rhee, M., Hommes, G., Cook, R., and Marian, J., "A dislocation dynamics study of the transition from homogeneous to heterogeneous deformation in irradiated body-centered cubic iron," *Acta Materialia*, vol. 60, no. 9, pp. 3748–3757, 2012.

[9] Asaro, R. J. and Rice, J. R., "Strain localization in ductile single-crystals," *Journal of the Mechanics and Physics of Solids*, vol. 25, no. 5, pp. 309–338, 1977.

[10] Ashby, M. F., "Deformation of plastically non-homogeneous materials," *Philosophical Magazine*, vol. 21, no. 170, pp. 399–424, 1970.

[11] Ashmawi, W. M. and Zikry, M. A., "Prediction of grain-boundary interfacial mechanisms in polycrystalline materials," *Journal of Engineering Materials and Technology*, vol. 124, no. 1, pp. 88–96, 2001.

[12] Aubry, S. and Arsenlis, A., "Use of spherical harmonics for dislocation dynamics in anisotropic elastic media," *Modelling and Simulation in Materials Science and Engineering*, vol. 21, no. 6, 2013.

[13] Aubry, S., Fitzgerald, S. P., and Arsenlis, A., "Methods to compute dislocation line tension energy and force in anisotropic elasticity," *Modelling and Simulation in Materials Science and Engineering*, vol. 22, no. 1, 2014.

[14] Bacon, D. J., Barnett, D. M., and Scattergood, R. O., "Anisotropic continuum theory of lattice-defects," *Progress in Materials Science*, vol. 23, no. 2-4, pp. 51–262, 1978.

[15] Bacon, D. J., Kocks, U. F., and Scattergood, R., "Effet of dislocation self-interaction on orowan stress," *Philosophical Magazine*, vol. 28, no. 6, pp. 1241–1263, 1973.

[16] Bacon, D. J. and Vitek, V., "Atomic-scale modeling of dislocations and related properties in the hexagonal-close-packed metals," *Metallurgical and Materials Transactions a-Physical Metallurgy and Materials Science*, vol. 33, no. 3, pp. 721–733, 2002.

[17] Barnett, D. M., "The precise evaluation of derivatives of the anisotropic elastic green's functions," *Physica Status Solidi (b)*, vol. 49, no. 2, pp. 741–748, 1972.

[18] Basinski, Z. S., "Thermally activated glide in face-centred cubic metals and its application to the theory of strain hardening," *Philosophical Magazine*, vol. 4, no. 40, pp. 393–432, 1959.

[19] Bassani, J. L. and Wu, T.-Y., "Latent hardening in single crystals ii. analytical characterization and predictions," *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 435, no. 1893, pp. 21–41, 1991.

[20] Berbenni, S., Taupin, V., Djaka, K. S., and Fressengeas, C., "A numerical spectral approach for solving elasto-static field dislocation and g-disclination mechanics," *International Journal of Solids and Structures*, vol. 51, no. 23-24, pp. 4157–4175, 2014.

[21] Bertin, N., Capolungo, L., and Beyerlein, I. J., "Hybrid dislocation dynamics based strain hardening constitutive model," *International Journal of Plasticity*, vol. 49, no. 0, pp. 119–144, 2013.

[22] Bertin, N., Tome, C. N., Beyerlein, I. J., Barnett, M. R., and Capolungo, L., "On the strength of dislocation interactions and their effect on latent hardening in pure magnesium," *International Journal of Plasticity*, vol. 62, pp. 72–92, 2014.

[23] Bertin, N., Upadhyay, M. V., Pradalier, C., and Capolungo, L., "A fft-based formulation for efficient mechanical fields computation in isotropic and anisotropic periodic discrete dislocation dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 23, no. 6, p. 065009, 2015.

[24] Beyerlein, I. J., McCabe, R. J., and Tomé, C. N., "Stochastic processes of 10-12 deformation twinning in hexagonal close-packed polycrystalline zirconium and magnesium," *International Journal for Multiscale Computational Engineering*, vol. 9, no. 4, pp. 459–480, 2011.

[25] Beyerlein, I. J. and Tomé, C. N., "A dislocation-based constitutive law for pure zr including temperature effects," *International Journal of Plasticity*, vol. 24, no. 5, pp. 867–895, 2008.

[26] Brenner, R., Beaudoin, A. J., Suquet, P., and Acharya, A., "Numerical implementation of static field dislocation mechanics theory for periodic media," *Philosophical Magazine*, vol. 94, no. 16, pp. 1764–1787, 2014.

[27] Brisard, S. and Dormieux, L., "Fft-based methods for the mechanics of composites: A general variational framework," *Computational Materials Science*, vol. 49, no. 3, pp. 663–671, 2010.

[28] Brisard, S. and Dormieux, L., "Combining galerkin approximation techniques with the principle of hashin and shtrikman to derive a new fft-based numerical method for the homogenization of composites," *Computer Methods in Applied Mechanics and Engineering*, vol. 217, pp. 197–212, 2012.

[29] Brown, D. W., Beyerlein, I. J., Sisneros, T. A., Clausen, B., and Tome, C. N., "Role of twinning and slip during compressive deformation of beryllium as a function of strain rate," *International Journal of Plasticity*, vol. 29, pp. 120–135, 2012.

[30] Bulatov, V., Abraham, F. F., Kubin, L., Devincre, B., and Yip, S., "Connecting atomistic and mesoscale simulations of crystal plasticity," *Nature*, vol. 391, no. 6668, pp. 669–672, 1998.

[31] Bulatov, V. V., Hsiung, L. L., Tang, M., Arsenlis, A., Bartelt, M. C., Cai, W., Florando, J. N., Hiratani, M., Rhee, M., Hommes, G., Pierce, T. G., and de la Rubia, T. D., "Dislocation multi-junctions and strain hardening," *Nature*, vol. 440, no. 7088, pp. 1174–1178, 2006.

[32] Buroni, F. C. and Saez, A., "Unique and explicit formulas for green's function in three-dimensional anisotropic linear elasticity," *Journal of Applied Mechanics-Transactions of the Asme*, vol. 80, no. 5, 2013.

[33] Busso, E., *Cyclic Deformation of Monocrystalline Nickel Aluminide and High Temperature Coatings*. PhD thesis, Massachusetts Institute of Technology, 1990.

[34] Caceres, C. H. and Lukac, P., "Strain hardening behaviour and the taylor factor of pure magnesium," *Philosophical Magazine*, vol. 88, no. 7, pp. 977–989, 2008.

[35] Cai, W., Arsenlis, A., Weinberger, C. R., and Bulatov, V. V., "A non-singular continuum theory of dislocations," *Journal of the Mechanics and Physics of Solids*, vol. 54, no. 3, pp. 561–587, 2006.

[36] Cai, W. and Bulatov, V. V., "Mobility laws in dislocation dynamics simulations," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 387, pp. 277–281, 2004.

[37] Caillard, D. and Couret, A., "Dislocation movements controlled by friction forces and local pinning in metals and alloys," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 322, no. 1-2, pp. 108–117, 2002.

[38] Cao, F., Beyerlein, I. J., Addessio, F. L., Sencer, B. H., Trujillo, C. P., Cerreta, E. K., and Iii, G. T. G., "Orientation dependence of shock-induced twinning and substructures in a copper bicrystal," *Acta Materialia*, vol. 58, no. 2, pp. 549–559, 2010.

[39] Capolungo, L., "Dislocation junction formation and strength in magnesium," *Acta Materialia*, vol. 59, no. 8, pp. 2909–2917, 2011.

[40] Capolungo, L., Beyerlein, I. J., and Qwang, Z., "The role of elastic anisotropy on plasticity in hcp metals: a three-dimensional dislocation dynamics study," *Modelling and Simulation in Materials Science and Engineering*, vol. 18, no. 8, 2010.

[41] Capolungo, L., Beyerlein, I. J., and Tomé, C. N., "Slip-assisted twin growth in hexagonal close-packed metals," *Scripta Materialia*, vol. 60, no. 1, pp. 32–35, 2009.

[42] CGAL, "Computational geometry algorithms library."

[43] Chassagne, M., Legros, M., and Rodney, D., "Atomic-scale simulation of screw dislocation/coherent twin boundary interaction in al, au, cu and ni," *Acta Materialia*, vol. 59, no. 4, pp. 1456–1463, 2011.

[44] Cheong, K. S. and Busso, E. P., "Discrete dislocation density modelling of single phase fcc polycrystal aggregates," *Acta Materialia*, vol. 52, no. 19, pp. 5665–5675, 2004.

[45] Cheong, K. S., Busso, E. P., and Arsenlis, A., "A study of microstructural length scale effects on the behaviour of fcc polycrystals using strain gradient concepts," *International Journal of Plasticity*, vol. 21, no. 9, pp. 1797–1814, 2005.

[46] Chou, T. W. and Pan, Y. C., "Elastic energies of disclinations in hexagonal crystals," *Journal of Applied Physics*, vol. 44, no. 1, pp. 63–65, 1973.

[47] Christian, J. W. and Mahajan, S., "Deformation twinning," *Progress in Materials Science*, vol. 39, no. 1-2, pp. 1–157, 1995.

[48] Chu, H., Pan, E., Han, X., Wang, J., and Beyerlein, I., "Elastic fields of dislocation loops in three-dimensional anisotropic bimaterials," *Journal of the Mechanics and Physics of Solids*, vol. 60, no. 3, pp. 418–431, 2012.

[49] Clausen, B., Brown, D., Tomé, C., Balogh, L., and Vogel, S., "Engineering related neutron diffraction measurements probing strains, texture and microstructure," in *31st Risø International Symposium on Materials Science, Challenges in materials science and possibilities in 3D and 4D characterization techniques* (N. Hansen, D. Juul Jensen, S. N. H. P. and Ralph, B., eds.), (Risø National Laboratory for Sustainable Energy, Technical University of Denmark), 2010.

[50] Conrad, H. and Robertson, W. *Trans. AIME.*, vol. 209, p. 503–512, 1957.

[51] Cooley, J. W. and Tukey, J. W., "An algorithm for machine calculation of complex fourier series," *Mathematics of Computation*, vol. 19, no. 90, p. 297, 1965.

[52] Cottrell, A. H., "Theory of dislocations," *Progress in Metal Physics*, vol. 4, no. 0, pp. 205–264, 1953.

[53] Cottrell, A., *The Mechanical Properties of Materials*. New York: J. Willey, 1964.

[54] Cuitino, A. M. and Ortiz, M., "Computational modeling of single-crystals," *Modelling and Simulation in Materials Science and Engineering*, vol. 1, no. 3, pp. 225–263, 1993.

[55] Dai, H. and Parks, D. M., *Geometrically-necessary dislocation density in continuum crystal plasticity theory and fem implementation*. Submitted for publication, 1998.

[56] de Koning, M., Kurtz, R. J., Bulatov, V. V., Henager, C. H., Hoagland, R. G., Cai, W., and Nomura, M., "Modeling of dislocation–grain boundary interactions in fcc metals," *Journal of Nuclear Materials*, vol. 323, no. 2–3, pp. 281–289, 2003.

[57] de Koning, M., Miller, R., Bulatov, V. V., and Abraham, F. F., "Modelling grain-boundary resistance in intergranular dislocation slip transmission," *Philosophical Magazine a-Physics of Condensed Matter Structure Defects and Mechanical Properties*, vol. 82, no. 13, pp. 2511–2527, 2002.

[58] de Vaucorbeil, A., Poole, W., and Sinclair, C., "The superposition of strengthening contributions in engineering alloys," *Materials Science and Engineering: A*, vol. 582, pp. 147–154, 2013.

[59] Deng, C. and Sansoz, F., "Enabling ultrahigh plastic flow and work hardening in twinned gold nanowires," *Nano Letters*, vol. 9, no. 4, pp. 1517–1522, 2009.

[60] Deng, J., El-Azab, A., and Larson, B. C., "On the elastic boundary value problem of dislocations in bounded crystals," *Philosophical Magazine*, vol. 88, no. 30-32, pp. 3527–3548, 2008.

[61] Devincre, B., "Meso-scale simulation of the dislocation dynamics," *NATO ASI Series E Applied Sciences-Advanced Study Institute*, vol. 308, pp. 309–324, 1996.

[62] Devincre, B., "Dislocation dynamics simulations of slip systems interactions and forest strengthening in ice single crystal," *Philosophical Magazine*, vol. 93, no. 1-3, pp. 235–246, 2013.

[63] Devincre, B., Hoc, T., and Kubin, L., "Dislocation mean free paths and strain hardening of crystals," *Science*, vol. 320, no. 5884, pp. 1745–1748, 2008.

[64] Devincre, B., Hoc, T., and Kubin, L. P., "Collinear interactions of dislocations and slip systems," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 400, pp. 182–185, 2005.

[65] Devincre, B., Kubin, L., and Hoc, T., "Physical analyses of crystal plasticity by dd simulations," *Scripta Materialia*, vol. 54, no. 5, pp. 741–746, 2006.

[66] Devincre, B., Kubin, L., and Hoc, T., "Collinear superjogs and the low-stress response of fcc crystals," *Scripta Materialia*, vol. 57, no. 10, pp. 905–908, 2007.

[67] Devincre, B. and Kubin, L. P., "Mesoscopic simulations of dislocations and plasticity," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 234, pp. 8–14, 1997.

[68] Devincre, B., Kubin, L. P., Lemarchand, C., and Madec, R., "Mesoscopic simulations of plastic deformation," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 309, pp. 211–219, 2001.

[69] Devincre, B., Madec, R., Monnet, G., Queyreau, S., Gatti, R., and Kubin, L., "Modeling crystal plasticity with dislocation dynamics simulations: The micromegas code," in *Mechanics of Nano-objects* (Thomas, O., P. A. F. S., ed.), p. 81–100, Paris: Presses de l'Ecole des Mines de Paris, 2011.

[70] Devincre, B. and Kubin, L., "Scale transitions in crystal plasticity by dislocation dynamics simulations," *Comptes Rendus Physique*, vol. 11, no. 3-4, pp. 274–284, 2010.

[71] deWit, G. and Koehler, J. S., "Interaction of dislocations with an applied stress in anisotropic crystals," *Physical Review*, vol. 116, no. 5, pp. 1113–1120, 1959.

[72] deWit, R., "The continuum theory of stationary dislocations," *Solid State Physics-Advances in Research and Applications*, vol. 10, pp. 249–292, 1960.

[73] deWit, R., "Thermodynamic force on a dislocation," *Journal of Applied Physics*, vol. 39, no. 1, pp. 137–141, 1968.

[74] deWit, R., "Linear theory of static disclinations," in *Fundamental Aspects of Dislocation Theory* (J.A. Simmons, R. d. and Bullough, R., eds.), pp. 651–680, National Bureau of Standards, 1969.

[75] Djaka, K. S., Taupin, V., Berbenni, S., and Fressengeas, C., "A numerical spectral approach to solve the dislocation density transport equation," *Modelling and Simulation in Materials Science and Engineering*, vol. 23, no. 6, p. 065008, 2015.

[76] Dong, Y., Nogaret, T., and Curtin, W. A., "Scaling of dislocation strengthening by multiple obstacle types," *Metallurgical and Materials Transactions a-Physical Metallurgy and Materials Science*, vol. 41A, no. 8, pp. 1954–1960, 2010.

[77] El-Azab, A., "The boundary value problem of dislocation dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 8, no. 1, pp. 37–54, 2000.

[78] El Kadiri, H. and Oppedal, A. L., "A crystal plasticity theory for latent hardening by glide twinning through dislocation transmutation and twin accommodation effects," *Journal of the Mechanics and Physics of Solids*, vol. 58, no. 4, pp. 613–624, 2010.

[79] Eshelby, J. D., *The Determination of the Elastic Field of an Ellipsoidal Inclusion, and Related Problems*, vol. 241. 1957.

[80] Eshelby, J. D. and Laub, T., "Interpretation of terminating dislocations," *Canadian Journal of Physics*, vol. 45, no. 2P3, pp. 887–892, 1967.

[81] Eyre, D. J. and Milton, G. W., "A fast numerical scheme for computing the response of composites using grid refinement," *European Physical Journal-Applied Physics*, vol. 6, no. 1, pp. 41–47, 1999.

[82] Fan, H. D., Aubry, S., Arsenlis, A., and El-Awady, J. A., "The role of twinning deformation on the hardening response of polycrystalline magnesium from discrete dislocation dynamics simulations," *Acta Materialia*, vol. 92, pp. 126–139, 2015.

[83] Ferroni, F., Tarleton, E., and Fitzgerald, S., "Gpu accelerated dislocation dynamics," *Journal of Computational Physics*, vol. 272, pp. 619–628, 2014.

[84] Ferroni, F., Tarleton, E., and Fitzgerald, S., "Dislocation dynamics modelling of radiation damage in thin films," *Modelling and Simulation in Materials Science and Engineering*, vol. 22, no. 4, 2014.

[85] Fitzgerald, S. P., Aubry, S., Dudarev, S. L., and Cai, W., "Dislocation dynamics simulation of frank-read sources in anisotropic alpha-fe," *Modelling and Simulation in Materials Science and Engineering*, vol. 20, no. 4, 2012.

[86] Flynn, P., Mote, J., and Dorn, J., *On the Thermally Activated Mechanism of Prismatic Slip in Magnesium Single Crystals*. Materials Research Laboratory, Institute of Engineering Research, University of California, Berkeley, 1960.

[87] Foreman, A. J. E., "Bowing of a dislocation segment," *Philosophical Magazine*, vol. 15, no. 137, p. 1011, 1967.

[88] Foreman, A. J. E. and Makin, M. J., "Dislocation movement through random arrays of obstacles," *Philosophical Magazine*, vol. 14, no. 131, p. 911, 1966.

[89] Franciosi, P., "The concepts of latent hardening and strain-hardening in metallic single-crystals," *Acta Metallurgica*, vol. 33, no. 9, pp. 1601–1612, 1985.

[90] Franciosi, P. and Zaoui, A., "Multislip in fcc crystals: a theoretical approach compared with experimental-data," *Acta Metallurgica*, vol. 30, no. 8, pp. 1627–1637, 1982.

[91] Frank, F., "Crystal dislocations - elementary concepts and definitions," *Philosophical Magazine*, vol. 42, pp. 809–819, 1951.

[92] Friedel, J., *Electron Microscropy and Strength of Crystals*. New York: Interscience, 1963.

[93] Friedel, J., *Dislocations*, vol. 3. Oxford: Pergamon Press, 1964.

[94] Frigo, M. and Johnson, S. G., "The design and implementation of fftw3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005.

[95] Gerard, C., Cailletaud, G., and Bacroix, B., "Modeling of latent hardening produced by complex loading paths in fcc alloys," *International Journal of Plasticity*, vol. 42, pp. 194–212, 2013.

[96] Ghoniem, N. M. and Sun, L. Z., "Fast-sum method for the elastic field off three-dimensional dislocation ensembles," *Physical Review B*, vol. 60, no. 1, pp. 128–140, 1999.

[97] Ghoniem, N. M., Tong, S. H., and Sun, L. Z., "Parametric dislocation dynamics: A thermodynamics-based approach to investigations of mesoscopic plastic deformation," *Physical Review B*, vol. 61, no. 2, pp. 913–927, 2000.

[98] Gore, L. A. Ph.d. thesis, Stanford University, 1980.

[99] Green, A. E. and Zerna, W., *Theoretical elasticity*. Clarendon P, 2nd ed ed., 1968.

[100] Groh, S., Devincre, B., Feyel, F., Kubin, L., Roos, A., and Chaboche, J. L., *Discrete-continuum modeling of metal matrix composites plasticity*, vol. 115 of *Iutam Symposium on Mesoscopic Dynamics of Fracture Process and Materials Strength*. 2004.

[101] Groh, S., Devincre, B., Kubin, L. P., Roos, A., Feyel, F., and Chaboche, J. L., "Dislocations and elastic anisotropy in heteroepitaxial metallic thin films," *Philosophical Magazine Letters*, vol. 83, no. 5, pp. 303–313, 2003.

[102] Groh, S., Marin, E. B., Horstemeyer, M. F., and Bammann, D. J., "Dislocation motion in magnesium: a study by molecular statics and molecular dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 17, no. 7, 2009.

[103] Groma, I., Borbély, A., Székely, F., and Máthis, K., "Evolution of the statistical properties of dislocation ensembles," *Materials Science and Engineering: A*, vol. 400–401, pp. 206–209, 2005.

[104] Gélébart, L. and Mondon-Cancel, R., "Non-linear extension of fft-based methods accelerated by conjugate gradients to evaluate the mechanical behavior of composite materials," *Computational Materials Science*, vol. 77, no. 0, pp. 430–439, 2013.

[105] Haasen, P., "Plastic deformation of nickel single crystals at low temperatures," *Philosophical Magazine*, vol. 3, no. 28, p. 384, 1958.

[106] Haghighat, S. M. H., Fivel, M. C., Fikar, J., and Schaeublin, R., "Dislocation-void interaction in fe: A comparison between molecular dynamics and dislocation dynamics," *Journal of Nuclear Materials*, vol. 386-88, pp. 102–105, 2009.

[107] Han, X. and Ghoniem, N. M., "Stress field and interaction forces of dislocations in anisotropic multilayer thin films," *Philosophical Magazine*, vol. 85, no. 11, pp. 1205–1225, 2005.

[108] Han, X., Ghoniem, N. M., and Wang, Z., "Parametric dislocation dynamics of anisotropic crystals," *Philosophical Magazine*, vol. 83, no. 31-34, pp. 3705–3721, 2003.

[109] Heuser, B., "Small-angle neutron scattering study of dislocations in deformed single-crystal copper," *Journal of Applied Crystallography*, vol. 27, no. 6, pp. 1020–1029, 1994.

[110] Hirth, J. P., Zbib, H. M., and Lothe, J., "Forces on high velocity dislocations," *Modelling and Simulation in Materials Science and Engineering*, vol. 6, no. 2, pp. 165–169, 1998.

[111] Hirth, J. and Lothe, J., *Theory of Dislocations*. Malabar, Florida: Krieger Publishing Company, 1982.

[112] Hoc, T., Devincre, B., and Kubin, L., "Deformation stage of fcc crystals: constitutive modeling," in *Evolution of Deformation Microstructures in 3D* (Gundlach, C., ed.), p. 43–59, Denmark: Risø National Laboratory, 2004.

[113] Honeycombe, R., *The plastic deformation of metals*. Edward Arnold Publishers Ltd, 1968.

[114] Hordon, M. J. and Averbach, B. L., "X-ray measurements of dislocation density in deformed copper and aluminum single crystals," *Acta Metallurgica*, vol. 9, no. 3, pp. 237–246, 1961.

[115] Hosford, W. F., Fleischer, R. L., and Backofen, W. A., "Tensile deformation of aluminum single crystals at low temperatures," *Acta Metallurgica*, vol. 8, no. 3, pp. 187–199, 1960.

[116] Hunter, A., Saied, F., Le, C., and Koslowski, M., "Large-scale 3d phase field dislocation dynamics simulations on high-performance architectures," *International Journal of High Performance Computing Applications*, vol. 25, no. 2, pp. 223–235, 2011.

[117] Jain, A., Duygulu, O., Brown, D. W., Tome, C. N., and Agnew, S. R., "Grain size effects on the tensile properties and deformation mechanisms of a magnesium alloy, az31b, sheet," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 486, no. 1-2, pp. 545–555, 2008.

[118] Jin, Z. H., Gumbsch, P., Albe, K., Ma, E., Lu, K., Gleiter, H., and Hahn, H., "Interactions between non-screw lattice dislocations and coherent twin boundaries in face-centered cubic metals," *Acta Materialia*, vol. 56, no. 5, pp. 1126–1135, 2008.

[119] Jin, Z. H., Gumbsch, P., Ma, E., Albe, K., Lu, K., Hahn, H., and Gleiter, H., "The interaction mechanism of screw dislocations with coherent twin boundaries in different face-centred cubic metals," *Scripta Materialia*, vol. 54, no. 6, pp. 1163–1168, 2006.

[120] Kalidindi, S. R., "Modeling anisotropic strain hardening and deformation textures in low stacking fault energy fcc metals," *International Journal of Plasticity*, vol. 17, no. 6, pp. 837–860, 2001.

[121] Kelley, E. and Hosford, W. F., "Plane-strain compression of magnesium and magnesium alloy crystals," *Transaction of the Metallurgical Society of AIME*, vol. 242, pp. 5–13, 1968.

[122] Knezevic, M., Beyerlein, I. J., Brown, D. W., Sisneros, T. A., and Tome, C. N., "A polycrystal plasticity model for predicting mechanical response and texture evolution during strain-path changes: Application to beryllium," *International Journal of Plasticity*, vol. 49, pp. 185–198, 2013.

[123] Knezevic, M., Capolungo, L., Tome, C. N., Lebensohn, R. A., Alexander, D. J., Mihaila, B., and McCabe, R. J., "Anisotropic stress-strain response and microstructure evolution of textured alpha-uranium," *Acta Materialia*, vol. 60, no. 2, pp. 702–715, 2012.

[124] Knezevic, M., Levinson, A., Harris, R., Mishra, R. K., Doherty, R. D., and Kalidindi, S. R., "Deformation twinning in az31: Influence on strain hardening and texture evolution," *Acta Materialia*, vol. 58, no. 19, pp. 6230–6242, 2010.

[125] Kocks, U. F., "Statistical treatment of penetrable obstacles," *Canadian Journal of Physics*, vol. 45, no. 2, pp. 737–755, 1967.

[126] Kocks, U. F., Argon, A. S., and Ashby, M. F., "Thermodynamics and kinetics of slip," *Progress in Materials Science*, vol. 19, pp. 1–281, 1975.

[127] Kocks, U. F. and Mecking, H., "Physics and phenomenology of strain hardening: the fcc case," *Progress in Materials Science*, vol. 48, no. 3, pp. 171–273, 2003.

[128] Koslowski, M., Cuitiño, A. M., and Ortiz, M., "A phase-field theory of dislocation dynamics, strain hardening and hysteresis in ductile single crystals," *Journal of the Mechanics and Physics of Solids*, vol. 50, no. 12, pp. 2597–2635, 2002.

[129] Kröner, E., "Das fundamentalintegral der anisotropen elastischen differentialgleichungen," *Zeitschrift fur Physik*, vol. 136, p. 402–410, 1953.

[130] Kröner, E., *Statistical Continuum Mechanics: Course Held at the Department of General Mechanics, October, 1971*. Springer-Verlag, 1972.

[131] Kröner, E., *Kontinuumstheorie der versetzungen und eigenspannungen*, vol. 5. Springer, 1958.

[132] Kubin, L., Devincre, B., and Hoc, T., "Modeling dislocation storage rates and mean free paths in face-centered cubic crystals," *Acta Materialia*, vol. 56, no. 20, pp. 6040–6049, 2008.

[133] Kubin, L., Devincre, B., and Hoc, T., "Toward a physical model for strain hardening in fcc crystals," *Materials Science and Engineering: A*, vol. 483–484, pp. 19–24, 2008.

[134] Kubin, L. P., Canova, G., Condat, M., Devincre, B., Pontikis, V., and Bréchet, Y., "Dislocation microstructures and plastic flow: a 3d simulation," in *Solid State Phenomena*, vol. 23, pp. 455–472, Trans Tech Publ, 1992.

[135] Kubin, L. P., Madec, R., and Devincre, B., "Dislocation intersections and reactions in fcc and bcc crystals," *MRS Proceedings*, vol. 779, p. W1. 6, 2003.

[136] Kuhlmann-Wilsdorf, D., "A new theory of workhardening," *Trans. Metall. Soc. AIME*, vol. 224, pp. 1047–1061, 1962.

[137] Kulkarni, A. J., Krishnamurthy, K., Deshmukh, S. P., and Mishra, R. S., "Effect of particle size distribution on strength of precipitation-hardened alloys," *Journal of Materials Research*, vol. 19, no. 9, pp. 2765–2773, 2004.

[138] Kumar, K. S., Van Swygenhoven, H., and Suresh, S., "Mechanical behavior of nanocrystalline metals and alloys1," *Acta Materialia*, vol. 51, no. 19, pp. 5743–5774, 2003.

[139] Landau, L. D. and Lifshitz, E., "Theory of elasticity, vol. 7," *Course of Theoretical Physics*, vol. 3, p. 109, 1986.

[140] Lavrentev, F. F., "The type of dislocation interaction as the factor determining work-hardening," *Materials Science and Engineering*, vol. 46, no. 2, pp. 191–208, 1980.

[141] Lavrentev, F. F. and Pokhil, Y. A., "Effect of forest dislocations in 11-22 ¡11-23¿ system on hardening in mg single-crystals under basal slip," *Physica Status Solidi a-Applied Research*, vol. 32, no. 1, pp. 227–232, 1975.

[142] Lavrentev, F. F. and Pokhil, Y. A., "Relation of dislocation density in different slip systems to work-hardening parameters for magnesium crystals," *Materials Science and Engineering*, vol. 18, no. 2, pp. 261–270, 1975.

[143] Lebensohn, R. A., "N-site modeling of a 3d viscoplastic polycrystal using fast fourier transform," *Acta Materialia*, vol. 49, no. 14, pp. 2723–2737, 2001.

[144] Lebensohn, R. A., Kanjarla, A. K., and Eisenlohr, P., "An elasto-viscoplastic formulation based on fast fourier transforms for the prediction of micromechanical fields in polycrystalline materials," *International Journal of Plasticity*, vol. 32-33, pp. 59–69, 2012.

[145] Lee, T. C., Robertson, I. M., and Birnbaum, H. K., "Prediction of slip transfer mechanisms across grain-boundaries," *Scripta Metallurgica*, vol. 23, no. 5, pp. 799–803, 1989.

[146] Lee, V.-G., "Derivatives of the three-dimensional green's functions for anisotropic materials," *International Journal of Solids and Structures*, vol. 46, no. 18-19, pp. 3471–3479, 2009.

[147] Lemarchand, C., Devincre, B., and Kubin, L. P., "Homogenization method for a discrete-continuum simulation of dislocation dynamics," *Journal of the Mechanics and Physics of Solids*, vol. 49, no. 9, pp. 1969–1982, 2001.

[148] LeSar, R. and Rickman, J. M., "Multipole expansion of dislocation interactions: Application to discrete dislocations," *Physical Review B*, vol. 65, no. 14, 2002.

[149] Li, B., Joshi, S., Azevedo, K., Ma, E., Ramesh, K. T., Figueiredo, R. B., and Langdon, T. G., "Dynamic testing at high strain rates of an ultrafine-grained magnesium alloy processed by ecap," *Materials Science and Engineering: A*, vol. 517, no. 1–2, pp. 24–29, 2009.

[150] Liao, X., Zhao, Y., Srinivasan, S., Zhu, Y., Valiev, R., and Gunderov, D., "Deformation twinning in nanocrystalline copper at room temperature and low strain rate," *Applied physics letters*, vol. 84, no. 4, pp. 592–594, 2004.

[151] Lim, L. C. and Raj, R., "The role of residual dislocation arrays in slip induced cavitation, migration and dynamic recrystallization at grain boundaries," *Acta Metallurgica*, vol. 33, no. 12, pp. 2205–2214, 1985.

[152] Liu, B., Eisenlohr, P., Roters, F., and Raabe, D., "Simulation of dislocation penetration through a general low-angle grain boundary," *Acta Materialia*, vol. 60, no. 13–14, pp. 5380–5390, 2012.

[153] Liu, B., Raabe, D., Roters, F., Eisenlohr, P., and Lebensohn, R. A., "Comparison of finite element and fast fourier transform crystal plasticity solvers for texture prediction," *Modelling and Simulation in Materials Science and Engineering*, vol. 18, no. 8, 2010.

[154] Liu, X. H. and Schwarz, K. W., "Modelling of dislocations intersecting a free surface," *Modelling and Simulation in Materials Science and Engineering*, vol. 13, no. 8, pp. 1233–1247, 2005.

[155] Liu, Z. L., Liu, X. M., Zhuang, Z., and You, X. C., "A multi-scale computational model of crystal plasticity at submicron-to-nanometer scales," *International Journal of Plasticity*, vol. 25, no. 8, pp. 1436–1455, 2009.

[156] Livingston, J. D. and Chalmers, B., "Multiple slip in bicrystal deformation," *Acta Metallurgica*, vol. 5, no. 6, pp. 322–327, 1957.

[157] Lothe, J. and Hirth, J. P., "Dislocation climb forces," *Journal of Applied Physics*, vol. 38, no. 2, pp. 845–848, 1967.

[158] Lu, L., Chen, X., Huang, X., and Lu, K., "Revealing the maximum strength in nanotwinned copper," *Science*, vol. 323, no. 5914, pp. 607–610, 2009.

[159] Lu, L., Shen, Y. F., Chen, X. H., Qian, L. H., and Lu, K., "Ultrahigh strength and high electrical conductivity in copper," *Science*, vol. 304, no. 5669, pp. 422–426, 2004.

[160] Lu, L., You, Z. S., and Lu, K., "Work hardening of polycrystalline cu with nanoscale twins," *Scripta Materialia*, vol. 66, no. 11, pp. 837–842, 2012.

[161] Ma, A. and Roters, F., "A constitutive model for fcc single crystals based on dislocation densities and its application to uniaxial compression of aluminium single crystals," *Acta Materialia*, vol. 52, no. 12, pp. 3603–3612, 2004.

[162] Ma, A., Roters, F., and Raabe, D., "A dislocation density based constitutive model for crystal plasticity fem including geometrically necessary dislocations," *Acta Materialia*, vol. 54, no. 8, pp. 2169–2179, 2006.

[163] Ma, A., Roters, F., and Raabe, D., "On the consideration of interactions between dislocations and grain boundaries in crystal plasticity finite element modeling – theory, experiments, and simulations," *Acta Materialia*, vol. 54, no. 8, pp. 2181–2194, 2006.

[164] Madec, R., *Des interactions entre dislocations à la plasticité du monocristal cfc : Etude par Dynamique des Dislocations.* PhD thesis, Universite Paris XI ORSAY, 2001.

[165] Madec, R., Devincre, B., Kubin, L., Hoc, T., and Rodney, D., "The role of collinear interaction in dislocation-induced hardening," *Science*, vol. 301, no. 5641, pp. 1879–1882, 2003.

[166] Madec, R., Devincre, B., and Kubin, L. P., "On the nature of attractive dislocation crossed states," *Computational Materials Science*, vol. 23, no. 1-4, pp. 219–224, 2002.

[167] Madec, R., Devincre, B., and Kubin, L. P., "Simulation of dislocation patterns in multislip," *Scripta Materialia*, vol. 47, no. 10, pp. 689–695, 2002.

[168] Michel, J. C., Moulinec, H., and Suquet, P., "A computational scheme for linear and non-linear composites with arbitrary phase contrast," *International Journal for Numerical Methods in Engineering*, vol. 52, no. 1-2, pp. 139–158, 2001.

[169] Mises, R. V., "Mechanik der plastischen formänderung von kristallen," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 8, no. 3, pp. 161–185, 1928.

[170] Misra, A. and Gibala, R., "Slip transfer and dislocation nucleation processes in multiphase ordered ni-fe-al alloys," *Metallurgical and Materials Transactions A*, vol. 30, no. 4, pp. 991–1001, 1999.

[171] Mohles, V., "Orowan process controlled dislocation glide in materials containing incoherent particles," *Materials Science and Engineering A-Structural Materials Properties Microstructure and Processing*, vol. 309, pp. 265–269, 2001.

[172] Mohles, V., "Simulations of dislocation glide in overaged precipitation-hardened crystals," *Philosophical Magazine a-Physics of Condensed Matter Structure Defects and Mechanical Properties*, vol. 81, no. 4, pp. 971–990, 2001.

[173] Mohles, V. and Nembach, E., "The peak- and overaged states of particle strengthened materials: Computer simulations," *Acta Materialia*, vol. 49, no. 13, pp. 2405–2417, 2001.

[174] Monchiet, V. and Bonnet, G., "A polarization-based fft iterative scheme for computing the effective properties of elastic composites with arbitrary contrast," *International Journal for Numerical Methods in Engineering*, vol. 89, no. 11, pp. 1419–1436, 2012.

[175] Monnet, G. and Devincre, B., "Solute friction and forest interaction," *Philosophical Magazine*, vol. 86, no. 11, pp. 1555–1565, 2006.

[176] Monnet, G., Devincre, B., and Kubin, L. P., "Dislocation study of prismatic slip systems and their interactions in hexagonal close packed metals: application to zirconium," *Acta Materialia*, vol. 52, no. 14, pp. 4317–4328, 2004.

[177] Moulinec, H. and Silva, F., "Comparison of three accelerated fft-based schemes for computing the mechanical response of composite materials," *International Journal for Numerical Methods in Engineering*, vol. 97, no. 13, pp. 960–985, 2014.

[178] Moulinec, H. and Suquet, P., "A fast numerical-method for computing the linear and nonlinear mechanical-properties of composites," *Comptes Rendus De L Academie Des Sciences Serie Ii*, vol. 318, no. 11, pp. 1417–1423, 1994.

[179] Moulinec, H. and Suquet, P., "A numerical method for computing the overall response of nonlinear composites with complex microstructure," *Computer*

Methods in Applied Mechanics and Engineering, vol. 157, no. 1-2, pp. 69–94, 1998.

[180] Mura, T., "Continuous distribution of moving dislocations," *Philosophical Magazine*, vol. 8, no. 89, pp. 843–857, 1963.

[181] Mura, T., *Micromechanics of Defects in Solids*. Dordrecht: Martinus Nijhoff, 1982.

[182] Mura, T. and Kinoshit.N, "Greens functions for anisotropic elasticity," *Physica Status Solidi B-Basic Research*, vol. 47, no. 2, pp. 607–618, 1971.

[183] Nembach, E., *Particle strengthening of metals and alloys*. New York: Wiley, 1996.

[184] Nie, J. F., "Effects of precipitate shape and orientation on dispersion strengthening in magnesium alloys," *Scripta Materialia*, vol. 48, no. 8, pp. 1009–1015, 2003.

[185] Nie, J.-F., "Precipitation and hardening in magnesium alloys," *Metallurgical and Materials Transactions a-Physical Metallurgy and Materials Science*, vol. 43A, no. 11, pp. 3891–3939, 2012.

[186] NVIDIA Corporation, *CUDA Toolkit Documentation v7.0*. NVIDIA Corporation, 2015.

[187] Nye, J., "Some geometrical relations in dislocated crystals," *Acta metallurgica*, vol. 1, no. 2, pp. 153–162, 1953.

[188] Obara, T., Yoshinga, H., and Morozumi, S., "11-22 ¡11-23¿ slip system in magnesium," *Acta Metallurgica*, vol. 21, no. 7, pp. 845–853, 1973.

[189] Oppedal, A. L., El Kadiri, H., Tome, C. N., Kaschner, G. C., Vogel, S. C., Baird, J. C., and Horstemeyer, M. F., "Effect of dislocation transmutation on modeling hardening mechanisms by twinning in magnesium," *International Journal of Plasticity*, vol. 30-31, pp. 41–61, 2012.

[190] Pan, Y. C. and Chou, T. W., "Point force solution for an infinite transversely isotropic solid," *Journal of Applied Mechanics-Transactions of the Asme*, vol. 43, no. 4, pp. 608–612, 1976.

[191] Pant, P., Schwarz, K. W., and Baker, S. P., "Dislocation interactions in thin fcc metal films," *Acta Materialia*, vol. 51, no. 11, pp. 3243–3258, 2003.

[192] Parameswaran, V., Weertman, J., and Urabe, N., "Dislocation mobility in aluminum," *Journal of Applied Physics*, vol. 43, no. 7, pp. 2982–2986, 1972.

[193] Partridge, P. G., "The crystallography and deformation modes of hexagonal close-packed metals," *Metallurgical Reviews*, vol. 12, no. 1, pp. 169–194, 1967.

[194] Peirce, D., Asaro, R. J., and Needleman, A., "An analysis of nonuniform and localized deformation in ductile single-crystals," *Acta Metallurgica*, vol. 30, no. 6, pp. 1087–1119, 1982.

[195] Proust, G., Tomé, C. N., and Kaschner, G. C., "Modeling texture, twinning and hardening evolution during deformation of hexagonal materials," *Acta Materialia*, vol. 55, no. 6, pp. 2137–2148, 2007.

[196] Proust, G., Tome, C. N., Jain, A., and Agnew, S. R., "Modeling the effect of twinning and detwinning during strain-path changes of magnesium alloy az31," *International Journal of Plasticity*, vol. 25, no. 5, pp. 861–880, 2009.

[197] Queyreau, S., Marian, J., Wirth, B. D., and Arsenlis, A., "Analytical integration of the forces induced by dislocations on a surface element," *Modelling and Simulation in Materials Science and Engineering*, vol. 22, no. 3, 2014.

[198] Queyreau, S., Monnet, G., and Devincre, B., "Slip systems interactions in alpha-iron determined by dislocation dynamics simulations," *International Journal of Plasticity*, vol. 25, no. 2, pp. 361–377, 2009.

[199] Queyreau, S., Monnet, G., and Devincre, B., "Orowan strengthening and forest hardening superposition examined by dislocation dynamics simulations," *Acta Materialia*, vol. 58, no. 17, pp. 5586–5595, 2010.

[200] Rauch, E. F., Gracio, J. J., Barlat, F., and Vincze, G., "Modelling the plastic behaviour of metals under complex loading conditions," *Modelling and Simulation in Materials Science and Engineering*, vol. 19, no. 3, 2011.

[201] Reed-Hill, R. E. and Robertson, W. D., "Additional modes of deformation twinning in magnesium," *Acta Metallurgica*, vol. 5, no. 12, pp. 717–727, 1957.

[202] Rhee, M., Stolken, J. S., Bulatov, V. V., de la Rubia, T. D., Zbib, H. M., and Hirth, J. P., "Dislocation stress fields for dynamic codes using anisotropic elasticity: methodology and analysis," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 309, pp. 288–293, 2001.

[203] Rhee, M., Zbib, H. M., Hirth, J. P., Huang, H., and de la Rubia, T., "Models for long-/short-range interactions and cross slip in 3d dislocation simulation of bcc single crystals," *Modelling and Simulation in Materials Science and Engineering*, vol. 6, no. 4, pp. 467–492, 1998.

[204] Robertson, I. M., Beaudoin, A., Al-Fadhalah, K., Chun-Ming, L., Robach, J., Wirth, B. D., Arsenlis, A., Ahn, D., and Sofronis, P., "Dislocation–obstacle interactions: Dynamic experiments to continuum modeling," *Materials Science and Engineering: A*, vol. 400–401, pp. 245–250, 2005.

[205] Rodney, D. and Phillips, R., "Structure and strength of dislocation junctions: An atomic level analysis," *Physical Review Letters*, vol. 82, no. 8, pp. 1704–1707, 1999.

[206] Roters, F., Raabe, D., and Gottstein, G., "Work hardening in heterogeneous alloys - a microstructural approach based on three internal state variables," *Acta Materialia*, vol. 48, no. 17, pp. 4181–4189, 2000.

[207] Saada, G., "Sur le durcissement dû a la recombinaison des dislocations," *Acta Metallurgica*, vol. 8, no. 12, pp. 841–847, 1960.

[208] Salem, A. A., Kalidindi, S. R., and Semiatin, S. L., "Strain hardening due to deformation twinning in alpha-titanium: Constitutive relations and crystal-plasticity modeling," *Acta Materialia*, vol. 53, no. 12, pp. 3495–3502, 2005.

[209] Schoeck, G. and Frydman, R., "Contribution of dislocation forest to flow-stress," *Physica Status Solidi B-Basic Research*, vol. 53, no. 2, p. 661, 1972.

[210] Seeger, A. in *Radiation Damage in Solids*, vol. 1, p. 101–127, Vienna: IAEA, 1962.

[211] Seeger, A. and Schöck, G., "Die aufspaltung von versetzungen in metallen dichtester kugelpackung," *Acta Metallurgica*, vol. 1, no. 5, pp. 519–530, 1953.

[212] Senger, J., Weygand, D., Gumbsch, P., and Kraft, O., "Discrete dislocation simulations of the plasticity of micro-pillars under uniaxial loading," *Scripta Materialia*, vol. 58, no. 7, pp. 587–590, 2008.

[213] Serra, A. and Bacon, D. J., "Computer-simulation of screw dislocation interactions with twin boundaries in hcp metals," *Acta Metallurgica Et Materialia*, vol. 43, no. 12, pp. 4465–4481, 1995.

[214] Shen, Z., Wagoner, R. H., and Clark, W. A. T., "Dislocation pile up and grain-boundary interactions in 304 stainless-steel," *Scripta Metallurgica*, vol. 20, no. 6, pp. 921–926, 1986.

[215] Shen, Z., Wagoner, R. H., and Clark, W. A. T., "Dislocation and grain-boundary interactions in metals," *Acta Metallurgica*, vol. 36, no. 12, pp. 3231–3242, 1988.

[216] Shin, C., Fivel, M., and Kim, W. W., "Three-dimensional computation of the interaction between a straight dislocation line and a particle," *Modelling and Simulation in Materials Science and Engineering*, vol. 13, no. 7, pp. 1163–1173, 2005.

[217] Shin, C. S., Fivel, M. C., Verdier, M., and Oh, K. H., "Dislocation-impenetrable precipitate interaction: a three-dimensional discrete dislocation dynamics analysis," *Philosophical Magazine*, vol. 83, no. 31-34, pp. 3691–3704, 2003.

[218] Shin, C. S., Fivel, M. C., Verdier, M., and Robertson, C., "Dislocation dynamics simulations of fatigue of precipitation-hardened materials," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 400, pp. 166–169, 2005.

[219] Shin, C., Fivel, M., and Oh, K., "Nucleation and propagation of dislocations near a precipitate using 3d discrete dislocation dynamics simulations," *Le Journal de Physique IV*, vol. 11, no. PR5, pp. Pr5–27–Pr5–34, 2001.

[220] Sills, R. B. and Cai, W., "Efficient time integration in dislocation dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 22, no. 2, 2014.

[221] Sobie, C., Bertin, N., and Capolungo, L., "Analysis of obstacle hardening models using dislocation dynamics: Application to irradiation-induced defects," *Metallurgical and Materials Transactions A*, vol. 46, no. 8, pp. 3761–3772, 2015.

[222] Spearot, D. E. and Sangid, M. D., "Insights on slip transmission at grain boundaries from atomistic simulations," *Current Opinion in Solid State and Materials Science*, vol. 18, no. 4, pp. 188–195, 2014.

[223] Staker, M. R. and Holt, D. L., "The dislocation cell size and dislocation density in copper deformed at temperatures between 25 and 700 c," *Acta Metallurgica*, vol. 20, no. 4, pp. 569–579, 1972.

[224] Staroselsky, A. and Anand, L., "A constitutive model for hcp materials deforming by slip and twinning: application to magnesium alloy az31b," *International Journal of Plasticity*, vol. 19, no. 10, pp. 1843–1864, 2003.

[225] Stohr, J. F. and Poirier, J. P., "Electron-microscope study of pyramidal slip 11-22 ¡11-23¿ in magnesium," *Philosophical Magazine*, vol. 25, no. 6, p. 1313, 1972.

[226] Takeuchi, T., "Work hardening of copper single crystals with multiple glide orientations," *Transactions of the Japan Institute of Metals*, vol. 16, no. 10, pp. 629–640, 1975.

[227] Taylor, G. I., *The Mechanism of Plastic Deformation of Crystals. Part I. Theoretical*, vol. 145. 1934.

[228] Teodosiu, C., Raphanel, J., and Tabourot, L., "Finite element simulation of the large elastoplastic deformation of multicrystals," in *Teodosiu, Raphanel and Sidoroff*, p. 153–168, Balkena, Rotterdam: Mecamat'91, 1993.

[229] Thompson, N., "Dislocation nodes in face-centred cubic lattices," *Proceedings of the Physical Society of London Section B*, vol. 66, no. 402, pp. 481–492, 1953.

[230] Ting, T. C. T. and Lee, V. G., "The three-dimensional elastostatic green's function for general anisotropic linear elastic solids," *Quarterly Journal of Mechanics and Applied Mathematics*, vol. 50, pp. 407–426, 1997.

[231] Urabe, N. and Weertman, J., "Dislocation mobility in potassium and iron single-crystals," *Materials Science and Engineering*, vol. 18, no. 1, pp. 41–49, 1975.

[232] Van der Giessen, E. and Needleman, A., "Discrete dislocation plasticity: a simple planar model," *Modelling and Simulation in Materials Science and Engineering*, vol. 3, no. 5, p. 689, 1995.

[233] Van Swygenhoven, H., Derlet, P. M., and Frøseth, A. G., "Nucleation and propagation of dislocations in nanocrystalline fcc metals," *Acta Materialia*, vol. 54, no. 7, pp. 1975–1983, 2006.

[234] Vattre, A., Devincre, B., Feyel, F., Gatti, R., Groh, S., Jamond, O., and Roos, A., "Modelling crystal plasticity by 3d dislocation dynamics and the finite element method: The discrete-continuous model revisited," *Journal of the Mechanics and Physics of Solids*, vol. 63, pp. 491–505, 2014.

[235] Vattre, A., Devincre, B., and Roos, A., "Dislocation dynamics simulations of precipitation hardening in ni-based superalloys with high gamma ' volume fraction," *Intermetallics*, vol. 17, no. 12, pp. 988–994, 2009.

[236] Vattre, A., Devincre, B., and Roos, A., "Orientation dependence of plastic deformation in nickel-based single crystal superalloys: Discrete-continuous model simulations," *Acta Materialia*, vol. 58, no. 6, pp. 1938–1951, 2010.

[237] Vattre, A. J. and Demkowicz, M. J., "Determining the burgers vectors and elastic strain energies of interface dislocation arrays using anisotropic elasticity theory," *Acta Materialia*, vol. 61, no. 14, pp. 5172–5187, 2013.

[238] Vattré, A., Abdolrahim, N., Kolluri, K., and Demkowicz, M., "Computational design of patterned interfaces using reduced order models," *Scientific reports*, vol. 4, 2014.

[239] Vattré, A. and Demkowicz, M., "Effect of interface dislocation burgers vectors on elastic fields in anisotropic bicrystals," *Computational Materials Science*, vol. 88, pp. 110–115, 2014.

[240] Verdier, M., Fivel, M., and Groma, I., "Mesoscopic scale simulation of dislocation dynamics in fcc metals: Principles and applications," *Modelling and Simulation in Materials Science and Engineering*, vol. 6, no. 6, pp. 755–770, 1998.

[241] Volterra, V., "Sur l'équilibre des corps élastiques multiplement connexes," *Annales Scientifiques de l'Ecole Normale Supérieure*, vol. 24, pp. 401–517, 1907.

[242] von Blanckenhagen, B., Arst, E., and Gumbsch, P., "Discrete dislocation simulation of plastic deformation in metal thin films," *Acta Materialia*, vol. 52, no. 3, pp. 773–784, 2004.

[243] Wang, H., Wu, P. D., Wang, J., and Tome, C. N., "A crystal plasticity model for hexagonal close packed (hcp) crystals including twinning and detwinning mechanisms," *International Journal of Plasticity*, vol. 49, pp. 36–52, 2013.

[244] Wang, H. and Xiang, Y., "An adaptive level set method based on two-level uniform meshes and its application to dislocation dynamics," *International Journal for Numerical Methods in Engineering*, vol. 94, no. 6, pp. 573–597, 2013.

[245] Wang, J. and Misra, A., "An overview of interface-dominated deformation mechanisms in metallic multilayers," *Current Opinion in Solid State and Materials Science*, vol. 15, no. 1, pp. 20–28, 2011.

[246] Wang, Y. U., Jin, Y. M., Cuitino, A. M., and Khachaturyan, A. G., "Nanoscale phase field microelasticity theory of dislocations: Model and 3d simulations," *Acta Materialia*, vol. 49, no. 10, pp. 1847–1857, 2001.

[247] Wang, Z. Q. and Beyerlein, I. J., "An atomistically-informed dislocation dynamics model for the plastic anisotropy and tension-compression asymmetry of bcc metals," *International Journal of Plasticity*, vol. 27, no. 10, pp. 1471–1484, 2011.

[248] Wang, Z. Q., Beyerlein, I. J., and Lesar, R., "Dislocation motion in high strain-rate deformation," *Philosophical Magazine*, vol. 87, no. 16-17, pp. 2263–2279, 2007.

[249] Wang, Z. Q., Beyerlein, I. J., and LeSar, R., "The importance of cross-slip in high-rate deformation," *Modelling and Simulation in Materials Science and Engineering*, vol. 15, no. 6, p. 675, 2007.

[250] Wang, Z. Q., Beyerlein, I. J., and LeSar, R., "Plastic anisotropy in fcc single crystals in high rate deformation," *International Journal of Plasticity*, vol. 25, no. 1, pp. 26–48, 2009.

[251] Wang, Z., Ghoniem, N., Swaminarayan, S., and LeSar, R., "A parallel algorithm for 3d dislocation dynamics," *Journal of Computational Physics*, vol. 219, no. 2, pp. 608–621, 2006.

[252] Weinberger, C. R., Aubry, S., Lee, S. W., Nix, W. D., and Cai, W., "Modelling dislocations in a free-standing thin film," *Modelling and Simulation in Materials Science and Engineering*, vol. 17, no. 7, 2009.

[253] Weygand, D., Friedman, L. H., van der Giessen, E., and Needleman, A., "Discrete dislocation modeling in three-dimensional confined volumes," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 309, pp. 420–424, 2001.

[254] Weygand, D., Friedman, L. H., Van der Giessen, E., and Needleman, A., "Aspects of boundary-value problem solutions with three-dimensional dislocation dynamics," *Modelling and Simulation in Materials Science and Engineering*, vol. 10, no. 4, pp. 437–468, 2002.

[255] Weygand, D., Poignant, M., Gumbsch, P., and Kraft, O., "Three-dimensional dislocation dynamics simulation of the influence of sample size on the stress-strain behavior of fcc single-crystalline pillars," *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing*, vol. 483-84, pp. 188–190, 2008.

[256] Wickham, L. K., Schwarz, K. W., and Stölken, J. S., "Rules for forest interactions between dislocations," *Physical Review Letters*, vol. 83, no. 22, pp. 4574–4577, 1999.

[257] Willis, J. R., "Stress fields produced by dislocations in anisotropic media," *Philosophical Magazine*, vol. 21, no. 173, pp. 931–949, 1970.

[258] Willot, F., "Fourier-based schemes for computing the mechanical response of composites with accurate local fields," *Comptes Rendus Mecanique*, vol. 343, no. 3, pp. 232–245, 2015.

[259] Willot, F., Abdallah, B., and Pellegrini, Y.-P., "Fourier-based schemes with modified green operator for computing the electrical response of heterogeneous media with accurate local fields," *International Journal for Numerical Methods in Engineering*, vol. 98, no. 7, pp. 518–533, 2014.

[260] Wu, C. C., Chung, P. W., Aubry, S., Munday, L. B., and Arsenlis, A., "The strength of binary junctions in hexagonal close-packed crystals," *Acta Materialia*, vol. 61, no. 9, pp. 3422–3431, 2013.

[261] Wu, X. L., Liao, X. Z., Srinivasan, S. G., Zhou, F., Lavernia, E. J., Valiev, R. Z., and Zhu, Y. T., "New deformation twinning mechanism generates zero macroscopic strain in nanocrystalline metals," *Physical Review Letters*, vol. 100, no. 9, p. 095701, 2008.

[262] Xiang, Y., Cheng, L. T., Srolovitz, D. J., and E, W. N., "A level set method for dislocation dynamics," *Acta Materialia*, vol. 51, no. 18, pp. 5499–5518, 2003.

[263] Yin, J., Barnett, D. M., Fitzgerald, S. P., and Cai, W., "Computing dislocation stress fields in anisotropic elastic media using fast multipole expansions," *Modelling and Simulation in Materials Science and Engineering*, vol. 20, no. 4, 2012.

[264] Yin, J., Barnett, D. M., and Cai, W., "Efficient computation of forces on dislocation segments in anisotropic elasticity," *Modelling and Simulation in Materials Science and Engineering*, vol. 18, no. 4, 2010.

[265] Yuasa, M., Masunaga, K., Mabuchi, M., and Chino, Y., "Interaction mechanisms of screw dislocations with 10(1)over-bar1 and 10(1)over-bar2 twin boundaries in mg," *Philosophical Magazine*, vol. 94, no. 3, pp. 285–305, 2014.

[266] Zbib, H. M., Rhee, M., and Hirth, J. P., "On plastic deformation and the dynamics of 3d dislocations," *International Journal of Mechanical Sciences*, vol. 40, no. 2-3, pp. 113–127, 1998.

[267] Zeman, J., Vondřejc, J., Novák, J., and Marek, I., "Accelerating a fft-based solver for numerical homogenization of periodic media by conjugate gradients," *Journal of Computational Physics*, vol. 229, no. 21, pp. 8065–8071, 2010.

[268] Zeng, R., *Precipitation hardening in AZ91 magnesium alloy*. Ph.d., University of Birmingham, 2013.

[269] Zhang, F., Feng, X., Yang, Z., Kang, J., and Wang, T., "Dislocation-twin boundary interactions induced nanocrystalline via spd processing in bulk metals," *Sci. Rep.*, vol. 5, 2015.

[270] Zhao, D., Huang, J., and Xiang, Y., "A new version fast multipole method for evaluating the stress field of dislocation ensembles," *Modelling and Simulation in Materials Science and Engineering*, vol. 18, no. 4, 2010.

[271] Zhou, C., Biner, S. B., and LeSar, R., "Discrete dislocation dynamics simulations of plasticity at small scales," *Acta Materialia*, vol. 58, no. 5, pp. 1565–1577, 2010.

[272] Zhou, C. and LeSar, R., "Dislocation dynamics simulations of plasticity in polycrystalline thin films," *International Journal of Plasticity*, vol. 30-31, pp. 185–201, 2012.

[273] Zhu, T., Li, J., Samanta, A., Kim, H. G., and Suresh, S., "Interfacial plasticity governs strain rate sensitivity and ductility in nanostructured metals," *Proceedings of the National Academy of Sciences*, vol. 104, no. 9, pp. 3031–3036, 2007.

[274] Zhu, Y. T., Liao, X. Z., and Wu, X. L., "Deformation twinning in nanocrystalline materials," *Progress in Materials Science*, vol. 57, no. 1, pp. 1–62, 2012.

[275] Zhu, Y. T., Wu, X. L., Liao, X. Z., Narayan, J., Kecskes, L. J., and Mathaudhu, S. N., "Dislocation-twin interactions in nanocrystalline fcc metals," *Acta Materialia*, vol. 59, no. 2, pp. 812–821, 2011.

[276] Zikry, M. A. and Kao, M., "Inelastic microstructural failure modes in crystalline materials: The s33a and s11 high angle grain boundaries," *International Journal of Plasticity*, vol. 13, no. 4, pp. 403–434, 1997.