Title

Computational Tools for Modeling Planar Defects in Alloys and Designing Materials Science Curricula

Author

Chen, Enze

Publication Date

2023

Peer reviewed|Thesis/dissertation

Computational Tools for Modeling Planar Defects in Alloys and Designing Materials Science Curricula

By

Enze Chen

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering – Materials Science and Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Mark Asta, Co-chair
Dr. Timofey Frolov, Co-chair
Professor Daryl C. Chrzan
Professor Grace X. Gu

Fall 2023

Abstract

Computational Tools for Modeling Planar Defects in Alloys and Designing Materials Science Curricula

by

Enze Chen

Doctor of Philosophy in Engineering – Materials Science and Engineering

University of California, Berkeley

Professor Mark Asta, Co-chair

Dr. Timofey Frolov, Co-chair

In recent years, the increased availability of computing resources has enabled the construction of public databases/datasets from automated, high-throughput calculations for materials structure and properties, often at the atomic level. Rising alongside the heightened popularity of data science and machine learning (ML) techniques, these datasets have greatly benefited the community by guiding materials design and the development of ML models such as universal force fields. Thus far, these computational workflows have primarily been developed for bulk crystals and molecules, and they are relatively less mature for interfacial properties. In the third chapter, I will present my work building a workflow for calculating antiphase boundary (APB) energies in Ni_3Al, which is important for the high-temperature performance of Ni-based superalloys in jet turbines. This workflow combines statistical thermodynamics, first-principles calculations, and machine learning to produce a database and predictive models for the APB energy. In the fourth chapter, I will present a computational tool, the GRand canonical Interface Predictor (GRIP), that can be used for high-throughput, grand canonical optimization of grain boundary (GB) structures. We specifically demonstrate the application of GRIP on tilt GBs in hexagonal close-packed (HCP) titanium, as previous GB optimization studies focused solely on cubic systems. By allowing the density of atoms in the GB to vary, we discover novel GB phases in HCP Ti with distinct localized dislocation cores. We use high-temperature molecular dynamics to validate phase stability and study phase transformations through a novel point defect-induced dislocation pairing mechanism. In the fifth chapter, I present some complementary work where I use many of these same tools to enhance materials informatics education, as there exists a critical need to train materials scientists to harness the growing amounts of computing resources and scientific data. Our nation's workforce must be proficient in these skills in order to accelerate materials development to solve grand challenges in areas such as sustainability and public

health. Specifically, I use the Jupyter Book software to design interactive, digital texts that we have successfully used for a summer research curriculum and integrated into a materials characterization laboratory course. I will conclude with some ideas for how to build on the work in this dissertation to continue pushing the frontiers of research and teaching in materials science and engineering.

To my grandparents

and EJR

Contents

1 Introduction **1**
1.1 Planar defects in metal alloys 1
1.2 Computational alloy design 3
1.3 Training data-proficient engineers 4

2 Computational Methods **5**
2.1 Atomistic simulations 5
2.1.1 Density functional theory (DFT) 5
2.1.2 Molecular dynamics (MD) 6
2.2 Controlling degrees of freedom 7
2.2.1 Solute distributions in intermetallics 7
2.2.2 GB structure optimization 10
2.2.3 GRand canonical Interface Predictor (GRIP) 11
2.3 Jupyter ecosystem 15
2.3.1 Jupyter Books for digital textbooks 16

3 Antiphase boundaries in Ni-based superalloys **17**
3.1 Introduction to Ni-based superalloys 17
3.2 Methods 20
3.2.1 Solute site preference 20
3.2.2 DFT calculations for γ_{111} 21
3.2.3 Machine learning 23
3.3 Results 24
3.3.1 Solute site preference 24
3.3.2 APB energy 26
3.3.3 Machine learning modeling 28
3.3.4 Extrapolation to multicomponent chemistries 31
3.4 Discussion 32

4 Grain boundary optimization in α-Ti using GRIP **35**
4.1 Introduction to GB structure search 35
4.2 Methods 37

4.2.1 GB structure search . . . 37
4.2.2 High-temperature MD simulations . . . 37
4.3 Results . . . 38
4.3.1 Grand canonical optimization . . . 38
4.3.2 Phase transitions and coexistence . . . 43
4.4 Discussion . . . 44

5 Data-centric materials science and engineering education 49
5.1 Introduction . . . 49
5.2 MI summer internship . . . 50
5.2.1 Implementation . . . 51
5.2.2 Results and Discussion . . . 54
5.3 MSE 104L data analysis . . . 56
5.3.1 Implementation . . . 57
5.3.2 Results and Discussion . . . 58

6 Conclusion and outlook 64
6.1 Open-access GB database . . . 64
6.2 Universal design for learning . . . 66

Bibliography 68

A APBs in Ni_3Al 89
A.1 ML model without ordering energy . . . 89

B GBs in α-Ti 93

C MSE education 97

Acknowledgments

First and foremost, I am grateful for the steadfast mentorship from my co-advisors, Mark Asta and Timofey Frolov. Mark, for giving me freedom to explore my varied interests, keeping his office door open for any conversation, and expanding my views of what's possible. And Tim, for always championing my professional success and teaching me to embrace the uncertainties of research and life—that it's OK to not schedule the next (or any) meeting.

The remaining list of people is far too long to fit here. Certainly I thank my family members—Mom, Dad, Harry—and my partner Cora for their generous love and support, especially through the COVID-19 pandemic. I also thank Daryl Chrzan and Grace Gu (dissertation committee), and Andrew Minor and Gerbrand Ceder (qualifying exam committee) for their energy and insightful comments through the years. Additional mentors include Michelle Wilkerson, Linda von Hoene, the MSE Department Staff, the Asta group members, and my PhD cohort ("Squad")—thank you all for being along for the ride.

I am also grateful for the many sources of financial support that made this work possible, the primary source being the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-2146752. I also acknowledge summer funding from the Materials and Chemistry Institute at Lawrence Livermore National Laboratory (LLNL); the Computational Chemistry & Materials Science Institute at LLNL; and the U.S. Department of Energy (DOE), Office of Science, Office of Basic Energy Sciences, Materials Sciences and Engineering Division, under Contract No. DE-AC02-05-CH11231 within the Materials Project program (KC23MP).

Computing resources for this work come from the LLNL Institutional Computing facilities; the National Energy Research Scientific Computing Center (NERSC), a DOE Office of Science User Facility operated under Contract No. DE-AC02-05CH11231; and Bridges-2 at Pittsburgh Supercomputing Center through allocation DMR110087 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program. The DataHub servers are generously supported by the College of Computing, Data Science, and Society at UC Berkeley.

Chapter 1

Introduction

Materials science and engineering (MSE) as an academic discipline has undergone significant transformation over the years [1]. What began as departments focused on the practice of mining, metallurgy, and ceramics (including at UC Berkeley) have since evolved to encompass a diverse range of other materials including soft matter, composites, nanomaterials, and materials for opto-electronic applications. This interdisciplinary field that I have called home for the past ten years continues to be shaped by societal needs and forces, which begs the question: What will the future of MSE look like?

This question is far too broad for a single dissertation to address, so we will narrow the discussion to one particular area rooted in the metallurgical origins of the field. The ubiquity of metals and their alloys in society underscores their importance, but in order to solve global challenges in sustainability, infrastructure, aerospace, national security, and more, we have to revamp the traditional alloy design process by taking advantage of new developments in physics-based modeling and artificial intelligence. These technological advancements hope to accelerate the materials research and development (R&D) process, bring better materials to market, faster, and with reduced cost. While there has been rapid progress in this space, as detailed in the following chapters, most of the successes from research labs have not yet been transferred into the classroom, creating a talent gap that could be bridged with improved pedagogy. Consequently, there is an exciting opportunity at the intersection of materials, computing, and education to maximize the future impact MSE will have on our society, and this convergence will be further explored in this dissertation.

1.1 Planar defects in metal alloys

Most of the metallic materials of practical significance are crystalline, meaning they have long-range, periodic order as illustrated in Figure 1.1a. While crystallinity is often a desirable characteristic that results in predictable behavior, it is in fact the places where this crystalline order breaks down—and *defects* are introduced—that imbue a material with truly remarkable properties [2]. Given the plethora of ways one can destroy the perfect arrangement of atoms

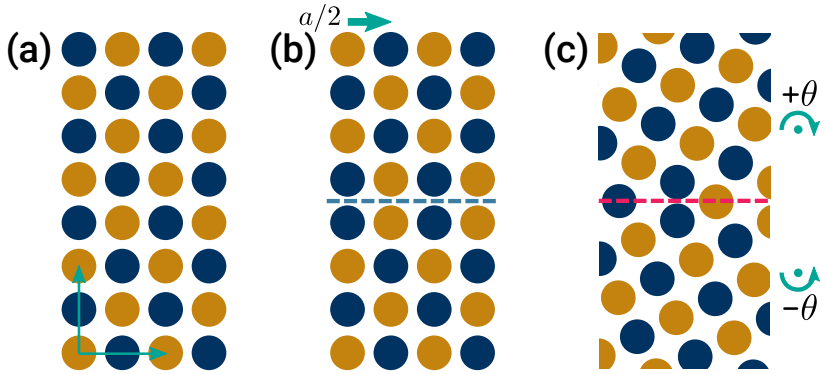


Figure 1.1: Illustration of planar defects. (a) A bulk perfect crystal with two species colored blue and gold on a simple cubic lattice. (b) The top half has been shifted by half of a lattice vector to create an antiphase boundary (APB; dashed blue line). (c) The top half has been rotated by an angle $+\theta$ (rotation axis normal to the page) and the bottom by an angle $-\theta$ to create a symmetric tilt grain boundary (STGB; dashed magenta line).

in a crystal, defects in solids are often grouped by their dimensionality for convenience and to illustrate the extent of their influence. Point defects (zero-dimensional) occur at individual sites in a crystal, such as an atom that is missing from its equilibrium site (vacancy), an atom occupying the small gaps in the crystal (interstitial), or an atom that has swapped into another atom's site (substitution). One-dimensional line defects such as dislocations occur in segments that snake throughout the crystal and help accommodate atomic displacements during deformation, among other important properties. Planar defects form when different crystalline regions, or grains, meet at a two-dimensional interface, where the structural mismatch is often associated with a variety of excess properties, such as excess energy and volume (that differ from the bulk solid). Lastly, there are 3D volume defects at larger length scales such as pores and precipitates that will also influence a material's properties and performance. Ultimately, it can be said that the goal of the MSE discipline, and what sets it apart from other fields of study, is obtaining deeper understanding about and greater control over the defects that appear in the materials we use.

This dissertation will primarily focus on planar defects in structural alloys [3], although the interplay between planar defects and other classes of defects will also be discussed. One common type of planar defect that occurs in ordered solids is an antiphase boundary (APB), as illustrated in Figure 1.1b. An APB can form when part of the crystal is sheared relative to the other half, perhaps by the passage of a dislocation, disrupting the ordering locally near the APB. As the formation of APBs creates energetically unfavorable bonding environments, the APB energy (γ_{apb}) quantifies an energetic barrier to dislocation motion and this barrier contributes to the strengthening of precipitation-strengthened alloys. One such example is the strength provided by Ni_3Al precipitates in Ni-based superalloys, which have applications in turbine blades and land-based power generators [4]. Chapter 3 will further explore the

properties of APBs in Ni_3Al and how they are affected by chemical heterogeneity in the precipitate phase. I will share our computational workflow for high-throughput generation of APB data, which were then used to train algorithms for predicting the APB energy.

More generally, two grains in a polycrystalline material can arbitrarily intersect to form a grain boundary (GB), which has a variety of excess properties with important consequences for strengthening [5, 6], embrittlement [7], diffusion [8], and microstructure evolution [9]. Based on the orientations of the two parent grains, the crystallography of a GB is traditionally described using five macroscopic degrees of freedom (DOF) [3]. There are three DOF associated with the 3D misorientation of one grain relative to the other, and two more DOF for the inclination of the GB plane separating the two crystals. Moreover, there are many more microscopic DOF including the relative translation of one grain and the addition/removal of atoms at the GB, leading to a rich spectrum of GB structures. While the GB character in a real material is quite complex, there are simpler models of GBs that are still quite powerful and commonly employed to provide insights into GB properties. One such class is the symmetric tilt grain boundary (STGB), shown in Figure 1.1c, where the top and bottom grains are rotated in opposite direction by an angle θ relative to an axis perpendicular to the GB normal. STGBs are relatively easy to construct and they include an important subclass of twin boundaries (TBs), which are coherent interfaces with mirror symmetry across the GB plane and hence low GB energy (E_{gb}). The clear crystallography of STGBs makes them attractive models to study GB-related phenomena, and this will be further discussed in Chapter 4 in the context of hexagonal closed-packed titanium (α-Ti). In particular, I will share our workflow for grand canonical optimization of GB structures, which we used to discover new GB phases in α-Ti.

1.2 Computational alloy design

The structure-property relationships of interfaces like APBs and GBs have been exploited for decades to achieve many successful results in alloy design. Going forward, there exists an opportunity to combine experimental efforts with powerful computing techniques to deepen our understanding and accelerate the development process through an integrated computational materials engineering (ICME) framework [10]. Analogous to the field of MSE, ICME is a multiscale framework ranging from the atomic to the macroscale and the work presented in this dissertation will be focused at the atomic level, i.e., atomistics.

The use of atomistic simulations in materials research has rapidly increased due to the maturation of density functional theory (DFT) [11] and molecular dynamics (MD) [12] codes, and advancements in computing have enabled simulations of increasing size and complexity. Nowadays, it is relatively routine to develop computational workflows for high-throughput simulations that generate vast amounts of data [13, 14]. These data are not only used for downstream tasks in the traditional ICME framework (e.g., inputs for coarse-grained models), but also analyzed using data science (DS) techniques to build predictive statistical models. The Materials Genome Initiative [15, 16] is a recent effort to develop appropriate

infrastructure to accelerate materials discovery by leveraging advanced machine learning (ML) algorithms to analyze the wealth of complex data [17]. While significant progress in data-driven materials design has been made in the field of structural alloys [18, 19], the majority of databases and studies focus on bulk crystals and molecules, with relatively less work on interfaces. Chapter 2 reviews several of the computational techniques that were used to perform the work in this dissertation, with a particular focus on how they can be applied to study interfacial structure–property relationships at the atomic level [20].

1.3 Training data-proficient engineers

While computational methods and paradigms like ICME are often celebrated for their anticipated impact, a critical aspect of this transformation is workforce development, ensuring that current practitioners and next-generation scientists will be able to take full advantage of these tools. Recent surveys of academic departments in the US have shown an interest from faculty in promoting more computational literacy in MSE courses to meet post-graduate needs [21, 22], and there are several programs around the country that have enhanced their curriculum accordingly [23, 24]. These programs have documented success in modernizing MSE curricula and helping our students seize the opportunities of tomorrow.

However, given the rapidly changing landscape of computational materials science, there are still many opportunities remaining in this space. As materials research becomes increasingly invested in DS and ML methods, there is a desire to create new materials informatics (MI) curricula to meet those needs [25]. Most of the aforementioned examples focus on physics-based computational models, and it remains unclear (and untested) how to overcome gaps and barriers for MI education [26]. Furthermore, there are many accessibility challenges for computational MSE education, often arising from difficulties in transferring research codes into pedagogical content. These barriers make it difficult for instructors to create new curricula and they often hamper the student learning experience as students are faced with software troubleshooting. In Chapter 5, I will discuss some of the ways we have tried to address these problems using the open-source Jupyter ecosystem. These include a standalone MI research curriculum and instances where we integrate computing and DS into core MSE courses at UC Berkeley.

Finally, Chapter 6 rounds out the dissertation by summarizing the ideas presented herein and formulating some future directions for the computational study of planar defects and MSE education.

Chapter 2

Computational Methods

This chapter aims to cover the general theoretical foundations behind the methods presented in this dissertation and how those motivate the choice of appropriate tools. Specific choices of settings (e.g., input parameters) pertaining to a particular application will be discussed in the relevant chapters.

2.1 Atomistic simulations

Atomistic simulations rely on the fundamental laws of physics to parameterize atomic-level interactions and have been used since the early 50's to gain insights into the fundamental behavior of materials. Generally speaking, these algorithms allow users to precisely specify a set of atomic positions (i.e., xyz-coordinates) and atomic attributes (e.g., elemental identity, charge), from which a new set of positions and properties are calculated. With advances in computing, it is now possible to augment traditional materials design methods with high-throughput atomistic simulations to rapidly explore regions of the design space [13, 14], as described in subsequent chapters. While atomistic simulations are attractive for their ability to elucidate fundamental structure–property relationships in materials, there exists an inherent trade-off between how detailed we model the physical interactions (i.e., the accuracy) and the computational complexity (i.e., the runtime) of the simulations. Below, we briefly discuss two common approaches to atomistic modeling that are used for the work in this dissertation.

2.1.1 Density functional theory (DFT)

Ideally, we would like to use the fundamental equations of quantum mechanics, such as the Schrödinger equation, to model the atoms' electrons and solve these equations to obtain the energy, forces, and other properties of the system [27]. Given that this is an impossible task for all but the simplest systems, there have been several alternative approaches developed to approximate such a solution, and many of these first-principles, or *ab initio*, methodologies

have been quite successful. One such method is called density functional theory (DFT), initially developed by Hohenberg and Kohn in 1964 [28] and extended by Kohn and Sham the following year [29].

In DFT, instead of solving the true Schrödinger equation for the ground-state energy and wave function, we can solve a variational form that is expressed as a functional of the electron density. This functional contains terms with well-known expressions in the independent-particle approximation (e.g., Coulombic interactions, non-interacting kinetic energy) as well as an unknown exchange-correlation term that subsumes all the remaining quantum mechanical contributions to the energy. While there have been many analytical forms proposed for the exchange-correlation functional (a so-called Jacob's ladder of increasing complexity [30]), we have chosen the generalized gradient approximation (GGA) by Perdew, Burke, and Ernzerhof [31] for all of the DFT calculations here, given its reasonable accuracy on our metallic systems of interest. For further computational efficiency, we use projector-augmented wave (PAW) potentials [32,33] as implemented in the Vienna Ab Initio Simulation Package (VASP) [34–37].

Ab initio techniques like DFT are attractive because we can use detailed electronic interactions to solve for the energy of any chemical system from first principles. In exchange for its flexibility and (relative) accuracy, DFT suffers from being computationally expensive and works practically for at most a few hundred atoms (for the most common algorithms and computing resources). Periodic boundary conditions are often employed to emulate the "infinite" periodicity of a crystal, which means particular care must be taken when simulating extended defects like APBs to avoid periodic image effects, as discussed in Chapter 3. For more information regarding DFT and other electronic-structure methods, we encourage the reader to consult a reference textbook like the ones by Sholl and Steckel [11] and Martin [27].

2.1.2 Molecular dynamics (MD)

Alternatively, instead of worrying about the electrons, we can also treat the atoms classically and model their interactions through descriptions of the structure of neighboring particles (e.g., distances, bond angles). In molecular dynamics (MD), the Hamiltonian equations of motion are integrated to evolve the system over time toward an equilibrium state imposed by the boundary conditions. Unlike DFT, MD simulations that make use of classical interatomic potentials routinely contain hundreds of thousands of atoms and are leveraged to study deformation behavior [38], phase transitions [39], defect mobility [40], and more. They can achieve this scale because the interatomic interactions are now parameterized using empirical potentials fit to experimental and simulated data. The potentials range in complexity from classical pair potentials with simple analytical forms [41] to modern machine-learned potentials based on deep neural networks [42]. The MD simulations in this dissertation (Chapter 4) use interatomic potentials (IAPs) with the forms corresponding to the embedded-atom method (EAM) formalism by Daw and Baskes [43] and the modified embedded-atom method (MEAM) formalism by Baskes [44]. Both take the following general form:

$$E_{\text{tot}} = \frac{1}{2}\sum_{i \neq j} \phi_{ij}(r_{ij}) + \sum_{i} F_i(\rho_i) \tag{2.1}$$

where the total energy of the system is a sum of pair interactions, ϕ_{ij}, and a sum of embedded environment interactions, $F_i(\cdot)$, with ρ_i being the density around atom i. For EAM potentials, this density is assumed to vary radially with respect to neighboring atoms j; that is,

$$\rho_i = \sum_{j \neq i} \rho_j^a(r_{ij})$$

whereas for MEAM potentials, this density has both radial and angular components,

$$\rho_i = \sum_{j \neq i} \rho_j^a(r_{ij}) + \sum_{j,k} f(r_{ij}) f(r_{ik}) g[\cos(\theta_{jik})]$$

where the second term contains cutoff functions $f(\cdot)$ that vary with radial distance in addition to an angular contribution ($g[\cdot]$) between atoms j, i, and k.

As expected, the choice of IAP formalism presents yet another trade-off between capturing the relevant physics and computational efficiency. An additional consideration for their accuracy in modeling interfacial phenomena is whether the training data for the fitting procedure includes sufficient local environments representative of the diversity of non-bulk atomic arrangements at interfaces like grain boundaries. Given these factors, we perform the GB structure search in Chapter 4 using both EAM and MEAM potentials. For more information regarding molecular dynamics, we encourage the reader to consult a reference textbook like the ones by Frenkel and Smit [12] and Allen and Tildesley [45].

2.2 Controlling degrees of freedom

An important part of designing atomistic simulations is determining what the degrees of freedom (DOF) are and how to control them. In the context of the work presented in this dissertation, we will discuss compositional DOF for modeling APBs in Ni_3Al-based alloys and structural DOF for modeling GBs in α-Ti.

2.2.1 Solute distributions in intermetallics

An *intermetallic* is an alloy where the constituent elements form an ordered compound with distinct sublattices for the different species. Figure 2.1 shows the $L1_2$ crystal structure for the Ni_3Al intermetallic, where Ni atoms are on the face-centers and Al atoms are on the corners of a simple cubic lattice. When a third solute species is added to the system, it might preferentially occupy the Al sublattice (as shown for Ti), the Ni sublattice, or a mix of the two. Different occupancies will change the overall stoichiometry of the system, which would affect the results (e.g., energy) of our calculations. As experimental data on solute

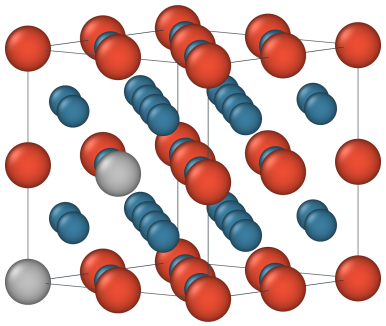


Figure 2.1: $L1_2$ structure of Ni_3Al. Ni atoms (blue) are on the face-centers and Al atoms (red) are on the corners of a cubic cell. The unit cell has been repeated once along each dimension and two of the Al atoms have been replaced by Ti atoms (gray).

site preference (e.g., from atom probe tomography) are limited, below we briefly review the theory underlying the methodology implemented in PyDII (Python framework for Defects In Intermetallics) [46] that we used to calculate solute site preference for arbitrary solutes.

PyDII employs the grand canonical, dilute-solution model (DSM) thermodynamic formalism based on a statistical-mechanical low-temperature expansion following the framework described by Woodward, et al. [47], as shown below. The model produces a set of equations that are solved to predict equilibrium defect concentrations in intermetallics as a function of composition and temperature. Since we are working in a grand canonical ensemble, we can define the relevant thermodynamic potential Ω ("grand potential") [48] as a function of the natural variables μ, V, and T through a Legendre transform of the Helmholtz free energy. The partition function associated with the Ω potential is

$$Z = \sum_{\sigma} \exp\left[-\left(E^{\sigma} - \sum_{i} \mu_i N_i^{\sigma} \right) \Big/ k_{\mathrm{B}} T \right] \tag{2.2}$$

where σ is an arrangement of atoms over the N lattice sites.

At this point, it is convenient to make a few simplifications and definitions for reasons we will see shortly. We can factor out the ground-state energy from Equation 2.2 by rewriting $E^{\sigma} = E^0 + \delta E^{\sigma}$ and $N_i^{\sigma} = N_i^0 + \delta N_i^{\sigma}$, where the superscript '0' indicates the energy and number of atoms on site i in the reference ground state structure and δ represents the difference with respect to that state for configuration σ. Now Equation 2.2 becomes

$$
\begin{aligned}
Z &= \sum_{\sigma} \exp\left[-\left((E^0 + \delta E^\sigma) - \sum_i \mu_i (N_i^0 + \delta N_i^\sigma) \right) \Big/ k_\mathrm{B} T \right] \\
&= \exp\left[-\left(E^0 - \sum_i \mu_i N_i^0 \right) \Big/ k_\mathrm{B} T \right] \sum_{\sigma} \exp\left[-\left(\delta E^\sigma - \sum_i \mu_i \delta N_i^\sigma \right) \Big/ k_\mathrm{B} T \right] \\
&= \exp\left[-\left(E^0 - \sum_i \mu_i N_i^0 \right) \Big/ k_\mathrm{B} T \right] \left(1 + \sum_{\sigma'} \exp\left[-\left(\delta E^{\sigma'} - \sum_i \mu_i \delta N_i^{\sigma'} \right) \Big/ k_\mathrm{B} T \right] \right)
\end{aligned}
\tag{2.3}
$$

where σ' contains all arrangements except the ground-state (i.e., all possible point defects). In addition, to simplify the counting of atoms and sites, we can introduce concentration variables $c_i(p)$ to denote the atomic fraction of species i at site p in an ordered intermetallic. Here, p is either the Ni or Al sublattice and i can be Ni, Al, or the solute species. We can also define $\lambda(p)$ to denote the multiplicity of each site p in the computational cell, which allows us to write $N_i = \sum_p \lambda(p) c_i(p)$.

Now through the conventional relation $\Omega = -k_\mathrm{B} T \ln Z$, where k_B is the Boltzmann constant and T is temperature, we can plug in Equation 2.3 and take the first-order low-temperature expansion of Ω, where we specifically assume $k_\mathrm{B} T \ll \delta E^\sigma - \sum_i \mu_i \delta N_i^\sigma$ in the exponential. This gives us (assuming zero stress):

$$
\begin{aligned}
&\Omega = -k_\mathrm{B} T \ln Z \\
&= -k_\mathrm{B} T \ln \left(\exp\left[-\left(E^0 - \sum_i \mu_i N_i^0 \right) \Big/ k_\mathrm{B} T \right] \right) - \\
&k_\mathrm{B} T \ln \left(1 + \sum_{\sigma'} \exp\left[-\left(\delta E^{\sigma'} - \sum_i \mu_i \delta N_i^{\sigma'} \right) \Big/ k_\mathrm{B} T \right] \right) \\
&\approx \left(E^0 - \sum_i \mu_i N_i^0 \right) - k_\mathrm{B} T \sum_{\sigma'} \exp\left[-\left(\delta E^{\sigma'} - \sum_i \mu_i \delta N_i^{\sigma'} \right) \Big/ k_\mathrm{B} T \right] \\
&= E^0 - \sum_i \mu_i \sum_p \lambda(p) c_i^0(p) - k_\mathrm{B} T \sum_p \lambda(p) \sum_{\epsilon} \exp\left(-\left[\delta E^\epsilon(p) - \sum_i \mu_i \delta c_i^\epsilon(p) \right] \Big/ k_\mathrm{B} T \right)
\end{aligned}
\tag{2.4}
$$

In Equation 2.4, E^0 and $c_i^0(p)$ are the energy and concentrations, respectively, in the ground-state configuration, which corresponds to binary, stoichiometric Ni_3Al. μ_i is the chemical potential of species i. $\delta E^\epsilon(p)$ and $\delta c_i^\epsilon(p)$ are the defect "excitation energy" and change in site concentration, respectively, associated with introducing a substitutional or vacancy point defect at site p. Note that in going from the second to the third line we used

the first-order Taylor series approximation $\ln(1+x) \approx x$, but higher-order terms can be included to reflect defect-defect interactions [49].

Equation 2.4 can be differentiated with respect to μ_i to derive an equation for the average concentration:

$$\begin{aligned}\langle c_i(p)\rangle &= -\frac{\partial\Omega}{\partial\mu_i}(p) \\ &= c_i^0(p) + \sum_{\epsilon} \delta c_i^{\epsilon}(p) \times \exp\left(-\left[\delta E^{\epsilon}(p) - \sum_j \mu_j \delta c_j^{\epsilon}(p)\right] \Big/ k_{\mathrm{B}}T\right)\end{aligned} \tag{2.5}$$

where $\langle c_i(p)\rangle$ denotes the ensemble-averaged concentration of species i at site p. For a system with n species and a fixed T, we can fix the values for the chemical potentials by specifying the $n-1$ relative concentrations. Combining Equation 2.4 and 2.5 and the condition of thermodynamic equilibrium with respect to vacancies [47] produces a non-linear system of n equations and n unknown μ_i's that can be solved at a specified T to obtain a set of chemical potentials, $\{\mu_i\}$. The $\{\mu_i\}$ are then used to determine sublattice concentrations for each species as a function of the Al concentration. While the PyDII code [50] was already implemented with this functionality, I have extended it to properly handle cases where the multiplicity of sites ($\lambda(p)$) is greater than one, as is the case for Ni_3Al.

In addition to sublattice occupancy, another consideration for the solute species is ordering on the particular sublattice(s). For example, Ti atoms on the Al sublattice could prefer to be arranged as first nearest neighbors, second nearest neighbors (as shown in Figure 2.1), etc., and this ordering will further lead to energy differences, especially when a defect like an APB is introduced to the system. While this ordering could be investigated using a technique like *ab initio*-based Monte Carlo (MC) simulations, we adopt a simpler approach, where we assume the solute species will be ideally randomly distributed on the appropriate sublattice using a special quasirandom structures (SQS) approach [51]. Using the SQS algorithm as implemented in the Alloy Theoretic Automated Toolkit (ATAT) [52, 53] allows us to use simpler pair and triplet correlation functions (attempting to match those in an ideally random ordering) to populate the atomic sites.

2.2.2 GB structure optimization

As mentioned in the Introduction, GBs have five macroscopic DOF and many more microscopic structural DOF. Even among relatively simple and common STGBs, allowing the density of atoms to vary at the interface has lead to the discovery of multiple GB phases with different properties [39,54]. This grand canonical sampling is omitted by the traditional γ-surface method, which takes two bulk perfect crystals and only applies relative translations followed by a static relaxation at 0 K, and does not always produce the ground-state structure. Therefore, it is critical for a GB structure search algorithm to sample different

atomic densities in the GB, and this has been done in the past using high-temperature MD simulations [39, 55], MC sampling [56–59], and evolutionary algorithms (EA) [59–64].

Each of these methods has its respective advantages and trade-offs. High-temperature MD has the potential to discover entropy-stabilized phases that are normally unstable at 0 K [62], although the simulation parameters have to be tuned to the particular system and it may require long simulation times. MC sampling takes a more data-efficient approach, where successive structures are biased by a Metropolis criterion and the swaps may perturb the system far away from the equilibrium in MD to discover new structures. EA uses a similar framework where structures are generated in groups ("generations") and then evolved over the course of the simulation by a fitness metric such as E_{gb}. Between each generation, EA adapts ideas from evolutionary biology by taking the existing ("parent") structures and creating successive ("child") structures through a series of operations including mutation (e.g., atom deletion and displacement), crossover (e.g., one half of each parent is spliced together), and selection based on fitness.

Previously, we have used the EA implemented in the USPEX code [65, 66] to optimize the $\{11\bar{2}4\}$ TB in α-Ti to discover competing GB phases [67]. The ground-state structure closely matched experimental observations using high-resolution transmission electron microscopy (HRTEM) [68] and was a strained polymorph of a metastable bulk body-centered orthorhombic (BCO) phase [69]. While this was a successful application of EA to GB structure search, it remains unclear if existing tools can handle more challenging systems such as low-angle twist GBs where the periodic area is large. Moreover, as all of the published tools are either private or no longer maintained, the community lacks a robust, open-source tool that can systematically explore the configuration space of GBs.

2.2.3 GRand canonical Interface Predictor (GRIP)

The results presented in Chapter 4 are from our open-source GRand canonical Interface Predictor (GRIP) tool for GB structure search, which rigorously explores structural DOF through grand canonical sampling and MD simulations. Bicrystal slabs can be automatically generated in the code using the Atomic Simulation Environment (ASE) [70] library in Python, or supplied as external files. The simulation cell is oriented such that the y-axis is the tilt axis direction, the x-axis is the orthogonal in-plane direction, and the z-axis is the out-of-plane normal direction (see Figure 2.2). The cell is periodic in the GB plane (xy-plane) and has non-periodic boundaries in the z direction that are sufficiently far apart (default ≥ 7 nm) to minimize cell size effects. In addition to the slabs supplied in POSCAR format, all user settings are contained in a `params.yaml` file for ease of customization.

For an individual STGB, each iteration of the algorithm has two stages. In the first stage, the algorithm randomly samples an $m \times n$ replication of the unit GB cell, translates the upper slab in the xy-plane, and removes atoms from the GB (i.e., creates vacancies). Then it performs a finite-temperature MD simulation using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [77] in the microcanonical (NVE) ensemble with a Langevin thermostat and a time step of 2 fs. The temperature and duration of the MD

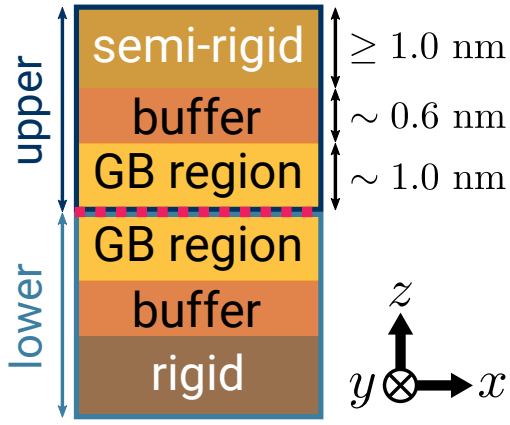


Figure 2.2: Simulation cell setup for GRIP. The cell is oriented such that the y-axis is the tilt axis direction, the x-axis is the orthogonal in-plane direction, and the z-axis is the out-of-plane normal direction. The upper and lower bulk crystals are joined with periodic boundary conditions in the GB plane (xy-plane). We translate the upper crystal, remove atoms from the GB (magenta line), and perform MD simulations. During MD, atoms in the GB regions are free to move while atoms in the buffer and semi-rigid regions in the upper slab are constrained to move together, and atoms in the lower two regions are fixed. During relaxation, atoms in both buffer regions are free to move while the semi-rigid region is still constrained. For the γ-surface method, the upper slab is only allowed to translate as a whole before relaxation is applied.

are also randomly sampled based on the user-specified ranges and only the atoms in the "GB region" are allowed to move freely during this part. Finally, the temperature is quickly ramped down to 100 K for 1000 time steps. There is an additional user-specified probability where the algorithm skips the MD subroutine for one iteration.

In the second part, each configuration is fully relaxed at 0 K using a conjugate gradient minimization scheme, where atoms in the GB and buffer regions can move freely while the semi-rigid region is constrained to move together. The convergence criteria are 10^{-15} for relative energy and 10^{-15} eV Å^{-1} for forces, with a maximum of 10^5 evaluations for each criterion. The algorithm repeats these stages on each processor independently until termination, saving each relaxed structure to disk and periodically deleting duplicates. Duplicates are defined as structures with the value of E_{gb} (to three decimal places) and n, and the algorithm will keep the structure with a smaller reconstruction and relative translations. In addition to the results for α-Ti presented in Chapter 4, we validate our GRIP tool by reproducing literature results for tilt and twist GBs in elemental cubic metals and more challenging covalently-bonded systems (Figure 2.3).

For each relaxed structure, the GB energy (E_{gb}) is computed according to:

$$E_{\mathrm{gb}} = \frac{E^{\mathrm{gb}}_{\mathrm{total}} - N^{\mathrm{gb}}_{\mathrm{total}} E^{\mathrm{bulk}}_{\mathrm{coh}}}{A^{\mathrm{gb}}_{\mathrm{plane}}} \tag{2.6}$$

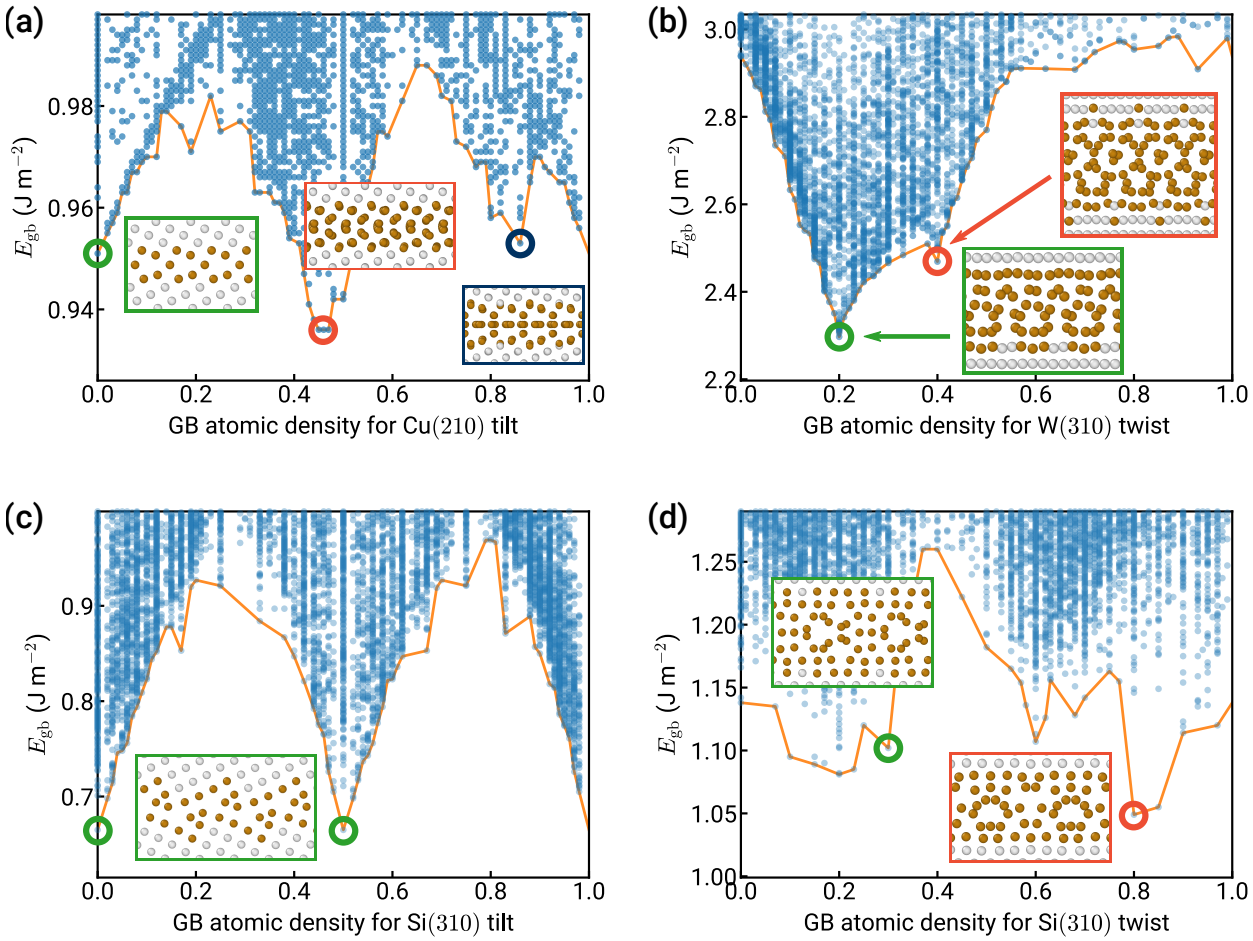


Figure 2.3: Validation of the GRIP tool. We perform grand canonical optimization of the (a) Σ5(210)[001] tilt GB in Cu (EAM potential [71]), (b) Σ5(310)[001] twist GB in W (EAM [72]), (c) Σ5(310)[001] tilt GB in Si (Stillinger-Weber [73]), and (d) Σ5(310)[001] twist GB in Si. The energy plots and low-energy structures match those in the literature for Cu [39], W [74], and Si [57,75]. The atoms are colored according to CNA [76], where gray are bulk-coordinated atoms and brown are non-bulk-coordinated atoms (in the GB).

where $E^{\text{gb}}_{\text{total}}$ and $N^{\text{gb}}_{\text{total}}$ are the energy and number, respectively, of atoms in the GB region, $E^{\text{bulk}}_{\text{coh}}$ is the cohesive energy per atom in bulk α-Ti, and $A^{\text{gb}}_{\text{plane}}$ is the area of the GB plane. These quantities are illustrated in Figure 2.4 and exploit the ability to obtain per-atom energies in LAMMPS to simplify the calculation, i.e., we can ignore the free surfaces in the simulation cell.

As with previous studies [39,62], we track the number of atoms in the GB by calculating the fraction of atoms remaining in one plane ("GB atomic density"), n, according to:

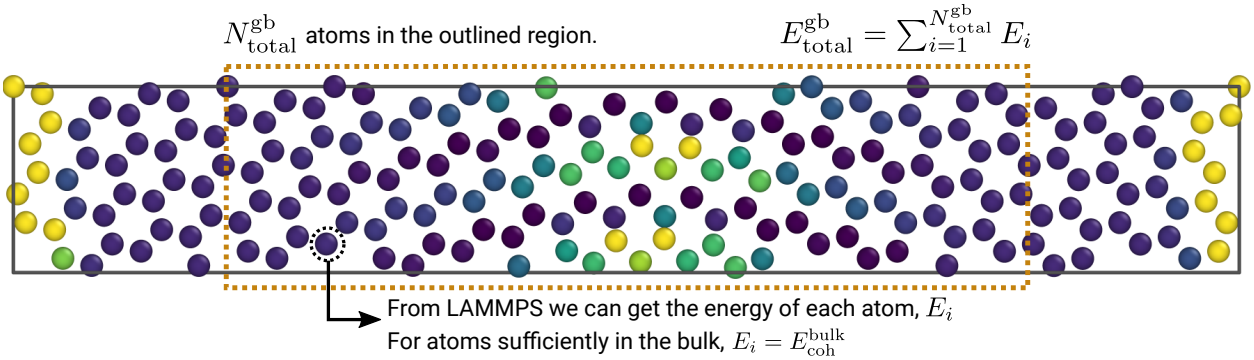


Figure 2.4: How E_{gb} is calculated in this work (Equation 2.6). The excess energy of the GB is the additional energy of atoms in the outlined region compared to if all $N^{\text{gb}}_{\text{total}}$ atoms were bulk atoms with energy $E^{\text{bulk}}_{\text{coh}}$. The atoms are colored according to their energy E_i.

$$n = \frac{N_{\text{total}} \mod N^{\text{bulk}}_{\text{plane}}}{N^{\text{bulk}}_{\text{plane}}} \in [0, 1) \tag{2.7}$$

where N_{total} is the number of atoms in the simulation cell and $N^{\text{bulk}}_{\text{plane}}$ is the number of atoms in one plane of the bulk structure. Previous calculations of $N^{\text{bulk}}_{\text{plane}}$ simply counted the number of atoms at a single z value in the bulk; however, due to the 2-atom basis of the HCP crystal structure, atoms associated with one plane may be offset in the z-direction (see Figure 2.5). Therefore, we calculate $N^{\text{bulk}}_{\text{plane}}$ as the number of atoms within a region equal to the minimum

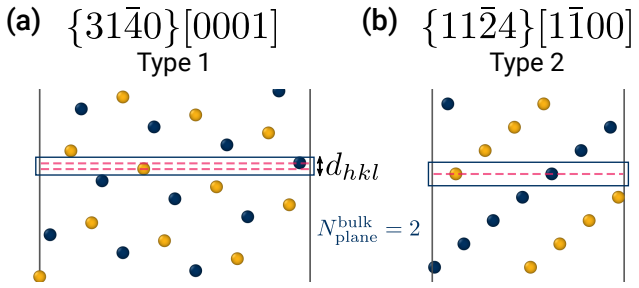


Figure 2.5: How $N^{\text{bulk}}_{\text{plane}}$ is calculated in HCP crystals. (a) In some STGBs like $\{31\bar{4}0\}[0001]$, the two basis atoms (distinguished as blue and gold) are offset (emphasized by the dashed magenta lines) and the calculation includes both of these atoms by counting all atoms within a region of thickness equal to the smallest normal component of a lattice vector, which is equivalent to the interplanar spacing of the hexagonal lattice (d_{hkl}). (b) In other STGBs like $\{11\bar{2}4\}[1\bar{1}00]$, both basis atoms have the same z-coordinate and adjacent planes are still spaced by d_{hkl}. We refer to these boundaries as "Type 1" and "Type 2," respectively.

normal component of a lattice vector; in the case of α-Ti, this is equivalent to the interplanar spacing of the hexagonal lattice (d_{hkl}) given by [78]:

$$\frac{1}{d_{hkl}^2} = \frac{4}{3}\left(\frac{h^2 + hk + k^2}{a^2}\right) + \left(\frac{l}{c}\right)^2 \tag{2.8}$$

where h, k, and l are the Miller indices, and a and c are the HCP lattice constants. We note this extended definition of $N_{\text{plane}}^{\text{bulk}}$ reduces to taking a planar slice for unary, single-basis systems like elemental BCC and FCC metals, consistent with previous studies [39, 62].

The GRIP tool is written in Python with minimal dependencies (NumPy [79], ASE [70], LAMMPS [77]) and in a modular fashion that decouples DOF sampling and energy minimization. Therefore, it is straightforward to plug in alternative approaches such as MC sampling and *ab initio* codes as future extensions. The open-source code will be publicly accessible at `https://github.com/enze-chen/grip`. To the best of our knowledge at the time of writing, GRIP is the only publicly available code for grand canonical optimization of GB structures.

2.3 Jupyter ecosystem

While there exists immense interest from students, instructors, employers, and other stakeholders to incorporate more computing and data science (DS) into MSE classrooms [22, 25], the path to adoption presents multiple barriers that require careful consideration to overcome [26]. Much of the discussion has centered around what should be the relevant topics to cover and how the examples/applications can be germane to MSE to increase student engagement; however, there are also infrastructure considerations, as many of the research-grade tools may not transfer as easily into classroom settings to be pedagogically useful. We hear too often about students getting stuck at the initial stages of software configuration/installation, frustrating their experience and often preventing further progress.

Borrowing ideas from DS education and interactive computing, instructors in chemistry and MSE have started using Jupyter notebooks [80] to create new chemistry/materials informatics curricula [81–83]. These digital notebooks merge text, code, graphics, and additional multimedia elements into rich computational narratives that facilitate student learning. Their popularity in scientific research for data analysis and presentation makes for a smoother transition for instructional use. Moreover, the interactivity enabled by Jupyter notebooks allows students to be active participants in the learning process, achieving the highest level of the ICAP (Interactive, Constructive, Active, Passive) framework by Chi and Wiley [84]. The ICAP framework for active learning provides specific definitions of engagement activities in a strict hierarchy—that is, as activities move from the passive end of the spectrum to interactive, students will undergo different knowledge-change processes that lead to deeper learning. As active learning is known to increase student performance in STEM [85], particularly for underrepresented minorities [86], having the ability to engage

MSE students in interactive informatics education will greatly expand the impact of such curricula.

2.3.1 Jupyter Books for digital textbooks

Building on the success of Jupyter notebooks, the Project Jupyter team has recently launched Jupyter Book as a means to compile a set of Jupyter notebooks into a digital textbook [87]. This digital text can then be hosted online for ease of reading and advanced MyST markdown features [88] enable elements like presentation slides, polls, and videos to be embedded in the text. Following the example from UC Berkeley's DATA courses [89], we were the first to use Jupyter Book in the context of MSE, creating an interactive textbook for a materials informatics summer internship. Given all of the chaos during the height of the pandemic, having a central hub (i.e., the Jupyter Book) for all of the learning materials was our top priority for reducing cognitive load [90] and facilitating access to learning MSE and DS in context. Additional details about the curriculum and learning outcomes will be discussed in Chapter 5.

There are two additional strengths of the Jupyter Book framework that are important to point out. One is the interface between the digital text and an online Python kernel such as JupyterHub [91] or Google Colaboratory [92]. This allows students to avoid the hassle of trying to install Python on their personal computers and launch Jupyter notebooks locally (hint: you cannot double click it). Instead, students can work in the cloud using a computing environment provisioned for them, saving instructors work as well. A second benefit of the Jupyter Book framework for instructors is the ability to use common research workflows (e.g., Python, GitHub, command line) to build computational curricula, lowering the barrier for adoption. There are several existing tools for learning computational MSE with documented success (e.g., PhET [93], MaterialSim [94], Molecular Workbench [95]), but modifying them requires a deep knowledge of additional programming languages (e.g., JavaScript) that the majority of instructors will not have prior experience with and may not have the bandwidth to learn. By making it easier for both instructors and students to engage with the curricula, we hope to provide a curriculum design framework for how to integrate computing and DS within MSE education.

Chapter 3

Antiphase boundaries in Ni-based superalloys

The work presented in this chapter was published as a peer-reviewed Article titled "Modeling antiphase boundary energies of Ni3Al-based alloys using automated density functional theory and machine learning" in *npj Computational Materials* **8**, 80 (2022) by Enze Chen, Artur Tamm, Tao Wang, Mario E. Epler, Mark Asta, and Timofey Frolov [96], and is reproduced here with permission of the co-authors.

3.1 Introduction to Ni-based superalloys

Owing to their excellent high-temperature mechanical properties and corrosion resistance, Ni-based superalloys are the materials of choice for several important technologies, including airplane turbines and land-based power generators [4, 97–99]. The microstructure of these alloys consists of a Ni-rich, disordered FCC matrix (γ phase) that envelops coherent Ni_3Al precipitates (γ' phase) which give them their strength [100]. The Ni_3Al precipitates are ordered in the $L1_2$ structure, where Al atoms are on the corners and Ni atoms are on the face centers of a simple cubic lattice. Upon the passage of a matrix $\frac{a}{2}\langle 110\rangle$-type dislocation through the ordered precipitates, high-energy antiphase boundaries (APBs) are created on the $\{111\}$ family of planes to contribute towards γ'-phase strengthening in superalloys [101–103]. One example of where this strengthening mechanism manifests is in the yield strength of the superalloy when dislocations shear the Ni_3Al precipitates. Depending on the microstructure, dislocation pairs can be weakly-coupled or strongly-coupled, and Reppich found that the yield strength (σ_y) reaches a peak at the transition point between the two regimes, where it can be approximated by:

$$\sigma_{y,peak} \propto \frac{\gamma_{APB} f^{1/2}}{2b} \tag{3.1}$$

where f is the volume fraction of the γ' phase, b is the magnitude of the Burgers vector,

and γ_{APB} is the APB energy. Relationships such as Eq. 3.1 reinforce the importance of the APB energy in modulating superalloy properties and justify why it has garnered so much attention in superalloy design.

In addition, Ni-based superalloys have multicomponent chemistries often with ten or more elements [4, 97], and this alloying further improves the properties of the base Ni–Al alloy to enhance high-temperature performance [104–106]. The APB energy is sensitive to alloy composition, which provides a rich design space one can explore when optimizing the composition of alloys to provide higher yield strength and creep resistance. With so many degrees of freedom, however, it would be difficult and prohibitively expensive to rely solely on experimental methods to probe the composition dependence of the APB energy [107,108]. As a result, the community has explored the use of computational methods to understand this correlation and exploit it in alloy design.

Numerous computational investigations into the APB energy in Ni_3Al and Ni_3Al-based alloys have been undertaken over the past two decades using a plethora of different methods. *Ab initio* approaches include density functional theory (DFT) [109–114] and DFT-based methods such as the cluster variation method [115, 116], coherent potential approximation (CPA) [117, 118], and cluster expansion (CE) with Monte Carlo (MC) sampling [119, 120]. In addition, semi-empirical methods include molecular dynamics [121] and Monte Carlo simulations [122] based on classical potentials, and the CALculation of PHAse Diagram (CALPHAD) approach [113, 123]. These studies have contributed greatly to the fundamental understanding of the APB energy in Ni_3Al-based alloys, including its dependence on temperature, spin polarization, composition, and ordering, all of which have implications for alloy design.

Focusing on the composition dependence, an early DFT study by Chandran and Sondhi found a large strengthening effect due to the addition of Ti, Ta, and Nb atoms, where the (111) APB energy (hereafter symbolized as γ_{111}) increased from its intrinsic value of $0.181\,\mathrm{J\,m^{-2}}$ in Ni_3Al to over $0.6\,\mathrm{J\,m^{-2}}$ in the ternary alloys [110]. Kumar et al. performed a more comprehensive DFT study with 16 different ternary solutes, considering a single atom near the APB in their 96-atom supercell for the ternary element [114]. That study also revealed the sensitivity of γ_{111} to sublattice occupancy of solute species in Ni_3Al, which must be precisely determined in order to obtain accurate calculations for γ_{111}. Crudden et al. used a combination of DFT and CALPHAD to calculate γ_{111} in several ternary Ni_3Al-based alloys with refractory metal solutes [113], and they found compositional dependencies that reasonably agreed with previous studies of the APB energy and yield strength in these alloys. The use of CALPHAD methods for calculating the APB energy is possible through its connection to the ordering energy (E_{ord}), as detailed by Miodownik and Saunders [123] and leveraged in the work by Crudden et al [113]. Intuitively, a correlation between these two quantities can be understood as the creation of an APB in the $L1_2$-ordered Ni_3Al precipitates forms an interfacial defect that disrupts the preferred ordered atomic arrangement in the vicinity of the APB. The relationship was also studied by Gorbatov et al., who used the CPA method to discover a strong correlation between γ_{111} and E_{ord} in ternary Ni_3Al-based alloys with $3d$ transition metal solutes at low concentrations (2.5 at.%) [118]. The correlation

between γ_{111} and E_{ord} has been extensively discussed in the community and is particularly attractive as it enables one to use materials thermodynamic databases, which are extensively developed for multicomponent Ni-based superalloy systems, to model the APB energy.

Building on the previous studies reviewed above, our goal is to further expand the predictive capabilities of APB energy models by leveraging machine learning (ML) algorithms to automatically learn the sophisticated relationships between structural and energetic properties and the APB energy from data. ML allows us to combine the ordering energy with other physically-meaningful features (derived from materials properties) as inputs into a flexible model which has the potential to predict the APB energy with greater accuracy and across a wider range of superalloy chemistries. The wealth of literature on the APB energy in Ni_3Al-based alloys suggests that having a transferable, fast, and predictive model can be impactful for alloy design in screening novel compositions. For this purpose, we require a training dataset that extends across a wider range of compositions than previously considered so that the ML model can learn the appropriate relationships for more exploratory designs with less data bias [124].

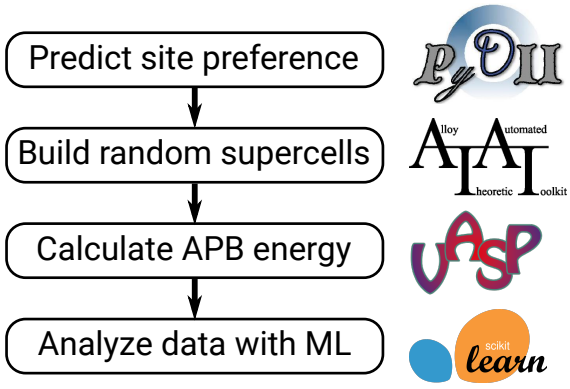


Figure 3.1: An outline of our computational workflow. For each ternary species, we first compute its sublattice preference using PyDII, and then use this prediction to construct model supercells using ATAT. Next we use VASP to perform DFT calculations to obtain γ_{111}. Finally, we use ML techniques implemented in scikit-learn to analyze correlations in the data and build predictive ML models for γ_{111}.

In the present work, we create an automated workflow to calculate APB energies at scale, leveraging recent advancements in high-throughput calculations for alloy design [14]. Using this workflow (Figure 3.1; detailed in Methods), we analyze the composition dependence of γ_{111} in 111 model ternary Ni_3Al-based alloys, generating significantly more DFT data than what are currently present in the literature. We present the results and compare them with previous studies to validate known behavior and uncover insights across a wide range of ternary elements. In particular, we employ ML to analyze feature correlations and build predictive models for the APB energy that can be used to rapidly screen alloy compositions at much lower computational cost. Finally, we discuss how the infusion of scientific domain

knowledge into data-driven methodologies can enhance superalloy design going forward.

3.2 Methods

We outlined our computational workflow in Figure 3.1 and elaborate on the details in the following subsections.

3.2.1 Solute site preference

The solute site preference was computed using PyDII [46] using the methodology described in Chapter 2. For Ni_3Al, the fraction of a species i on the Al sublattice can then be computed using the following equation:

$$f_i(\mathrm{Al}) = \frac{\langle c_i(\mathrm{Al})\rangle}{\langle c_i(\mathrm{Al})\rangle + 3\langle c_i(\mathrm{Ni})\rangle} \tag{3.2}$$

where $\langle c_i(X)\rangle$ is the average concentration of species i on each site of the X sublattice.

The PyDII code [50] is used to generate the default structures and settings for the DFT calculations of the excitation energies using the Vienna Ab initio Simulation Package (VASP) [34–37] with the projector augmented-wave (PAW) method [32, 33] and the generalized gradient approximation of Perdew, Burke, and Ernzerhof (GGA-PBE) [31]. The default VASP settings include a fixed plane wave energy cutoff of 520 eV, 5000 **k**-points per reciprocal atom, spin polarization, ionic relaxation, and convergence criteria for electronic self-consistency and ionic relaxation of 10^{-6} eV and 10^{-3} eV, respectively.

We then use the DFT-calculated defect excitation energies to calculate the equilibrium sublattice concentrations at $T = 1000$ K and a fixed concentration of 1 at.% for the ternary species. The sublattice preference calculated from PyDII is then used to guide supercell construction for model ternary Ni_3Al-based alloys according to the following distributions for the ternary species:

I. Solutes with strong Ni sublattice preference are distributed entirely on the Ni sublattice at all concentrations. The resulting alloy has composition $Ni_{0.75-x}Al_{0.25}Z_x$.

II. Solutes with strong Al sublattice preference are distributed entirely on the Al sublattice at all concentrations. The resulting alloy has composition $Ni_{0.75}Al_{0.25-x}Z_x$.

III. The remaining solutes with composition-dependent site preference are distributed such that 3/4 of the solute atoms are on the Ni sublattice and 1/4 of the atoms are on the Al sublattice. The resulting alloy has composition $Ni_{0.75(1-x)}Al_{0.25(1-x)}Z_x$.

For each composition, we randomize the placement of atoms on the appropriate sublattices using special quasirandom structures (SQSs) [51] generated using the Alloy Theoretic Automated Toolkit (ATAT) [52, 53]. We optimize for pair and triplet correlations out to

0.55 nm, which captures the first few nearest-neighbor correlations for each configuration. The maximum absolute value of the correlation difference between the SQS and ideally random is 0.017. We emphasize that these solute distributions are simply chosen to create model supercells for our high-throughput workflow. The species distribution in the γ' phase in a real superalloy will differ depending on the temperature, (relative) chemical potentials of the species, processing techniques, etc. Despite this simplification, we believe our method will still capture the dominant behaviors and trends in site preference—and, correspondingly, the APB energy—which we validate against other studies in Results. We find that the sensitivity of the APB energy to local ordering (e.g., using a different SQS) results in differences by as much as 0.01 J m^{-2} in γ_{111}, but should not affect our conclusions.

3.2.2 DFT calculations for γ_{111}

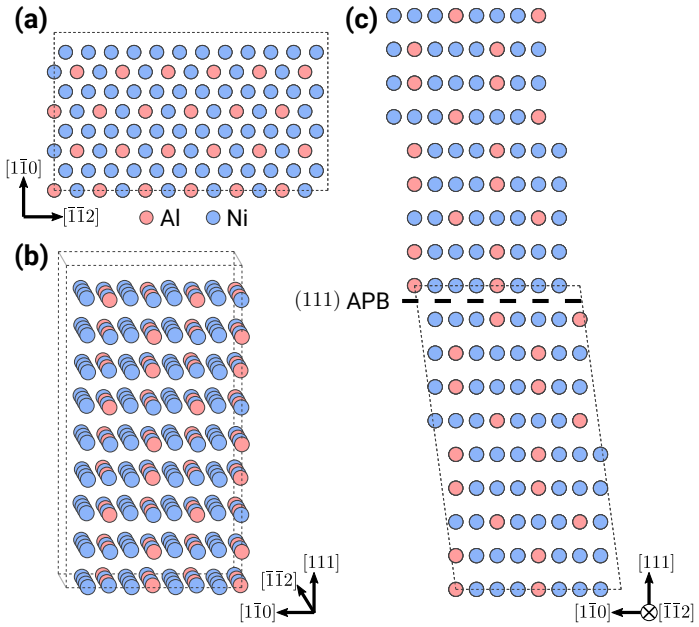


Figure 3.2: Schematics of our 288-atom DFT supercell. The Al and Ni sites are red and blue, respectively, and they are populated with the ternary species based on the predictions from PyDII followed by randomization with SQS. (a) Top view of the (111) plane. (b) Bulk cell without an APB. (c) A single (111) APB is created near the top of the supercell by tilting the cell by $\mathbf{b} = a/2[1\bar{1}0]$. The cell is projected along the $[\bar{1}\bar{1}2]$ direction and has been repeated once to show the APB (note the new Al–Al and Ni–Ni bonds).

We perform DFT calculations for γ_{111} using VASP with the PAW method and the GGA-PBE functional. For these calculations, we set the plane wave cutoff energy for each calculation to be at least 30 % greater than the highest cutoff energy (ENMAX in VASP) specified in the pseudopotential files (ENCUT is at least 350 eV). Spin-polarized calculations are

performed to account for the effects of local magnetic moments on the APB energy [109]. Calculations with spin polarization appear to give closer agreement to literature values as non-spin-polarized calculations result in values for γ_{111} that are approximately 30 % lower. Consistent with previous reports [109], we find persistent local moments of 0.2 μ_B-0.3 μ_B for Ni and different values for different solute species.

For all APB energy calculations, we use a 288-atom computational cell with the x-, y-, and z-axes oriented along the $[\bar{1}\bar{1}2]$, $[1\bar{1}0]$, and [111] directions, respectively (Figure 3.2). This corresponds to a $2 \times 2 \times 3$ $L1_2$ supercell and nine (111) planes separating the APBs, which we deem sufficient based on convergence tests (see Appendix Figure A.2). We use Γ-centered Monkhorst-Pack **k**-point grids [125] of $2 \times 3 \times 2$, corresponding to 5000 **k**-points per reciprocal atom, based on a test of energy convergence for $L1_2$. We apply a first-order Methfessel-Paxton smearing method [126] with a smearing width of $\sigma = 0.1$ eV. Structural relaxations are performed using a conjugate gradient algorithm in two steps: First, only the volume is allowed to relax for the bulk structure to capture the lattice expansion due to the solute. Then, the bulk structure is copied and tilted such that the top face translates by $\mathbf{b} = a/2[1\bar{1}0]$ (Figure 3.2c) to create a structure with a single (111) APB. Independent ionic relaxations are then performed on the bulk structure (without an APB) and the second structure with an APB. The convergence criteria for the electronic self-consistency and ionic relaxation loops are 10^{-5} eV and 10^{-3} eV, respectively.

For binary Ni_3Al, our DFT calculations give an equilibrium lattice constant of approximately 0.3565 nm, which agrees well with reported computational (e.g., refs. [114, 121]) and experimental values (0.357 nm) [4]. We find that γ_{111} for stoichiometric Ni_3Al is 0.188 J m^{-2}, and we observe a roughly linear increase in γ_{111} with increasing Al concentration (Supplementary Table 1), which are consistent with previous experimental [127, 128] and computational (e.g., refs. [109, 110, 118]) studies. We also observe a variability in γ_{111} depending on the species arrangements around the APB for a fixed composition. To account for these differences, we shift the (111) planes in the tilted supercell to create five different APBs to help quantify the statistical scatter. The energy of a single (111) APB, denoted by $\gamma_{111}^{(i)}$, is calculated according to Equation 3.3:

$$\gamma_{111}^{(i)} = \frac{E_{\text{apb}}^{(i)} - E_{\text{bulk}}}{A_{111}} \tag{3.3}$$

where $E_{\text{apb}}^{(i)}$ is the total energy of the ith structure with an APB, E_{bulk} is the total energy of the non-tilted supercell, and A_{111} is the area of the APB (Figure 3.2a). The mean and standard deviation are calculated for five different APB energies to obtain γ_{111} for each composition. We do this for each ternary alloy system at three different solute concentrations: low (1.39 at.%), medium (5.56 at.%), and high (9.72 at.%). We note that while these DFT calculations at 0 K do not account for the configurational entropy contributions to the free energy [115] nor the decrease in APB energy at high temperatures (> 1000 K) [119], they can still provide quantitatively accurate data for room-temperature properties [118] that can be used to build predictive models for screening new compositions.

To calculate the ordering energy E_{ord}, we perform additional DFT calculations using smaller, 80-atom supercells of similar compositions and a **k**-point grid of the same density as before. E_{ord} is the difference in energy between two structures: one with atoms on their preferred sublattice (which we determine using PyDII) and another with the same composition with atoms randomly distributed on all sites. In this case, all degrees of freedom are allowed to relax, but all other DFT settings remain the same as those for calculating γ_{111}. E_{ord} in units of eV atom^{-1} is then given by Equation 3.4:

$$E_{\text{ord}} = \frac{E_{\text{L1}_2} - E_{\text{random}}}{N} \tag{3.4}$$

where E_{L1_2} and E_{random} are the total energies of the structures with correct and random sublattice distributions, respectively, and N is the total number of atoms. Equation 3.4 is used to obtain the ordering energy for all ternary Ni_3Al-based alloys except for 9.72 at.% Ce whose calculations did not converge. All of the DFT data for APB energies and ordering energies are shared publicly on MPContribs [129].

3.2.3 Machine learning

We generate physically-motivated features from the alloy composition using the matminer Python package [130], as a similar approach by Ling et al. was shown to be effective for featurizing superalloy chemistries [131]. Briefly, the matminer featurizer creates an equal-length feature vector for every composition by calculating statistical quantities (we use the composition-weighted mean and standard deviation) over a set of elemental properties [132] of the alloying elements. To this elemental feature vector from matminer we also include several more derived features that we believe to be influential in modeling the APB energy, including the DFT-calculated ordering energy (E_{ord}), density, change in lattice parameter, and composition-weighted Pettifor number [133], among others (see Supplementary Table A.1). We generate a total of 70 features and remove all features with at least one undefined value and those that have a Spearman's rank correlation [134] greater than 0.9 with another feature.

We train a random forest (RF) regression model, as implemented in the scikit-learn Python package [135], to predict γ_{111} using the root-mean-square error (RMSE) as the error metric. The model is trained with 100 decision trees grown to full depth and with a third of the features considered at each split. To further downselect the features for predictive modeling, we use recursive feature elimination to recursively remove the least important feature from our model based on the feature importance metric in random forests [136]. The RF model's performance is evaluated on the full dataset using grouped 5-fold cross-validation (CV) [137], where the data are grouped by ternary species. This means that four of the folds are used as training data and the final fold is used for validation, while alloys with the same ternary species will always appear in the same fold. This training-validation procedure is repeated such that each of the five folds is used once as validation data and the model's overall performance (RMSE) is computed as the average of all five RMSE scores from CV.

The grouping process helps ensure that our ML model is extrapolating to new regions of chemical space and calculating more faithful validation metrics that are representative of novel alloy compositions encountered during screening.

3.3 Results

3.3.1 Solute site preference

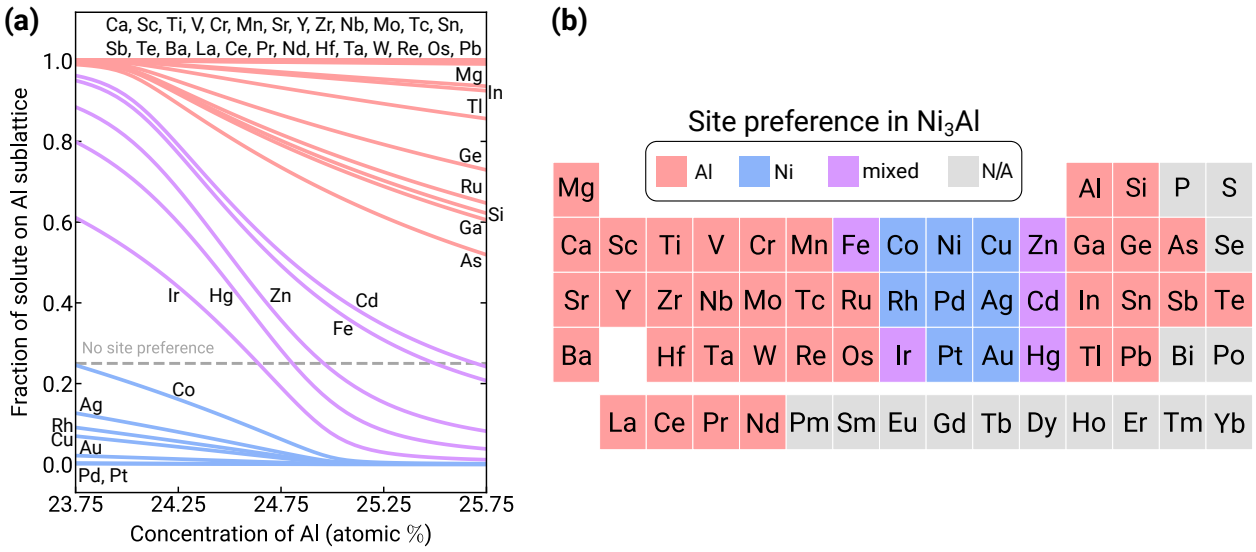


Figure 3.3: Site preference of ternary solute Z in $\mathbf{Ni}_{0.99-x}\mathbf{Al}_x Z_{0.01}$. (a) The calculations are performed at $T = 1000\,\mathrm{K}$, and the fractional site occupancy f (Eq. 3.2) is plotted as a function of Al concentration. The curves are labeled with the ternary element and the group of elements at the top (bottom) all have overlapping profiles at $f = 1.0$ ($f = 0.0$), indicating a strong sublattice preference. Based on the composition dependence in the range shown here, we group the elements into three categories for sublattice preference: Ni preference (blue), Al preference (red), and mixed preference (purple). The horizontal dashed line at $f^* = 0.25$ indicates no site preference. (b) An alternate representation of the solute site preference as a heatmap of the periodic table. A majority of the elements prefer the Al sublattice (red), except for a few near group X that prefer Ni (blue) or exhibit mixed preference (purple). Elements in gray are not tested.

Previous works have demonstrated the sensitivity of the APB energy [114] and mechanical properties [138] of Ni_3Al-based alloys on the solute site preference, i.e., the preference for the solute to reside on the Al or Ni sublattices of the $L1_2$ structure. Therefore, we first determine the site preference of ternary species in Ni_3Al using the DFT-based dilute-solution thermodynamic framework implemented in PyDII [46,50] (see Chapter 2 for details),

which has been previously used to predict the solute site preference in intermetallics such as NiAl [46] and NiTi [139]. From our DFT calculations and analysis with PyDII, we compute the fractional site occupancy on the Al sublattice ($f_i(\mathrm{Al}) \equiv f$) of 46 ternary elements in Ni_3Al as defined by Equation 3.2.

We perform the calculations at $T = 1000\,\mathrm{K}$ and obtain the results shown in Figure 3.3, where f is plotted as a function of the overall mole fraction (concentration) of Al in the Ni_3Al compound. The horizontal dashed line indicates no site preference and corresponds to $f^* = 0.25$, when atomic species Z randomly occupies the two sublattices independent of the Al concentration. By comparing the behavior of f with the threshold f^* in the composition range shown in Fig. 3.3a, we classify the sublattice preference of the ternary species into the following three categories:

I. If $f < f^*$ across the entire composition range (blue curves in Fig. 3.3a), then that solute prefers the Ni sublattice. For the model ternary supercells used to compute the APB energies (see below), these solutes are distributed entirely on the Ni sublattice at all concentrations. The resulting alloy has composition $\mathrm{Ni}_{0.75-x}\mathrm{Al}_{0.25}Z_x$.

II. If $f > f^*$ across the entire composition range (red curves), then that solute prefers the Al sublattice. For the APB supercell calculations, these solutes are distributed entirely on the Al sublattice at all concentrations, resulting in a model ternary alloy with composition $\mathrm{Ni}_{0.75}\mathrm{Al}_{0.25-x}Z_x$.

III. Otherwise, we classify the solute as having a mixed sublattice preference (purple curves). These species are distributed such that 3/4 of the solute atoms are on the Ni sublattice and 1/4 of the atoms are on the Al sublattice. The resulting alloy has composition $\mathrm{Ni}_{0.75(1-x)}\mathrm{Al}_{0.25(1-x)}Z_x$.

By automating these calculations and using the above classification system, we identify patterns across the entire periodic table (Figure 3.3b). We observe that most elements in groups IX, X, and XI strongly prefer the Ni sublattice (type I) when alloyed in Ni_3Al in dilute quantities. The exception is Ir, which—along with Fe and three group XII elements (Zn, Cd, Hg)—is a type III solute whose occupancy on the Al sublattice is strongly composition dependent, and these elements are adjacent to the type I elements in the periodic table. The vast majority of the elements prefer the Al sublattice (type II), although that preference is weaker among the p-block post-transition metals. These results may be rationalized by the larger size of Al atoms compared to Ni atoms (metallic radii of 143 pm and 124 pm, respectively) [4, 140], causing the vast majority of the elements to prefer the Al sublattice. The exception for elements near group X may be due to favorable d-orbital hybridization with the p orbitals in Al, which is a prominent interaction in Ni_3Al [141]. Additionally, we note that PyDII predicts most of the common alloying elements in Ni-based superalloys [4,97] to strongly prefer the Al sublattice in Ni_3Al, at least in the dilute regime in Figure 3.3.

Our approach using PyDII to calculate site preference of ternary solutes in Ni_3Al can be validated by comparing these results to previous studies [142–147]. In a similar study where

first-principles calculations were used to obtain defect formation enthalpies to parametrize a Wagner-Schottky model, Jiang and Gleeson [144] predicted the fractional sublattice occupancy in Ni_3Al at 1273 K and produced concentration profiles that are consistent with those in Figure 3.3a. They also found a majority of their transition metal solutes to strongly prefer the Al sublattice ($f \approx 1$ across the whole composition range) while elements like Ir and Fe had a strong composition-dependent site preference in their model. They calculated a stronger composition-dependent site preference for Co and Mn (PyDII predicts Ni and Al sublattice preference, respectively), but other solute profiles were consistent with ours in the composition range we considered. Liu, et al. used DFT calculations to parametrize the formation energies in a dilute-solution model and also found a strong Al site preference for most transition metal elements [146], which they attributed to an atomic size effect when the solute radius exceeds that of Ni by more than 15 %. When the size mismatch is below this threshold, they found through analysis of the differential charge density that the Al site preference for a solute Z may be correlated to the stronger Ni–Z interactions from d-band electrons compared to Al–Z interactions.

3.3.2 APB energy

Table A.2 in the Appendix shows the DFT-calculated values for γ_{111} in model ternary Ni_3Al-based alloys for 37 ternary species at 1.39 at.%, 5.56 at.%, and 9.72 at.% (111 compositions in total). These elements are chosen for their known contributions to high-temperature strength, oxidation resistance, and creep resistance [4, 97], including a few rare-earth metals [148], or otherwise to span a diverse range of compositions for our data-driven approach. The locations (i.e. site preference) of the ternary species in the supercells are determined using the predictions from PyDII and then randomized over that sublattice using special quasirandom structures (SQSs) [51] generated using the software developed by van de Walle, et al. [52, 53]. We find that an alloy with 9.72 at.% Ta has the highest APB energy of 0.464 J m^{-2}, which exceeds the intrinsic (111) APB energy for Ni_3Al by nearly 0.3 J m^{-2}. The strong influence of Ta on increasing γ_{111} is corroborated by several prior studies [110,112,113]. Notably, we find the highest value for γ_{111} at the medium solute concentration (5.56 at.%) is not for Ta (0.353 J m^{-2}), but rather for W (0.369 J m^{-2}), which appears to give rise to a strong increase in γ_{111} at lower concentrations. Our calculations show that alloying with Re produces the highest mean APB energy (0.227 J m^{-2}) at low concentration (1.39 at.%), although the variance in γ_{111} means that several other elements, including Ce, Ta, and W, are equally potent within the statistical scatter in our calculations. The contributions from Ta, W, and Re in maximizing γ_{111} at low solute concentrations have also been discussed in other DFT-based studies in the literature [111, 114].

To better illustrate the APB energy data in Table A.2, we plot in Figure 3.4 a few representative trends for the dependence of γ_{111} on solute concentration. In a majority of ternary systems, γ_{111} varies approximately linearly with concentration up to 10 at.%, which matches previous studies on γ_{111} in Ni_3Al-based alloys [112, 118–120]. Elements in groups IV and V of the periodic table, like Ti, help stabilize the γ' phase [4] and greatly increase

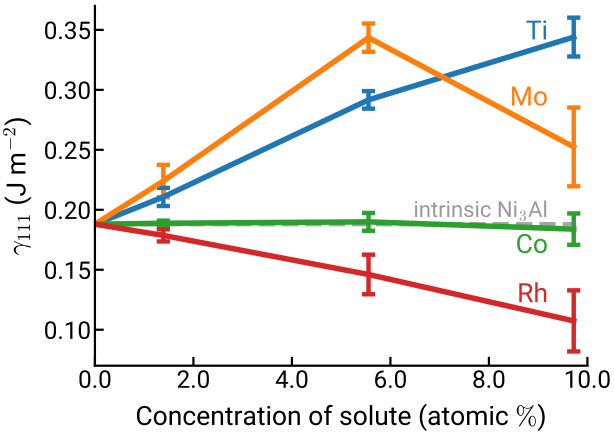


Figure 3.4: Representative profiles of the composition dependence of the APB energy. We plot γ_{111} against solute concentration for select species in model ternary Ni_3Al-based alloys, where error bars represent one standard deviation from the mean value. The intrinsic γ_{111} for Ni_3Al ($0.188\,J\,m^{-2}$) is given by the gray dashed line. For a majority of solutes, γ_{111} exhibits an approximately linear dependence on solute concentration, whether that trend is increasing (like Ti), decreasing (like Rh), or relatively constant (like Co). Solutes in groups VI (like Mo) and VII behave non-monotonically where γ_{111} increases up to the medium concentration and then decreases as the solute concentration further increases.

γ_{111} as a linear function of their concentration. A few of the elements behave similarly to Co, which has little effect on the APB energy [118], while others like Rh cause a roughly linear reduction in the APB energy with increasing solute concentration. We observe these non-increasing dependencies (exemplified by Co and Rh) most often in alloys where the ternary species is type I or type III, corresponding to Ni and mixed sublattice preference, respectively.

Perhaps the most striking dependence on concentration is displayed by the elements in groups VI and VII, with a representative example given by Mo in Figure 3.4. Adding these solutes increases the value of γ_{111} up to medium concentrations (5.56 at.%), but any further addition induces a drop in the APB energy. Gorbatov et al. reported a similar non-monotonic dependence in their CPA calculations for γ_{111} for Cr additions [118], while CALPHAD calculations by Crudden et al. revealed a similar trend in γ_{111} for Mo and W [113]. When we look at the variation in γ_{111} with respect to composition across the first three periods of the *d*-block (Figure 3.5), we see that this non-monotonic behavior is exhibited by all of the group VI and VII transition metals. Follow-up DFT calculations for γ_{111} without allowing for structural relaxations result in the same qualitative trend for Mo and W, leading us to hypothesize that the origin of such behavior is predominantly electronic. Vamsi and Karthikeyan found that W and Mo increase the contributions to the APB energy from Al–Al violations across the APB [149], which matches the dramatic strengthening effect

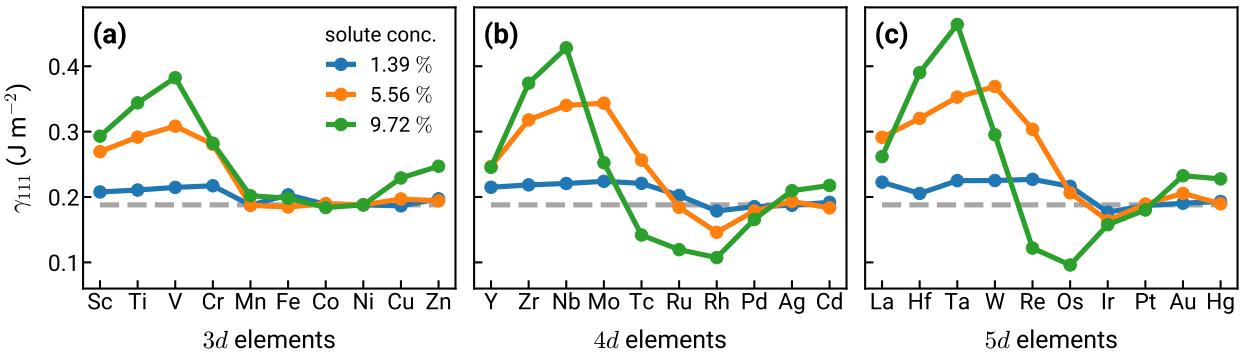


Figure 3.5: γ_{111} of model ternary Ni_3Al-based alloys for solutes from the first three periods of the d-block. γ_{111} at low, medium, and high solute concentrations are given by the blue, orange, and green profiles, respectively. γ_{111} for Ni at all concentrations is set to the γ_{111} value for Ni_3Al (0.188 J m^{-2}), which is also represented by the gray dashed lines.

we observe at the lower solute concentrations.

3.3.3 Machine learning modeling

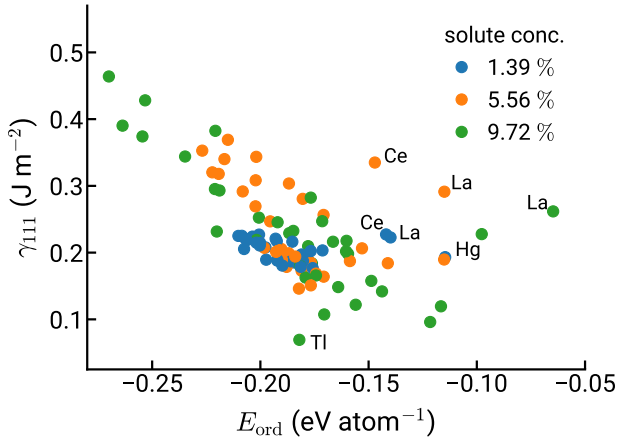


Figure 3.6: Visualizing the correlation between APB energy (γ_{111}) and ordering energy (E_{ord}) in Ni_3Al-based alloys. The two properties exhibit a negative, moderately strong linear correlation ($r = -0.541$). Points are color-coded based on solute concentration and select outliers are labeled with the ternary element.

To better understand the composition dependence of γ_{111}, we engineer several physically-motivated features and use ML methods to analyze their predictive capabilities in modeling the APB energy. We first use DFT to calculate the ordering energy (E_{ord}) according to

Equation 3.4 (see Methods) and obtain a value of $-0.118\,\mathrm{eV\,atom^{-1}}$ for Ni_3Al, which is consistent with the value of $-0.12\,\mathrm{eV\,atom^{-1}}$ obtained by Gorbatov et al. using DFT [118]. The trend for the ordering energy ratio $\left(E_{\mathrm{ord}}/E_{\mathrm{ord}}^{\mathrm{Ni_3Al}}\right)$ among the $3d$ elements also agrees qualitatively with the CPA results obtained by Gorbatov et al., although our energy ratios are higher by approximately 40 %. We plot γ_{111} against E_{ord} in Figure 3.6 and further differentiate the data points based on the ternary solute concentration. There exists a clear negative correlation between the two variables, with a Pearson correlation coefficient of $r = -0.541$, indicating a moderately strong linear correlation. This correlation appears to be present at all solute concentrations and agrees qualitatively with previous studies on the $3d$ transition metals [118].

Feature	r value
Mean Pettifor number	−0.543
Ordering energy	−0.541
Mean # of unfilled p orbitals	−0.527
Std. dev. of the # of unfilled d orbitals	0.692
Std. dev. of the column number	0.570
Mean covalent radius	0.547

Table 3.2: Linear correlations between the features and γ_{111}. The features with the three most negative and three most positive r values are shown, with four features having a stronger linear correlation to γ_{111} than E_{ord}. Except for the ordering energy, the feature name explains the statistic that is computed for the elemental values of that property.

In addition to the DFT-calculated ordering energy, density, and change in lattice parameter, we use several composition-based features generated using the matminer Python package [130] (see Methods). When we compute the Pearson correlation coefficient between each feature and γ_{111}, we find that four of the features have a stronger linear correlation to the APB energy than the ordering energy (Table 3.2). The feature with the strongest linear correlation ($r = 0.692$) is the weighted standard deviation of the number of unfilled d orbitals in each constituent element. The second strongest correlation ($r = 0.570$) is for the weighted standard deviation of the column number of each constituent element while the third ($r = 0.547$) is the weighted mean of the covalent radius of each constituent element. We find that the weighted mean Pettifor number [133] also has a marginally stronger linear correlation ($r = -0.543$) than the ordering energy. These features are all automatically generated from the alloy's composition using matminer [130] and their correlations to γ_{111} reflect the trends shown in Figure 3.5. For example, adding transition metals from the left side of the d block is shown to dramatically boost the APB energy and it increases the value of the first two features (those elements have more unfilled d orbitals than Ni (2) and Al

(0) and they have a smaller column number than Ni (10) and Al (13)). We also generally observe γ_{111} to decrease across a row and increase down a group, which agrees qualitatively with the trends in elemental covalent radii.

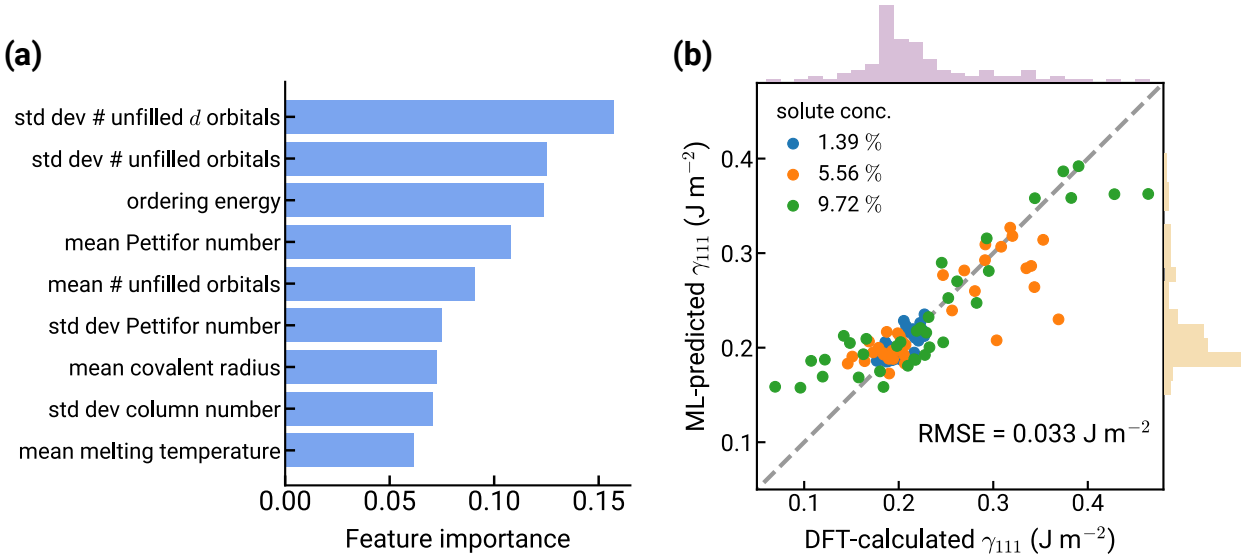


Figure 3.7: Results from training the random forest (RF) model using grouped 5-fold CV. (a) Ranked feature importances (normalized to sum to 1.0) for the nine features used in the final RF model. (b) ML-predicted γ_{111} vs. DFT-calculated γ_{111} for model ternary Ni_3Al-based alloys. Perfect agreement is given by the gray dashed line and points are color-coded based on solute concentration. The RF model achieves a RMSE of $0.033\,\mathrm{J\,m^{-2}}$ in 5-fold CV.

After filtering out features with undefined values and features that are highly correlated to another feature, we apply recursive feature elimination (RFE) to further downselect the features and use grouped cross-validation (CV) [137] to obtain the optimal number of nine features. We note that E_{ord} is the only physical parameter of the $L1_2$ structure that remains. We use these nine features to build a final random forest (RF) model and perform grouped 5-fold CV to predict γ_{111}. Figure 3.7a shows the normalized feature importances [136] computed by the RF algorithm for the nine features selected from RFE. The standard deviation of the elemental number of unfilled d orbitals surfaces as the most important feature, which follows from its large positive correlation to γ_{111}. The ordering energy is only the third most important feature in our model, preceded by the standard deviation of the elemental number of unfilled orbitals. The parity plot in Figure 3.7b compares the cross-validated predictions of γ_{111} using the RF model against the DFT-calculated values of γ_{111}. We again color the data points by solute concentration and observe relatively good agreement between the ML predictions and the DFT calculations. The overall RMSE from grouped 5-fold CV is $0.033\,\mathrm{J\,m^{-2}}$ ($R^2 = 0.753$), which is comparable in magnitude with the average experimental error of $0.027\,\mathrm{J\,m^{-2}}$ for multinary (> 2 elements) compositions in Table 1 from Crudden et

al. [113] As there may be interest in an ML model without E_{ord} as a feature, particularly for multicomponent compositions where this information may not be readily available, we report the 5-fold CV results for such a model in Appendix A.

3.3.4 Extrapolation to multicomponent chemistries

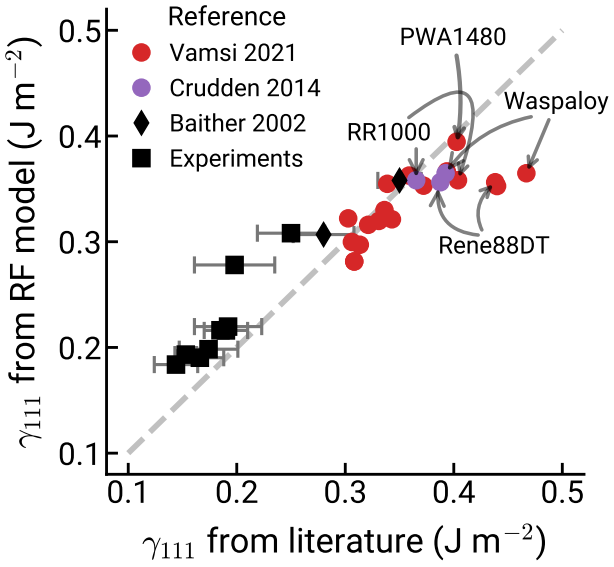


Figure 3.8: Testing the ML model on superalloy chemistries given in the literature. The ML model trained on only the ternary alloys with composition-based features achieves a test error of 0.038 J m^{-2} when used to predict γ_{111} in these systems. The computational data from Crudden et al. [113] and Vamsi and Karthikeyan [149] for commercial superalloys are shown in purple and red, respectively. Experimental measurements [107, 127, 150–153] with error bars are shown in black, with data for commercial superalloys from Baither et al. [108] shown as black diamonds. Select superalloys are labeled for clarity.

To demonstrate the generalizability of our ML model, we use our methodology to train an RF model to predict γ_{111} for real superalloy chemistries reported in the literature. Specifically, we use the compositions for the γ' phase reported by Baither et al. [108] and Vamsi and Karthikeyan [149], which were obtained through experiments and thermodynamic modeling, respectively. The γ' phase in these commercial superalloys contain multicomponent chemistries ranging from 5 to 11 constituent elements. As we do not have DFT-calculated data nor access to a thermodynamic database for these alloys, we first retrain the RF model with only composition-based features (does not include E_{ord}) for the full training set of 111 ternary Ni_3Al-based alloys (training error of 0.011 J m^{-2}; $R^2 = 0.974$). We then use this trained model to predict γ_{111} for the unseen test data (i.e., 23 commercial, multicomponent

superalloys) and compare the ML predictions with the γ_{111} values from the literature (Figure 3.8). While direct application of the model for ternary systems (which includes E_{ord}) is not possible, our analysis shows that models with comparable performance can be developed without E_{ord} (see Supplementary Methods). We also use several other experimental measurements from the literature [107, 127, 150–153] to test our model and we include those results in Figure 3.8 (black squares). The overall test RMSE is $0.038\,\text{J}\,\text{m}^{-2}$, which compares favorably to the CV error, even though the test set contains compositions that are more complex than what exist in our training set. Conversely, when we build a model that simply uses a linear combination of ternary APB energies [113, 120], i.e., $\Delta\gamma^{\text{total}} = \sum_{i=1}^{n} k_i \Delta\gamma_i^{\text{ternary}}$, we obtain a higher extrapolation error of $0.048\,\text{J}\,\text{m}^{-2}$ on the test set.

Compared with the experimental measurements by Baither et al. (black diamonds in Figure 3.8), our model predicts values for γ_{111} that are consistent with those reported in that study [108]. We find the error from our RF model reasonable given the comparison to experimental data and our model still outperforms other predictive models in the literature [120, 149] (Supplementary Figure A.1). The red points in Figure 3.8 represent the APB energies from the study by Vamsi and Karthikeyan, who used an environment-dependent nearest-neighbor bond (EDNNB) model that was also fit to DFT data [149]. They used a thermodynamic database to predict the composition of the γ'-phase for commercial superalloys and then used those compositions as inputs to their EDNNB model. Our ML model predicts values for γ_{111} that are consistent with their EDNNB model predictions for a majority of the superalloys, but under-predicts γ_{111} at higher values. We note, however, that our ML predictions for Waspaloy, Rene88DT, and RR1000 are in closer agreement with the predicted values from Crudden et al. (purple points) [113], which exemplifies the variability between different models for the APB energy.

3.4 Discussion

The APB energy is an important parameter for alloy design as it influences several mechanical properties of precipitation-strengthened superalloys and can be tuned by adjusting the chemical composition. By creating a computational workflow to automate DFT calculations, we generate a wealth of data including solute site preference, (111) APB energies, and physically-meaningful features such as the ordering energy to enable a data-driven assessment of the chemical contributions to the APB energy. The synergy between high-throughput calculations and machine learning presents an opportunity to explore vast swaths of alloy composition space and accelerate the process of alloy development.

It is interesting to note PyDII's predictions for Co and Cr, which are the two species with the greatest variability in site preference among previously published studies. For Co, we classify its behavior as preferring the Ni sublattice, but its concentration profile varies the most among type I elements, suggesting it has the possibility of occupying both sublattices [140, 144, 147] when analyzed with different compositions or at different temperatures. For Cr, PyDII predicts a very strong preference for the Al sublattice, which reasonably

agrees with previous computational studies, but differs from the MC study by Saito and Harada [154] and a few atom probe tomography experiments [155, 156]. These three studies notably used multicomponent superalloy chemistries and Booth-Morrison et al. found that Ta has a propensity to kick Cr out of Al sites [156], which could help explain the differences with our results for Cr. Competition between multiple solute species, which more closely mimics real superalloy chemistries, deserves further scrutiny in future investigations and can be incorporated in the dilute-solution thermodynamic formalism employed by PyDII. In general, validation against the literature confirms the viability of using PyDII as an automated tool to predict site preferences of solute species in ordered intermetallics, which can potentially be extended to other precipitation-strengthened alloys [157, 158].

Our DFT results also reveal that several elements exhibit a non-monotonic concentration dependence for γ_{111}, which can have important consequences for alloy design. As we scan across the *d*-block elements (Figure 3.5), Ta may be the element that maximizes γ_{111} at high concentrations, but the same cannot be said at lower concentrations, where other solutes like Mo and W perform comparably or better. Furthermore, several predictive models for the APB energy in the literature use a linear regression model based on each solute's concentration [113, 120]. Crudden et al. [113] calculated γ_{111} for 12.5 at.% Mo and W, and they assumed that the APB energy varied linearly between 0 at.% and 12.5 at.%. Our results show that for several elements, particularly those in groups VI and VII which are currently used in superalloys, the choice of those endpoints can affect the estimate of its chemical contribution to the APB energy in the γ' phase, and we must be more cautious of making assumptions about monotonicity. These data can also be used to guide the selection of alloying elements to maximize APB energy when facing additional constraints (e.g., total solute quantity, weight, cost, processing techniques).

Building on previous analytical models for the APB energy, our nonlinear ML model uses a derived set of physically-motivated features and is not constrained to a particular analytical form. Seeing ordering energy surface as one of the most important features for predicting γ_{111} agrees favorably with our physical intuition, given the known relationships between the APB energy and the ordering energy [113, 118, 123]. Furthermore, the ability to extract the ordering energy of an alloy system from CALPHAD [123] makes it a viable input feature for modeling the APB energy by integrating ML workflows with existing tools and databases from alloy design [159,160]. Using the stoichiometry alone, we can calculate several additional features that improve the generalizability and interpretability of our ML model. Evidenced by the grouped CV performance, these features are already effective at making predictions for alloy chemistries that are not present in the training dataset, albeit with room for improvement (W at medium concentration is the severely under-predicted orange data point in Figure 3.7b). This predictive ability is important for exploratory alloy development, where such a model is sufficiently accurate to be able to screen for promising compositions (i.e., rank those with high APB energies as shown in Figure 3.8) in new regions of chemical space and can couple with existing tools for increased predictability in the broader context of alloy design. The continued expansion of databases for multicomponent alloys presents exciting opportunities to enrich our ML models and extrapolate the methods presented here

to other extended defects (e.g., stacking faults). As materials data become increasingly digitized, abundant, and accessible, we foresee the coupling of materials thermodynamic databases and data-driven methodologies to have an outsize impact in this field.

Chapter 4

Grain boundary optimization in α-Ti using GRIP

The work presented in this chapter is based on a soon to be submitted manuscript titled "Grand canonically optimized grain boundary phases in hexagonal close-packed titanium" by Enze Chen, Tae Wook Heo, Brandon C. Wood, Mark Asta, and Timofey Frolov, and is reproduced here with permission of the co-authors.

4.1 Introduction to GB structure search

Grain boundaries (GBs) are interfacial defects in crystalline materials that have long been studied for their influence on materials properties and performance [3]. Given their ability to exist in multiple stable and metastable states, which have been termed GB phases [161] or complexions [162], it is desirable to obtain an atomic-level understanding of the GB structure and possible GB phase transitions between them [39, 163, 164]. The structure–property relationships of these interfacial phases are believed to influence phenomena such as diffusion [165], GB migration [166], and GB sliding [167] in materials.

As recent experiments have provided direct [168] and indirect [169] evidence for GB phase stability, coexistence, and transitions in metallic systems, atomistic simulations of interfaces [20] can provide a complementary method to search for GB phases [170]. Previous atomistic modeling studies have discovered a diverse array of GB phases present in face-centered cubic (FCC) [39, 171], body-centered cubic (BCC) [54], diamond cubic (DC) [75, 172,173], and other cubic systems [60,174]. One notable feature shared by the aforementioned works is the ability to add or remove atoms from the GB region in the simulation cell, i.e., grand canonical optimization (GCO), which was required to access new ground states and metastable states. While the exchange of atoms at an interface could naturally occur due to diffusion and radiation damage in real materials at finite temperature, this variation is omitted by the γ-surface computational method [175], which is the traditional technique for simulating GBs where only translations are allowed between two bulk slabs. The γ-surface

method is often adopted for its simplicity, but the deficiencies exposed by the previous studies suggest that more degrees of freedom (DOF) must be considered during optimization in order to find the true ground state structure in certain GBs. A few alternative approaches from the literature for atomistic modeling of GB structures include high-temperature molecular dynamics (MD) simulations [39, 55], Monte Carlo (MC) sampling [56–59], and evolutionary algorithms (EA) [59–64] that can access a greater diversity of structures and atomic densities at the interface, with a concomitant trade-off in computational complexity. These algorithms have employed GCO for GB structures in a variety of systems, although seldom in a high-throughput manner [56,176], and it remains unclear if the ubiquity of GB phases extends to non-cubic systems.

A particular system of immense technological relevance is the hexagonal close-packed (HCP) crystal structure, exemplified by elemental metals such as Mg, Zr, and Ti, the last of which (α-Ti) will be the focus of this work. Ti alloys are technologically important structural alloys for aerospace, biomedical, and energy applications, particularly where high specific strength and strong corrosion resistance are desired [177]. The importance of GBs in the α-Ti system is illuminated by recent studies that used grain refinement to mitigate low-temperature oxygen embrittlement in α-Ti [7] and 3D electron backscatter diffraction (EBSD) to map out the complete GB character distribution in this system [178]. In a previous study [67], we used the evolutionary algorithm USPEX [65,66] to discover a ground state structure for the $\{11\bar{2}4\}[1\bar{1}00]$ twin boundary (TB) in α-Ti that was in closer agreement to density functional theory (DFT) calculations and high-resolution transmission electron microscopy (HRTEM) results than previously reported structures. In addition to these experimental works, there are also several atomistic simulation studies in the literature that systematically model symmetric tilt grain boundaries (STGBs) in α-Ti along the [0001] [179–181], $[1\bar{1}00]$ [182–185], and $[1\bar{2}10]$ [185,186] tilt axes. Despite the simplicity of STGBs—only a tilt axis and tilt angle (2θ) are required to describe the crystallographic misorientation between two bulk crystals—they include coherent TBs as an important subclass, several of which are experimentally observed in deformation microstructures and thus important for mechanical properties in α-Ti [7,177]. STGBs are also model systems to study the geometric relationships of defects at the interface [186]; however, as the previous studies utilized the γ-surface method, it would be important to clarify the effects of GCO on STGB structure in α-Ti and more broadly whether interfacial phases exist in HCP metals.

Herein, we perform GCO of low-index STGBs in α-Ti using an open-source GRand canonical Interface Predictor (GRIP) tool that we developed to rigorously sample microscopic DOF at the GB. We use this tool along with empirical potentials to perform GB structure search, discovering new ground state structures and GB phases. We further employ high-temperature MD simulations to explore the $\{21\bar{3}0\}[0001]$ STGB and demonstrate GB phase (meta)stability and phase transitions through a novel dislocation-pairing mechanism. We conclude by discussing the broader implications of these results on GB phase behavior in HCP systems and how our tool can benefit the GB structure optimization community.

4.2 Methods

4.2.1 GB structure search

We perform atomistic simulations of STGBs using the open-source GRIP code, which rigorously explores structural DOF and was discussed in Chapter 2. The input slabs were generated for α-Ti using a combination of ASE and Pymatgen [187]. For computational tractability, we choose to simulate only [0001], [$1\bar{1}00$], and [$1\bar{2}10$] STGBs where all Miller-Bravais indices (for the plane and the in-plane x-direction) are less than or equal to 15, resulting in 16, 40, and 94 STGBs for each of the tilt axes, respectively (150 total). Additional simulation parameters include a GB region of 1 nm on each side (refer to Figure 2.2), a buffer region of 0.6 nm beyond each side of the GB region, a temperature between 300 K and 1200 K, and a duration up to 3×10^5 MD time steps.

We compare the results of structure optimization using two different interatomic potentials (IAPs), an embedded-atom method (EAM) potential for Ti–Al from Zope and Mishin [188] and a modified embedded-atom method (MEAM) potential for Ti from Hennig, et al. [189] For each STGB and IAP, we also optimize the structure using the γ-surface method [175] for comparison, using a 2×4 replication of the same bicrystals and translating the top slab in increments of 0.025 nm in the x- and y-directions prior to a conjugate gradient energy minimization.

4.2.2 High-temperature MD simulations

To study GB phase stability and transitions, we perform high-temperature MD simulations using methods adapted from previous work [39]. Briefly, we replicate the STGB in the x- and y-directions until the simulation cell is around 10 nm in the x-direction and 3 nm in the y-direction (along the tilt axis). We freeze the bottom 1 nm layer of atoms and constrain the top 1 nm layer to be semi-rigid (moves as one block using the `fix aveforce` command in LAMMPS) throughout the 20 ns simulation. We use periodic boundary conditions (PBCs) in the y-direction and both PBCs and open surfaces (with 1 nm of vacuum) in the x-direction. We scan a range of temperatures between 600 K and 1200 K.

To induce a phase transition, we either insert additional Ti atoms at interstitial sites in the GB region or delete Ti atoms (i.e., create vacancies) from a region near the top of the GB region. As seen in Appendix Figure B.4, the vacancies are created approximately 1.3 nm above the GB plane. These MD simulations are performed in the canonical (NVT) ensemble between 600 K and 1200 K for up to 20 ns, using the MEAM potential and associated structures. For clarity of visualization, we relax all structures at 0 K using a conjugate gradient minimization scheme.

4.3 Results

4.3.1 Grand canonical optimization

GB structure prediction is a long-standing challenge that requires rigorous and often advanced sampling of possible interfacial structures. Previous studies of GBs in HCP metals generated the interfaces using the common γ-surface method [175] in which two perfect half-crystals are joined together while sampling different relative grain translations, followed by a conjugate gradient energy minimization. As was illustrated in several previous studies of GBs in cubic systems, this method is not guaranteed to yield the true ground state configurations in a general case [62, 75]. The conjugate gradient minimization simply allows the atoms to fall into the nearby local minima which may be far away from the ground state. For example, complex GB core configurations may exist that require significant rearrangement of the constituent atoms [62, 75].

The other significant limitation of the γ-surface method is that it is not grand canonical: All GBs created using this method are composed of the same number of atoms that came from the constituent perfect half-crystals. This poses a significant constraint because many other structures, including true ground states, can be realized out of a different number of atoms at the interface [39, 54, 75]. For STGBs and a fixed reconstruction area, the number of distinct atomic densities that can give rise to different GB structures is given by the total number of atoms in one atomic plane parallel to the boundary, which we denote $N_{\text{plane}}^{\text{bulk}}$. This quantity is the limit because removing a full plane of atoms from a crystal will return the exact same configuration up to a relative grain translation.

In this work, we expand the notion of $N_{\text{plane}}^{\text{bulk}}$ to multi-basis crystals like HCP (Figure 2.5). HCP has two atoms per basis which we visualized using different colors (blue and gold) to obtain two sublattices. First, we calculate the distance d_{hkl} which corresponds to the smallest normal component of a lattice vector connecting two atoms on the same sublattice with different z positions along the GB normal. Then, we calculate the total number of atoms of both sublattices found inside the region spanned by d_{hkl}. These regions calculated for two different crystal orientations are illustrated in Figure 2.5. Due to the presence of two basis atoms, some GB misorientations are like that in Figure 2.5a, where the atoms inside the region have two distinct z positions, and we refer to these GBs as "Type 1" boundaries. In contrast, some orientations are like the one shown in Figure 2.5b where both basis atoms have the same z position, which we refer to as a "Type 2" boundary. We note that our definition of a plane of atoms works for both cases, allowing us to uniformly apply it in calculating GB atomic density (n; see Equation 2.7 in Chapter 2).

For each bicrystal studied, we use the open-source tool GRIP to systematically explore possible GB structures by sampling the entire range of possible translations and atomic densities. The different densities of GB atoms are generated by randomly introducing vacancies in the GB plane. For a fixed translation and number of GB atoms, the structure is then optimized by performing MD simulations at different temperatures sampled within a wide window between ambient temperature (300 K) and 1200 K (approximately $T_{\alpha\to\beta}$ for Ti), and

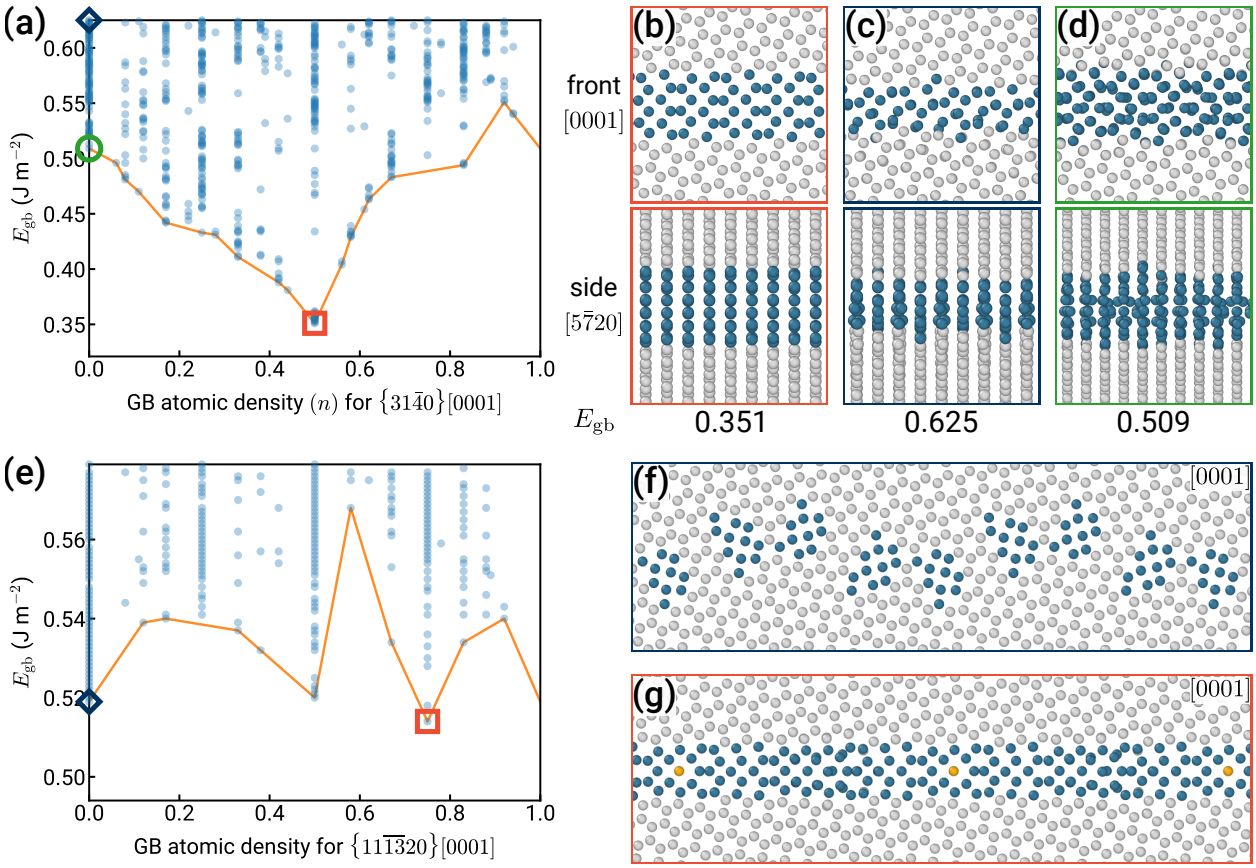


Figure 4.1: Representative results from grand canonical optimization (GCO). Shown here are the (a)-(d) $\{31\bar{4}0\}[0001]$ (Type 1) and (e)-(g) $\{55\bar{1}012\}[0001]$ (Type 2) STGB optimized using the MEAM potential [189]. (a) GB energy (E_{gb}) vs. GB atomic density (n), where each translucent dot is a distinct structure and the orange boundary serves as a guide to the eye. On the right, two orthogonal projections are shown for each minimum-energy structure obtained using (b) our algorithm at $n = 0.5$, indicating that GCO is necessary to find the ground-state structure for $\{31\bar{4}0\}$; (c) the γ-surface method; (d) our algorithm at $n = 0$. (e) E_{gb} vs. n for $\{55\bar{1}012\}$, where the minimum is at $n = 0.75$. On the right, frontal projections are shown for the (f) γ-surface structure and (g) GCO algorithm structure. The atoms are colored according to the common neighbor analysis in OVITO [76,190], where gray are HCP-coordinated atoms, gold are FCC-coordinated atoms (in panel (g) only), and blue are atoms in the GB with a different coordination.

for different durations sampled up to 300 ps. Due to the random vacancy initialization and different sampled MD parameters, for a fixed microscopic DOF we perform hundreds of such calculations in parallel to ensure rigorous structure exploration. At the end of each MD simulation, we perform conjugate gradient minimization at 0 K until convergence before calculating the GB energy (E_{gb}; see Equation 2.6 in Chapter 2).

Figure 4.1 illustrates the results of the grand canonical structure optimization performed for two representative GBs. Panels (a) and (e) show plots of E_{gb} vs. n, where each point on the plot represents a particular GB structure obtained after the energy minimization. Our thorough exploration generates thousands of different distinct structures covering different densities and energies. As seen in Figure 4.1a, $\{31\bar{4}0\}[0001]$ is an STGB in α-Ti that requires GCO to achieve the ground state (orange square). Specifically, we find that half of the atoms in one $\{31\bar{4}0\}$ plane must be removed ($n = 0.5$) to obtain the ground state structure (Figure 4.1b), which has an energy of $0.351\,\text{J}\,\text{m}^{-2}$ when evaluated with the modified embedded-atom method (MEAM) potential [189]. We observe a uniform pattern of hexagonal motifs in the GB with no atoms in between the (0002) planes and no localized cores for grain boundary dislocations when analyzed with the dislocation extraction algorithm (DXA) in OVITO [190, 191]. In contrast, the minimum-energy structure from the γ-surface method (blue diamond) is shown in Figure 4.1c, where the GB atoms (colored blue according to the common neighbor analysis in OVITO [76]) have a more disordered pattern and are slightly displaced out of the (0002) planes, as seen in the side projection. This GB structure has an excess energy of $E_{\text{gb}} = 0.625\,\text{J}\,\text{m}^{-2}$, almost 80 % higher than that of the ground state.

As demonstrated in previous studies [54, 62], the γ-surface method may not produce the lowest-energy structure even without varying the atomic density, and the GRIP tool finds a GB structure at $n = 0$ with $E_{\text{gb}} = 0.509\,\text{J}\,\text{m}^{-2}$ (green circle), approximately 18 % lower in energy. As shown in Figure 4.1d, this structure bears a closer resemblance to the ground state structure than the γ-surface structure, with the hexagonal units starting to form when viewed down the tilt axis; however, a prominent difference is the displacement of atoms in between the (0002) planes that is necessary to accommodate the structural motifs when no atoms are removed from the GB. While this structure is clearly metastable and would favor a different planar fraction of atoms in the GB, it underscores the importance of including MD simulations during GB structure optimization in order to sample more disordered configurations. A similar set of results is shown for $\{55\bar{1}012\}[0001]$ in Figure 4.1e–g, where the two selected structures look strikingly different. The minimum-energy structure at $n = 0.75$ (Figure 4.1g) discovered using GRIP lacks the localized dislocation cores present in the γ-surface structure (Figure 4.1f), which also has marginally higher energy. Notably, whereas $\{31\bar{4}0\}$ is a Type 1 GB with different basis atom terminations along the GB normal (see Figure 2.5), $\{55\bar{1}012\}$ is a Type 2 GB, highlighting the need for GCO across diverse misorientations and planar terminations.

Figure 4.2 summarizes the results from GCO for the entire family of [0001] STGBs. Each subplot is equivalent to the orange boundary in Figure 4.1a, where the green circles are the lowest-energy structures at different n and the orange square marks the ground state. The lowest-angle ($\leq 6.58°$) and highest-angle ($\geq 23.41°$) tilt boundaries in this family do not appear to require GCO to obtain the ground state, as there is close agreement between our algorithm and the γ-surface method (blue triangles) at $n = 0$. For intermediate angles, however, Figure 4.2 clearly shows the importance of GCO in finding the ground states, which are mostly located at $n = 0.5$. Similar to $\{31\bar{4}0\}$, the γ-surface method often performs poorly for these GBs, getting higher energies and different structures than the GRIP tool.

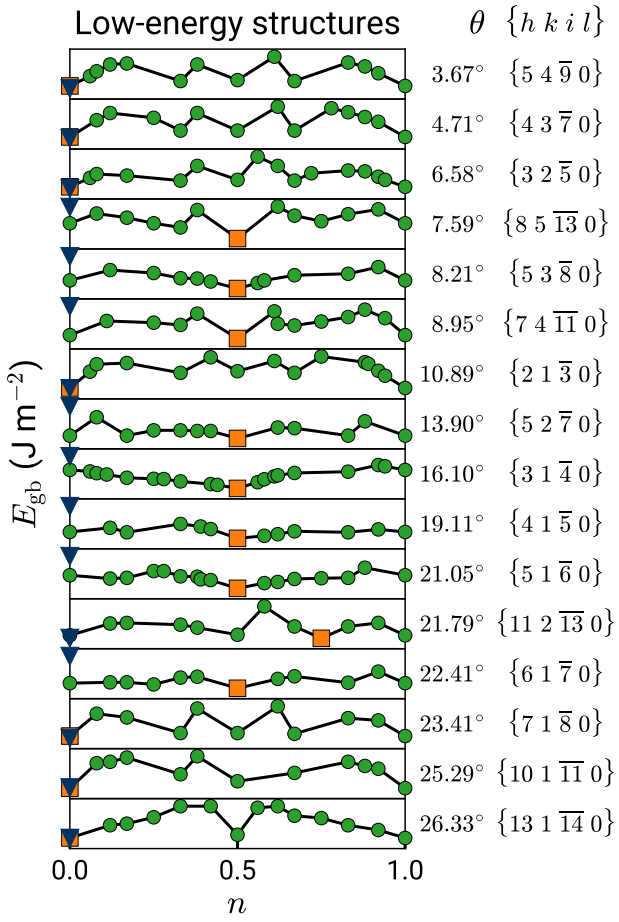


Figure 4.2: Energy map of [0001] **STGBs evaluated using the MEAM potential**. Each subplot is analogous to the orange boundary in Figure 4.1a, where the green circles are the lowest-energy structures at different GB atomic densities. The orange squares mark the ground state for each tilt angle obtained using GCO, while the blue triangles mark the γ-surface structure (necessarily at $n = 0$). The corresponding tilt angle (θ) and Miller-Bravais planar indices ($\{hkil\}$) are indicated in the right columns.

Even when the results are structurally similar, the sub-optimal atomic density from the γ-surface method leads to metastable configurations that have demonstrated implications for GB migration [166] and plastic deformation [192] (in other chemical systems). Additional energy maps for select $[1\bar{1}00]$ and $[1\bar{2}10]$ STGBs with low and high tilt angles are shown in Figure 4.3, and results for the embedded-atom method (EAM) potential [188] are presented in the Appendix (Figure B.2 and Figure B.3). Taken together, these results demonstrate the ubiquitous need for grand canonical sampling in locating the ground state structures for multiple tilt axes in HCP Ti.

While the need for GCO for so many [0001] STGBs is unexpected, finding the ground state

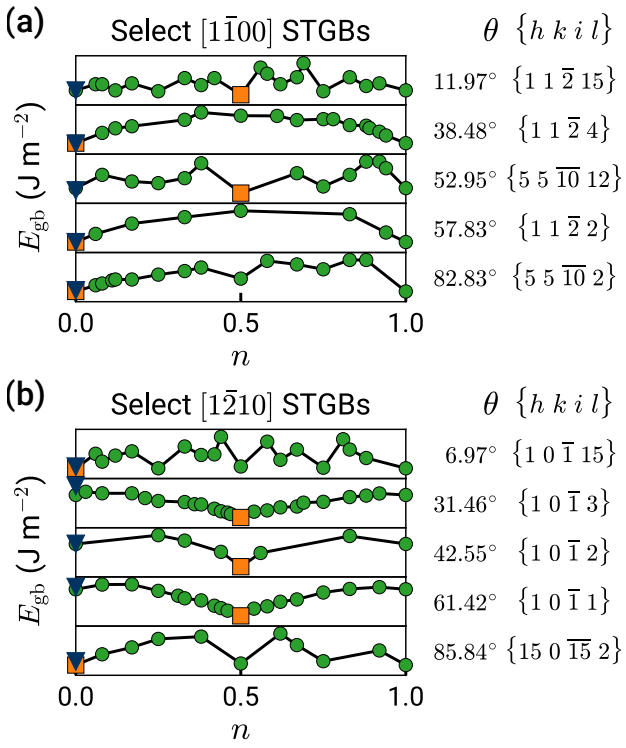


Figure 4.3: Additional energy maps of a few (a) [1$\bar{1}$00] and (b) [1$\bar{2}$10] STGBs. Minima at $n = 0.5$ indicate that GCO is broadly required to find the ground states in several families of STGBs in α-Ti.

at the particular value of $n = 0.5$ may not be. The two-atom basis of HCP crystals like α-Ti means that some misorientations result in the two basis atoms having different z-coordinates (Figure 2.5a), and those GBs will have different terminations for the bulk slabs depending on where they are cut in the z-direction. In turn, this cut will either leave a perfect planar periodicity in the z-direction ($n = 0$) or a half-terminated plane ($n = 0.5$), exposing different basis atoms and bonding environments. This nuance is well known within the plasticity community [193, 194], which historically proceeded to sample all terminations to find the "best" (e.g., lowest-energy) one. Indeed, we show in Appendix Figure B.1 how performing the traditional γ-surface method on two different terminations of $\{31\bar{4}0\}[0001]$ bulk crystals can reproduce the high-energy $n = 0$ structure and the low-energy $n = 0.5$ structure in Figure 4.1. Thus, another way to view the grand canonical sampling for certain GBs in HCP systems is incorporating additional flexibility in selecting the appropriate termination for the GB ($n \in \{0, 0.5\}$); however, our algorithm more precisely quantifies this selection and captures the full range of atomic densities $\in [0, 1)$, which is critical for Type 2 boundaries like $\{55\overline{1}012\}$ shown in Figure 4.1e or when larger reconstructions of the GB are required [62,75].

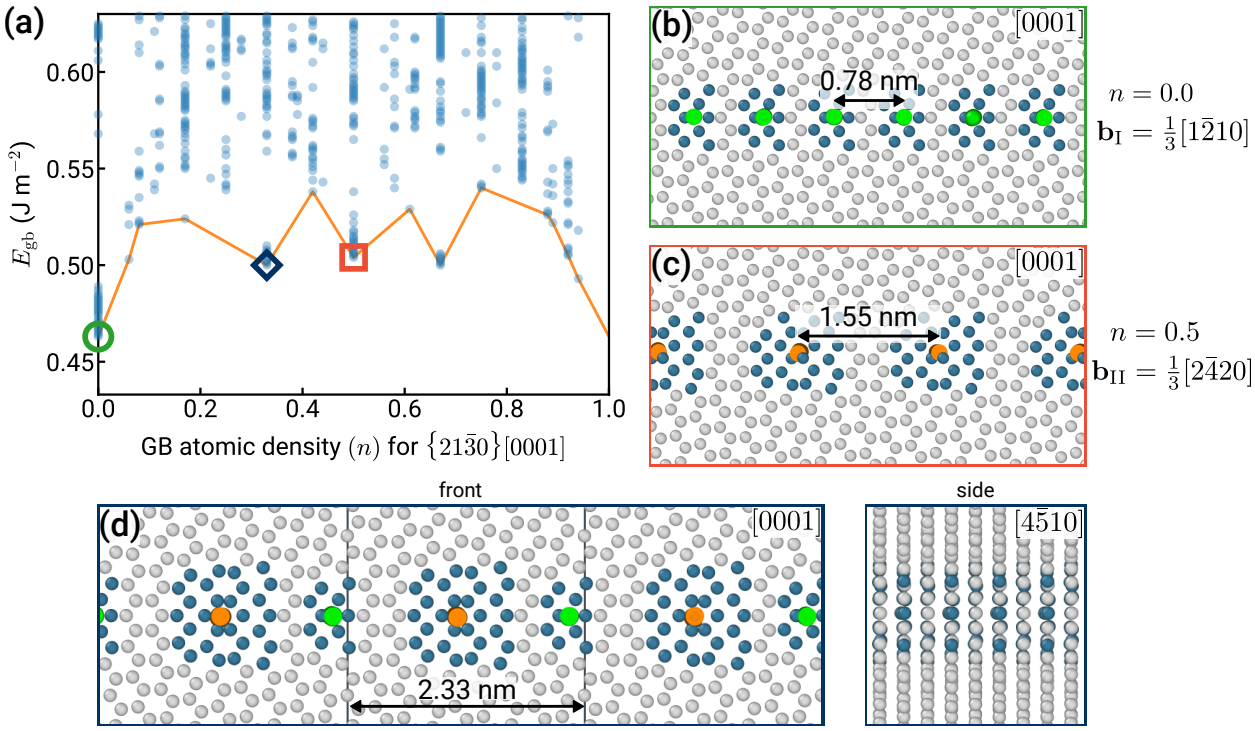


Figure 4.4: GB phase stability in $\{21\bar{3}0\}[0001]$. (a) The plot of E_{gb} vs. n reveals a global minimum at $n = 0$ (green circle) and a local minimum at $n = 0.5$ (orange square). (b) The ground state contains an array of $\mathbf{b}_I = \frac{1}{3}\langle 1\bar{2}10\rangle$ edge dislocations spaced approximately 0.78 nm apart (identified using DXA in OVITO [191]), while (c) the metastable structure requires half of one plane of atoms to be removed and the dislocations have reconfigured into $\mathbf{b}_{II} = \frac{1}{3}\langle 2\bar{4}20\rangle$ edge dislocations with twice the spacing. Both phases remain stable for up to 20 ns in MD simulations up to $T = 1150$ K. (d) The ground state for $n = 0.33$, which is a mix of $\mathbf{b}_I$ and $\mathbf{b}_{II}$ dislocations. The viewing directions are shown in the top-right corner.

4.3.2 Phase transitions and coexistence

By tracking how GRIP manipulates structural DOF, we can further discover phase transformation pathways. Here we focus on the $\{21\bar{3}0\}[0001]$ STGB, which has a relatively low misorientation angle of $\theta \approx 10.9°$. The structure search performed on this boundary is illustrated in Figure 4.4, where we identify two distinct GB phases corresponding to atomic densities $n = 0$ (green circle) and $n = 0.5$ (orange square). The $n = 0$ phase is the ground state (Figure 4.4b), while $n = 0.5$ has higher energy at 0 K (Figure 4.4c), with both corresponding to GB energy cusps with respect to n. Both phases are composed of periodic arrays of edge dislocations with distinct localized cores, which is characteristic of low-angle GBs [195]. The ground state is comprised of an array of $\mathbf{b}_I = \frac{1}{3}\langle 1\bar{2}10\rangle$ edge dislocations (identified using DXA [191]), while the second phase is comprised of $\mathbf{b}_{II} = \frac{1}{3}\langle 2\bar{4}20\rangle$ edge dis-

locations with Burgers vector twice that of the ground state and consequently half the line density within the GB plane. We perform MD simulations of each structure at temperatures as high as 1150 K for up to 20 ns to confirm that they are dynamically stable and indeed represent two GB phases. The other two energy cusps at $n = 0.33$ and $n = 0.67$ are the mixed states expected from the lever rule, where the GB region is patterned by weighted fractions of $\mathbf{b}_{\mathrm{I}}$ and $\mathbf{b}_{\mathrm{II}}$ dislocations corresponding to the proportions between $n = 0$ and $n = 0.5$ (Figure 4.4d).

These two GB phases and their dislocation content resemble the thermally activated dislocation-pairing transitions that occur in BCC Fe [196], and we demonstrate a distinct, point defect-induced mechanism in this system. We insert extra atoms into interstitial sites in the ground state structure to model irradiation phenomena, which triggers a dislocation-pairing transformation illustrated in Figure 4.5. During the transformation, the $\mathbf{b}_{\mathrm{I}} = \frac{1}{3}\langle 1\bar{2}10\rangle$ dislocations of the ground state structure pair up into $\mathbf{b}_{\mathrm{II}} = \frac{1}{3}\langle 2\bar{4}20\rangle$ and absorb the extra atoms. Analogously, adding Ti atoms to the left half of the metastable structure and performing high-temperature MD triggers a dislocation-unpairing transition as shown in Figure 4.5b ($\mathbf{b}_{\mathrm{II}} \rightarrow 2\mathbf{b}_{\mathrm{I}}$). The transformed states remain stable for up to 20 ns and the transformation can be reversed by introducing vacancies near the GB (Appendix Figure B.4). Thus, the transformed structures represent heterogeneous states containing two different GB phases that show stable coexistence in the closed system at high temperature.

To illustrate the shape of the nucleus during such a transformation, we increase the cross section of the GB and place interstitial atoms in a relatively small section. During the subsequent high-temperature simulations, the extra atoms diffuse to the boundary core and locally trigger the pairing transition. The equilibrium structure of the obtained nucleus is illustrated in Figure 4.5c. The dislocations of the parent structure shown in green ($\mathbf{b}_{\mathrm{I}}$) pair up on the nucleus boundary to form three individual orange segments ($\mathbf{b}_{\mathrm{II}}$). GB phase nucleation by absorption of point defects has been previously illustrated [39,74], which found the line defect separating the two different GB phases to depend on the Burgers vector and the core energy [197]. The important distinction of the transformation studied here is that it occurs in a low-angle GB; therefore, the structure of the GB phase junction is represented by a dislocation node, which is the point where two dislocations pair up into one. A collection of such dislocation nodes represents the structure of this GB phase junction, and we are unaware of previous reports of this nucleation mechanism for interfacial phases with distinct dislocation character.

4.4 Discussion

In this work, we perform grand canonical GB structure search to discover new GB phases in an HCP metal, α-Ti. While GBs in α-Ti have been investigated extensively [179–186], those simulations were restricted to a fixed number of atoms and clean surface terminations. By rigorously exploring all possible atomic densities at GBs, we show that the lowest-energy structures can be found for atomic densities inaccessible to the γ-surface method for both

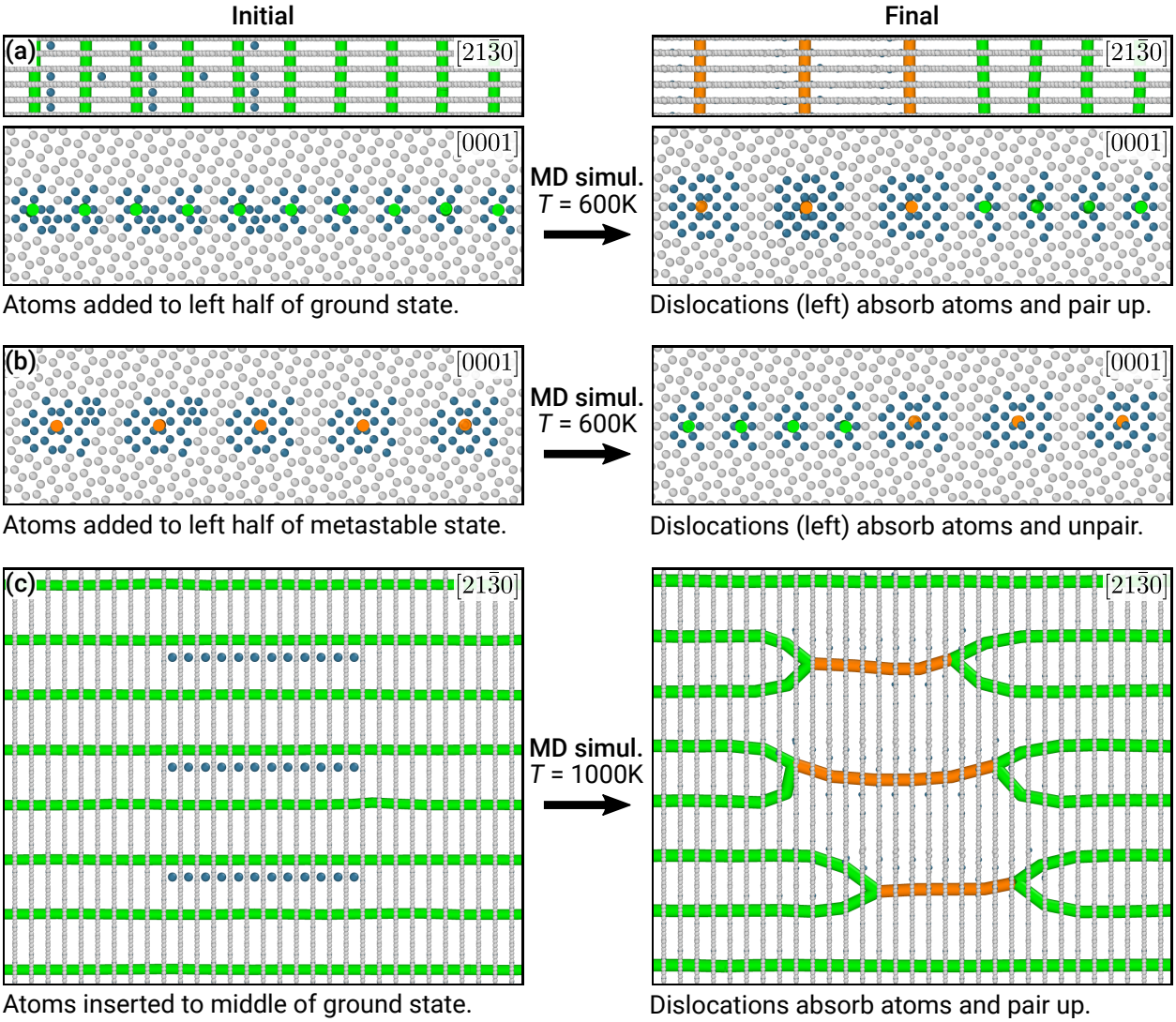


Figure 4.5: GB phase transformations in $\{21\bar{3}0\}[0001]$. (a) Adding Ti atoms to the left half of the ground state structure and performing MD simulations at $T = 600\,\mathrm{K}$ triggers a dislocation-pairing transformation ($2\mathbf{b}_\mathrm{I} \rightarrow \mathbf{b}_\mathrm{II}$) that remains stable for up to 20 ns. (b) Analogously, adding Ti atoms to the left half of the metastable structure and performing high-temperature MD triggers a dislocation-unpairing ($\mathbf{b}_\mathrm{II} \rightarrow 2\mathbf{b}_\mathrm{I}$) transition. (c) The two dislocation structures can coexist along the direction of the dislocation line when additional Ti atoms are inserted in the middle of one network, triggering a transformation locally. All structures are relaxed at 0 K for clarity of visualization and the viewing directions are shown in the top-right corner. Note the absence of atoms between (0002) planes in the final states.

high-angle and low-angle GBs. The need for GCO and presence of multiple GB phases with different atomic densities is consistent with phenomena previously illustrated in elemental

cubic metals with FCC [39, 171] and BCC [54] crystal structures.

Subsequent high-temperature MD simulations guided by our sampling of phase space confirm the discovery of a novel GB phase transformation mechanism. The two phases shown in Figure 4.4 are composed of periodic arrays of edge dislocations with distinct localized cores that contain different atomic densities in the GB. During a transition, dislocations of a less dense GB ($\mathbf{b}_{\mathrm{I}}$, $n = 0$) pair up to form a new dislocation core ($\mathbf{b}_{\mathrm{II}}$, $n = 0.5$) while doubling the Burgers vector and absorbing interstitial atoms. Previous studies of GB phase transitions in low-angle GBs revealed defect absorption by individual dislocation cores without the change of the Burgers vector [39, 74] and other studies demonstrated the change in the dislocation network topology for the same constituent number of atoms [196]. It is also well established and expected that individual dislocations absorb point defects by climb [198]; yet here, we demonstrate a different mechanism when there is a coupling between the dislocation network topology and the number of constituent atoms. This coupling could provide another mechanism for radiation tolerance for materials with a high density of low-angle GBs, such as additively manufactured metals that contain dense dislocation cellular walls [199]. Our work provides new insights about how low-angle GBs and dislocation arrays more generally interact with point defects, which may be prominent in nuclear applications for HCP metals like Zr [200, 201].

We extend the notion of GB number of atoms to non-cubic, multi-basis crystals like HCP. Previous studies on cubic metals such as FCC and BCC calculated this number ($N_{\mathrm{plane}}^{\mathrm{bulk}}$) as the total number of atoms located in one planar cut parallel to the GB, i.e., all these atoms are equidistant in the z-direction. This is not always the case for HCP materials or any multi-basis crystal (see Figure 2.5). Generally, $N_{\mathrm{plane}}^{\mathrm{bulk}}$ includes all atoms located inside a region with height equal to the minimum normal component of a lattice vector. Previous studies that used the γ-surface method for HCP demonstrated the importance of sampling different crystal terminations. In our study, we show that when these different terminations exist (Type 1 boundaries, Figure 2.5), they simply correspond to atomic fractions of $n = 0$ or $n = 0.5$, and therefore the search space is still too restricted. Also, the converse is not true, as $n = 0.5$ is not in general equivalent to a different crystal termination (Type 2 boundaries). In a proper GB search, all different atomic densities must be systematically explored. More broadly, all the special cases from previous studies are naturally handled within the single, consistent framework presented here.

We implement this framework for manipulating structural DOF and predicting new GB phases in the open-source GRIP tool, written in Python with minimal dependencies. The algorithm rapidly samples the configurational space described by relative translations and different atomic densities and moves the system toward equilibrium. The relevant DOF (e.g., atomic density, reconstructions, temperature) are specified by the user in a single input file and the code exhaustively explores the GB phase space by sampling as many structures as possible in parallel. To further underscore the need for rigorous sampling, Figure 4.6 shows the structural diversity with respect to the MD simulation parameters, namely temperature and duration (MD steps). We use GRIP to optimize a 6×2 reconstruction of the $\{21\bar{3}0\}[0001]$ GB with atomic density $n = 0.67$, as this boundary exhibits many competing phases (see

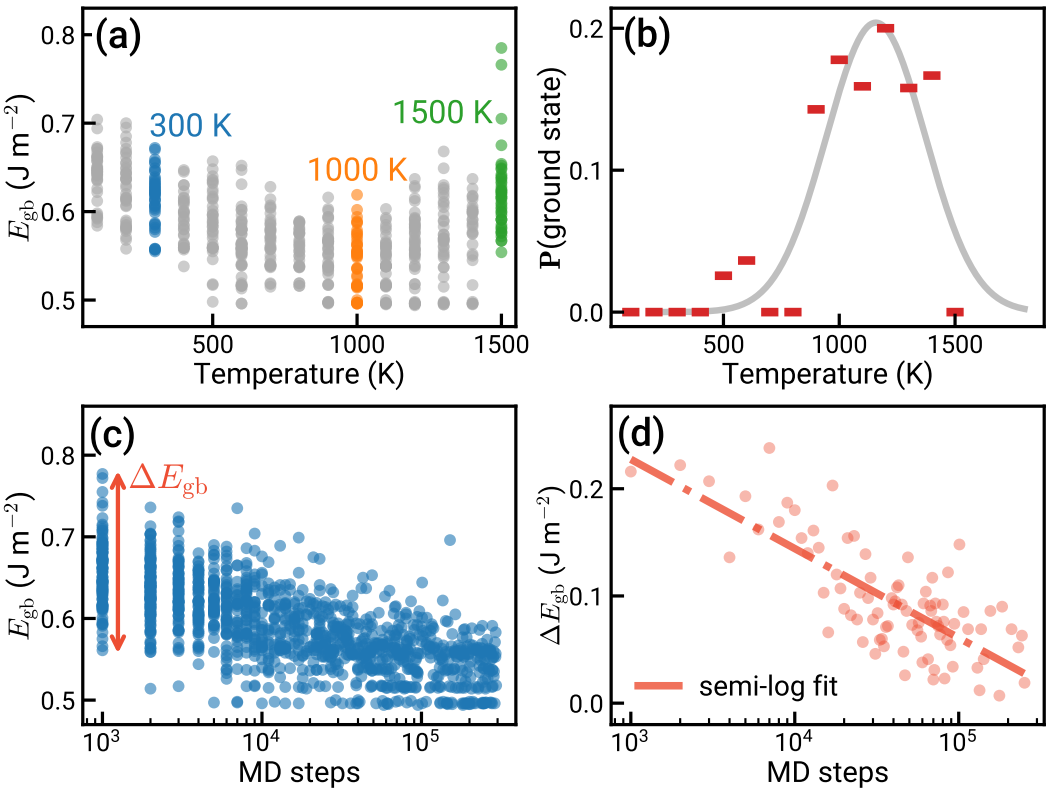


Figure 4.6: Variation of structures with respect to MD parameters. Here we optimize the structure for $\{21\bar{3}0\}[0001]$ with $n = 0.67$ and a 6×2 reconstruction. (a) If we fix the duration at 10^5 MD steps, there is significant variation in energy in the structures at each temperature, and a sufficiently high temperature (≥ 500 K) is required to obtain the ground state. (b) The fraction of ground state structures out of all structures sampled at each temperature are plotted and fitted with a Gaussian distribution ($\mu = 1158$ K, $\sigma = 76$ K). (c) If we fix the temperature at 1000 K, there is significant variation in energy in the structures for a fixed number of MD steps, and a sufficiently large number of steps (≥ 5000) is required to obtain the ground state. ΔE_{gb} is the range of energies at each increment of steps. (d) ΔE_{gb} is plotted against the number of MD steps and fit with a least-squares regression line.

Figure 4.4 and Figure 4.5). Figure 4.6a shows the range of E_{gb} as a function of temperature when the MD duration is fixed at 10^5 steps, and only at intermediate temperatures does the algorithm find the ground state structure. From this data we can compute the probability of finding the ground state structure (calculated as the fraction of ground-state structures out of all sampled structures at each temperature), which peaks at approximately 1150 K and is zero for very low and very high temperatures (Figure 4.6b). Analogously, simply choosing an optimal temperature (e.g., 1000 K) is insufficient, as too few MD steps will never achieve the ground state (Figure 4.6c). The spread in E_{gb} generally decreases as the MD duration

increases (Figure 4.6d), but as the optimal parameters are not known a priori, it is important to properly sample both temperature and duration in order to access all GB structures.

The energy calculations presented here use empirical IAPs, but other techniques such as DFT can be used as well, as those calculations are decoupled from the structure optimization steps; however, the use of IAPs enables us to access temperature-stabilized phases, low-angle GBs, and larger reconstructions with thousands of atoms. This methodology takes advantage of the increasing availability of computational resources and the advent of high-fidelity machine-learned (ML) IAPs will enable quantum-accurate atomistic simulations of large systems with extended defects [202]. Advances in sampling and structure generation algorithms (e.g., generative ML [203]) will further expand the diversity of results and the modular structure of the code enables different choices for the sampling algorithm to be easily plugged in. Of particular interest would be extensions to multicomponent systems, which could be handled using a MC approach [57–59] for compositional DOF and would enable grand canonical sampling of GB structures in technologically important alloy chemistries.

Chapter 5

Data-centric materials science and engineering education

The work presented in this chapter was published in two shorter, peer-reviewed manuscripts. The first was published as a Technology Report titled "Using Jupyter Tools to Design an Interactive Textbook to Guide Undergraduate Research in Materials Informatics" in the *Journal of Chemical Education* **99**, 10 (2022) by Enze Chen and Mark Asta [204], and the second as a conference paper titled "Integrating programming-based modules into a materials characterization laboratory course to reinforce data science and scientific writing" in the *Proceedings of the 2023 ASEE Annual Conference & Exposition* by Enze Chen, Mark Asta, and Andrew A. Minor [205]. Both works are reproduced here with permission of the co-authors.

5.1 Introduction

In recent years, the infusion of machine learning (ML) and artificial intelligence (AI) into the materials and chemical sciences [17] has led to significant advancements in modeling structure–property relationships [206], synthesis planning [207], molecular design [208], and drug discovery [209]. Scientists have leveraged ML/AI technologies to design new materials with the ultimate goal to achieve lower cost, accelerated time to market, and greater sustainability. Now, the maturation of these technologies in research present exciting opportunities to transfer them into the classroom, where educators can design new informatics curricula to train the next generation of informatics-skilled scientists [25, 26, 210–212].

With such pedagogical aspirations serving as crucial motivators, it is also important to identify the key challenges that could impede a successful implementation. The multidisciplinary nature of chemoinformatics calls for a robust curriculum that combines scientific domain knowledge, data science, and computing skills, which is difficult for students to fit into an already strained course load and for an instructor to gain mastery in all components [211]. Due to the comprehensive scope of materials informatics (MI), it has been

advised that MI curricula be structured as small modules to facilitate its integration with existing courses and to help alleviate some of the aforementioned problems [25, 26]; however, this integration presents its own challenges that must be carefully considered during instructional design. For example, it may not be obvious how a research-based tool can fit within the scope of the existing curriculum and whether additional software-specific training is necessary.

In particular, when designing any informatics curriculum, it is appealing to include programming exercises to give students hands-on experience with authentic problem solving tasks [213]. Given the popularity of the beginner-friendly Python programming language in scientific research [214], many intro-level programming courses in the physical sciences have chosen to use Python as well [215,216]. To improve code readability and reproducibility, there is a growing trend to write and execute Python code inside Jupyter notebooks [80], which are now commonplace in chemoinformatics/MI courses [81–83] and workshops [217–220]. These digital notebooks merge prose, Python code, and additional multimedia elements into rich computational narratives that provide a gentle introduction for students. They are often provided as standalone files (`.ipynb` extension) for users to run locally, but this requires installing additional Python packages and managing software configurations on personal devices [221], which can be daunting for beginners. To circumvent these technological challenges, another popular option is to execute notebooks in the cloud [218, 222], where popular services such as Binder [223] and Google Colaboratory (Colab) [92] can provide a more consistent and equitable user experience. Several of these options have been compared in the literature [224, 225], which underscores the impact of cloud-based computing on education.

5.2 MI summer internship

We first report our experience using a related suite of Jupyter authoring tools, including the recently developed Jupyter Book environment [87], to design an integrated, interactive online textbook to guide undergraduate research projects in the data-driven design of dielectric materials [226]. This work was inspired by existing textbooks published using Jupyter Book [89,227], which gave us confidence to reenvision the remote research experience in light of the COVID-19 pandemic and utilize an interactive text to boost student engagement and performance [228, 229]. Given the multitude of benefits of undergraduate research experiences [230, 231], the curriculum served as an experimental model to expand access to such opportunities while easing students into the research process. We discuss the details of the implementation that uses open-source tools familiar to those working in the computational and informatics sciences, which lowers the barrier for researchers to design new curricula. The Jupyter Book interface was easy to navigate and the ability to use JupyterHub [91] to run Python notebooks in the cloud enabled students to interact with the content entirely through a web browser without having to install anything locally. While the COVID-19 pandemic forced everyone to transition online, it also presented an opportunity to explore how the digital nature of informatics and rapid advancements in computing infrastructure

can improve the scalability and accessibility of undergraduate research experiences.

5.2.1 Implementation

The primary software used in designing this curriculum is Jupyter Book, developed by the Executable Books Project as "an open source project for building beautiful, publication-quality books and documents from computational material." [87] It can be installed using the standard Python package managers `pip` or `conda` and contains a set of command-line utilities for compiling the book from Markdown text (`.md`) and Jupyter notebook (`.ipynb`) files. Building off of Markdown files is advantageous due to the familiarity of the Markedly Structured Text (MyST) syntax [88] to researchers and the simplicity of the syntax, which is still powerful enough to create rich content pages with text, figures, citations, and embedded files and videos. Content that cannot be easily created in Markdown (e.g., presentation slides) can be created externally and then embedded into Markdown files using standard HTML tags. All of the content (presentations, lecture notes, Python notebooks, etc.) for each day was then grouped to form each "chapter" of the book. In this way, the digital textbook served as a centralized location where students can go to access all of the resources throughout the research internship. Once the pages are complete, the book can be compiled into a static PDF file for offline reading or a collection of HTML web pages for dynamic viewing, all without the need for instructors to learn additional frameworks like JavaScript or CSS. The online book can be accessed using any modern web browser, which frees instructors and students from compatibility concerns across different operating systems and Python environments. Further information on getting started with Jupyter Book can be found in the detailed tutorials on their website [87], which requires familiarity with using the command line and text editors. In addition to detailed instructions on setup and content creation, the documentation describes several options for sharing the book, which is also discussed below.

Section	Topics covered
Preamble	Module overview, Software setup
Week 1	Intro to Python/Jupyter, Intro to MI, Data management, Intro to ML, Dielectric materials, and MI research
Week 2	Self-directed research
Week 3	Self-directed research, GitHub pull requests, Project presentations, Reflections on the module

Table 5.1: Summary of the Jupyter Book contents. The first week is a series of tutorials to introduce students to the research topic and software tools, and the remaining time is for their self-directed research projects. (MI = materials informatics, ML = machine learning)

Table 5.1 summarizes the contents of the book, which was used by six undergraduate students for a research module on the data-driven design of dielectric materials for microelectronics applications. The module was part of a summer internship program supported through a partnership between the College of Engineering at the University of California, Berkeley and the Materials Sciences Division at Lawrence Berkeley National Laboratory. Briefly, students learned data management best practices and applied them to mine data for dielectric materials from the Materials Project [232, 233], an online database for materials properties computed using high-throughput density functional theory (DFT). Students used this DFT-generated data to train ML models to predict a material's dielectric constant (κ) and then used the ML model, along with other criteria of their choosing, to screen for the "best" high-κ dielectric material for nanoscale transistors. This research question is an active area of research [234, 235] with documented impact for the microelectronics industry [236, 237]. The content also aligned well with the two other modules in the summer research internship, which shows how a modular design can be flexibly sequenced with other curricula, consistent with the recommendations of other educators [22, 25, 26]. All files are freely available on GitHub under a Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) license and the textbook is hosted for free on GitHub Pages [226]. GitHub Pages was chosen for its simplicity and flexibility, and step-by-step instructions for hosting Jupyter Books on a variety of platforms can be found in the documentation [87].

The first iteration of the internship was structured to feature a full week of tutorials about MI followed by two weeks of self-directed research in teams, all performed remotely in Summer 2021. The text begins with an overview of the research module, including the learning outcomes, schedule of events, and account setup instructions (e.g., Materials Project, GitHub). Most of the Python notebooks are introduced in the first week, starting with a complete introduction to the Python language and followed by tutorials on data management, data visualization, ML, dielectrics, and MI research. Weeks 2 and 3 have fewer prescribed notebooks and more supporting content such as additional reading, check-in discussions, and presentation tips for students. Interspersed throughout the weeks are mini-lessons on how to read papers and documentation, graduate school applications, and other topics to support their educational development. There are no graded assessments or exams, and the final summative assessment was a presentation of their research and experience. While live mentoring took place over instant messaging and video conferencing platforms, students used the Jupyter Book interface to access all lecture content, assigned readings, and Python exercises (Fig. 5.1). The minimalist style of the web interface features several navigation shortcuts and a familiar rendering of Markdown and notebook content that facilitated students' daily use. One can use arrow keys at the bottom of each page to proceed sequentially through the book or use the navigation headings to jump to specific pages. The MyST syntax makes it simple for instructors to incorporate content from external sources by embedding them in the page or linking them, as shown in Fig. 5.1.

Several interactive options exist for readers to write and execute Python code within the notebooks. UC Berkeley affiliates can use the campus JupyterHub [238] to open the same web page in a Python kernel (by hovering over the rocket icon in the top of Fig. 5.1). The

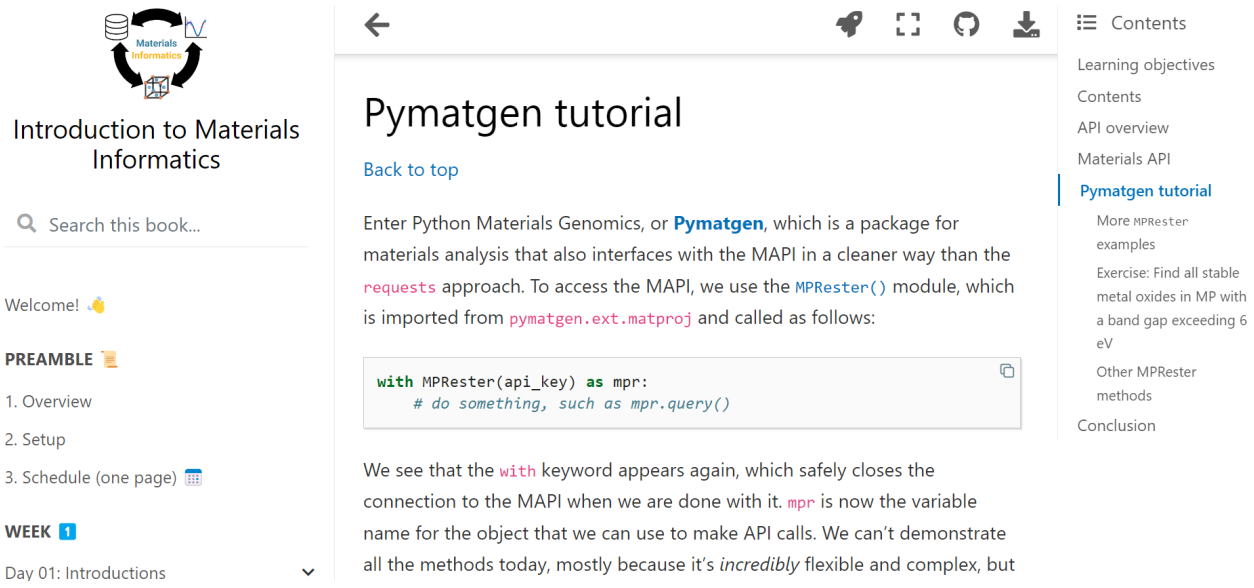


Figure 5.1: Screenshot of the Jupyter Book web interface [226]. The left margin contains a table of contents and a search bar for ease of navigation through the entire book, while the right margin contains section headers for the specific page. The content in the page, here rendered from a Jupyter notebook [80], contains a mix of text, Python code, and images. Hovering over the rocket icon in the top-right lists options for launching the same page in an interactive environment such as JupyterHub [91] or Google Colaboratory [92].

JupyterHub instance syncs with the GitHub repository for the Jupyter Book so that each student has an updated copy of the files that they can work on at their own pace. There are many advantages to using the JupyterHub platform, including a uniform computing environment (required packages can be pre-installed), more powerful resources (extra compute and/or memory can be allocated upon request), and preservation of their work (the modified files are saved to their account). These are significant benefits of JupyterHub that can only be partially satisfied with other cloud services, although a large-scale deployment of JupyterHub may require institutional support and funding (see refs. [91, 238] for deployment details). For non-UC Berkeley affiliates, there is also a Colab option that allows any user to complete the exercises with minimal changes (e.g., a few package installations and file paths), and this may be preferred if teaching at scale without JupyterHub. Nevertheless, the seamless integration of Jupyter Book with cloud-based Python kernels is one of the principal advantages of using the Jupyter Book software to design MI curricula as it enhances the scalability and accessibility of these interactive learning experiences.

5.2.2 Results and Discussion

This curriculum was taught to students from a variety of backgrounds: Some of them were new to Python (but had other programming experience), many were new to research, only one had a materials science background, and all were new to data science. These individual differences further validated the use of the Jupyter ecosystem to design tutorials during the first week to establish the requisite knowledge to carry out a successful MI research project. To maximize effective usage of the software, a walk-through of the Jupyter Book and JupyterHub interfaces was given on the first day, which is recommended when teaching with any new technology. Subsequent usage included a mix of group instruction (e.g., presentations, live coding) and individual work using the digital textbook. In addition to the final presentation, feedback was collected continuously throughout the internship in the form of daily check-ins and three anonymous surveys (made using Google Forms) distributed at the beginning, middle, and end (see Supporting Information for example questions).

Consistent with our expectations, the students found the Jupyter Book straightforward to navigate and helpful as a centralized resource for the entire module. No longer did they have to juggle multiple files and links (only the single URL to the online text), and could focus on learning the material. They enjoyed having the interactive exercises alongside the reading to strengthen comprehension and welcomed the opportunity to carefully work through the exercises at their own pace, demonstrating a greater sense of agency. By the end of the second week, most of the students were not only comfortable editing the existing notebooks, but also capable of creating new Jupyter notebooks on the JupyterHub to carry out their self-directed research. Their feedback on the merits of the Jupyter Book-powered curriculum aligns with the results reported in previous research on the benefits of interactive textbooks [228, 229].

Figure 5.2 shows the results from the final student survey on how well the curriculum achieved the learning outcomes of the module. There was a strong level of agreement, consistent with our observations of their final presentations (e.g., hearing comments about the importance of cleaning data, or how ML can be a helpful guide for experiments—not a replacement). While these learning outcomes are not directly tied to the software, the success of this module would not have been possible without the structure and interactivity enabled by the Jupyter Book. Giving students hands-on practice with informatics workflows and open-ended, group research projects are authentic assessments that cultivate problem solving skills in realistic contexts [213], which will better prepare them to meet the demands of the next-generation workforce. Although the impacts of this internship model will be evaluated in a separate communication, there were a few pieces of feedback directly regarding the software. As the students worked in groups on their research projects, a few of them wished that JupyterHub had the ability to collaborate in real time on the same notebook. Such a possibility was not explored for this iteration, although the suggestion of pair programming [239] through video conferencing software was found to be helpful for sharing programmatic solutions and strengthening group dynamics. Another collaboration tool available in Jupyter Book (that was not used) is Hypothesis [240], which allows users

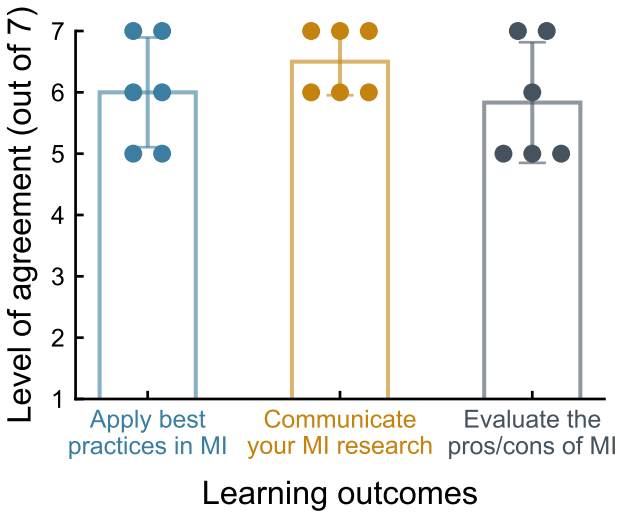


Figure 5.2: Student feedback on how well the module's learning outcomes were achieved. Questions are scored on a Likert-type scale where 7 indicates strong agreement and 1 indicates strong disagreement. The bars and lines indicate the mean and one standard deviation ($n = 6$), respectively, and the individual scores are separated for clarity.

to directly annotate the HTML for asynchronous group discussions, particularly if Jupyter Book is used for a course that involves more reading. Many students also desired the capability to automatically check the correctness of their code, which might be possible with auto-grading software [241, 242]. Such tools would definitely be valuable if Jupyter Book is used to design a large-scale, semester-long course where graded assessments carry greater importance. These suggestions, in addition to others like better scaffolding of the tutorial exercises, will help focus curricular improvements for future iterations of this module.

From the perspective of instructors, we are particularly excited to see computational software and infrastructure mature to the point where an entire MI research curriculum can be implemented online and taught remotely. These open-source, online platforms markedly increase the accessibility of this knowledge and continue to lower the barrier for new learners, whether that involves better content, more effective pedagogy, or fewer headaches with software installation. The fact that participants were gathered across three time zones shows how cloud-based solutions can scale far more effectively than traditional materials science education can, reaching larger and more diverse populations. The synergy between the Jupyter environment and robust GitHub workflows also lowers the barrier to creating educational content for scientific domain experts, as they can continue to use familiar tools from their research background. Moreover, creating open-source curricula invites participation from the broader community to collaborate on their development [210], as instructors can adapt these materials for their own needs or add chapters based on their areas of expertise. The use of GitHub presents an opportunity to teach students about team-based open science [243] and

each student was assigned to make a small contribution to the Jupyter Book to showcase their research accomplishments. Although none of the students had prior experience with `git`, they were able to successfully clone the repository, update their presentation abstracts, and submit the changes through pull requests, demonstrating the facility of collaborative development with Jupyter Book. We anticipate the increasing prevalence of Jupyter Books (for other examples, see refs. [89, 227] and the public gallery [87]) and other open-source computational tools in instructional design will continue to bolster teaching and learning of informatics in the materials and chemical sciences.

5.3 MSE 104L data analysis

Building on the success of our summer research program, I wanted to integrate some of those same modules into an existing course to test its flexibility and expand its impact. As we prepare our students to enter the workforce, we have to equip them with not only deep domain expertise, but also several transferable skills for them to succeed at their jobs [244]. For this intervention, we focus on two of these skills in particular, data science (DS) and scientific writing (SW), which have been discussed in recent reports from the National Academies [245,246], ABET [247], and university educators [204,248–250]. These reports collectively highlight the importance of DS and SW in engineering practice and identify opportunities for students to build these skills through courses and programs, but it remains unclear how we can add more topics into an already-packed materials science and engineering (MSE) curriculum. While many schools offer dedicated DS and SW courses, these courses are often not required and lack examples in the MSE domain, which can leave some students unaware of the applicability of these skills in MSE.

We believe that there is an opportunity, given advancements in computing software and STEM pedagogy, to better integrate DS and SW practices into the MSE curriculum. Many institutions (including UC Berkeley) require an introductory computing course in their engineering curriculum, which provides students with a general introduction to algorithms and computational thinking. This is the foundation on which we introduce DS concepts, facilitated by open-source software such as Python and Jupyter to enhance the accessibility and scalability of this knowledge. Instead of using canonical problems and datasets, we teach these tools using real experimental data collected by undergraduates in an upper-division materials characterization laboratory course, MSE 104L. In MSE 104L, students perform a series of experiments (see Table 5.2), analyze the data they collect, and write a lab report interpreting their data for each experiment. Student feedback from previous years indicate a desire for more support on data analysis and report writing (partly because the laboratory sessions are focused on machine operation and data collection), which motivated us to design programming-based modules that could easily integrate into the post-experiment procedures, similar to what other instructors have done for physical chemistry labs [82,215]. The existing course structure remains unchanged and provides a natural environment for modernizing the MSE curriculum as students learn DS and SW skills in context.

Lab #	Experiment / Topic	Python Notebooks
	Intro to modules	- How to use Jupyter Book - Intro to Python
1	X-ray emission	- Intro to plotting - Lab 1 exercises (new: plotting and curve fitting)
2	Powder X-ray diffraction	- Intro to tabular data - Lab 2 exercises (new: pandas DataFrames)
3	Precision X-ray diffraction	- Lab 3 exercises (new: functions and correlations)
4	Energy-dispersive X-ray spectroscopy & Scanning electron microscopy	- Lab 4 exercises (new: annotations and clustering)
5	Self-guided experiments	[None]
6	Transmission electron microscopy	[None]

Table 5.2: Outline of MSE 104L. We designed new Python-based learning modules corresponding to four out of the six laboratory experiments.

5.3.1 Implementation

Table 5.2 shows the six experiments in MSE 104L and the corresponding modules for each one. Each set of exercises is Python-based and written in a Jupyter notebook [80], which we again compile using the Jupyter Book software [87] and host online for free access [251]. As DS is a broad discipline, for this first iteration we emphasize *data visualization* as it is an important skill in MSE that can easily be misused [252]. Using figures to help structure the narrative is also an important skill for professional communication [253], which we hope enables our modules to support SW development from a "data-first" approach. Past student performance on lab reports in MSE 104L has shown some weaknesses in terms of data interpretation, visualization, and organization, which are consistent with previous reports [248, 249]. We create a series of scaffolded exercises, following the sequence of the laboratory experiments (Table 5.2), that teach students how to use the versatile Python visualization library Matplotlib [254]. In addition to specific exercises corresponding to each lab report, there are also general tutorials at the beginning for those less familiar with the programming environment. We include discussions of best practices in data visualization and additional exercises on tabular data (e.g., solving for lattice parameters in X-ray diffraction), curve fitting (e.g., Nelson-Riley method [255]), and k-means clustering (e.g., image segmentation). In doing so, we hope to expose students to common scientific computing

libraries such as NumPy [79] and pandas [256] so they are aware of how these powerful tools can support their experiments and deepen their understanding of the science. That being said, to avoid overburdening students in this initial rollout, we make the modules optional and let the students decide which tools they wish to use for data analysis in the labs.

At the start of the semester, we explained to all students in MSE 104L the purpose of the learning modules and the details of this research study, which qualified for exempt status through the UC Berkeley Institutional Review Board (CPHS #2022-10-15691). Students are surveyed anonymously at the start of the semester to assess their academic preparation and predispositions toward DS and SW, and then they are surveyed a second time (not linked) near the end to assess the effectiveness of the learning modules and any changes in their beliefs. The two surveys contain a mix of multiple-choice, 7-point Likert-type scale (7 being "Strongly agree"), and short-answer questions whose responses are inductively coded by me. For this study, we seek to answer the following research questions (RQs):

RQ1: How well are DS and SW skills integrated into the current MSE curriculum?

RQ2: How do MSE students view DS and SW in the context of their work?

5.3.2 Results and Discussion

We will only report and discuss the most salient results in this section, and the rest can be found in Appendix C. All 42 students in MSE 104L consented to participate in the study, with 88 % being juniors and above, and 88 % majoring in MSE. Over half of the students were male (60 %) and a little over a third were female (36 %). Other data such as race and ethnicity were not collected and the full demographic results are found in Supplementary Table C.1.

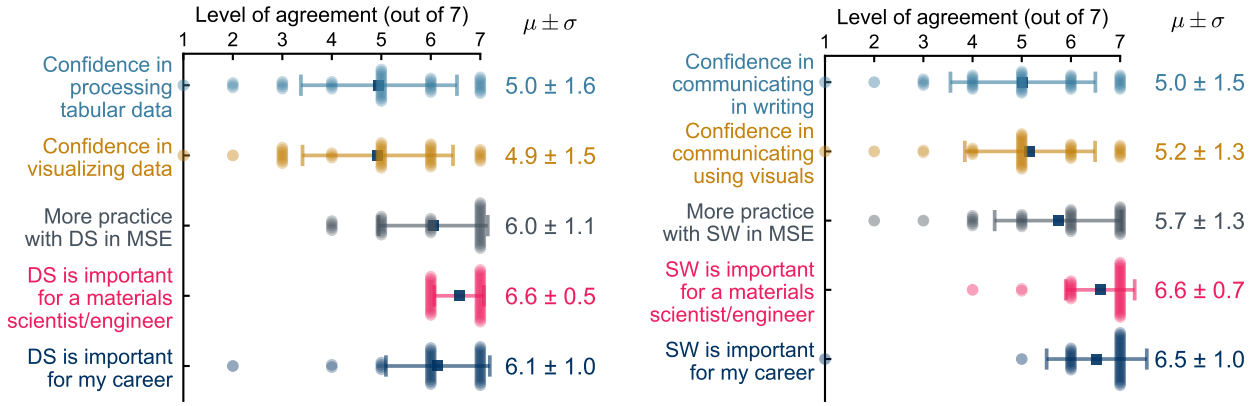
5.3.2.1 Pre-course survey

Subject	Yes	No
Computing	**33** (**79%**)	9 (21 %)
Data science / Statistics	7 (17 %)	**35** (**83%**)
Scientific writing	3 (7 %)	**39** (**93%**)

Table 5.3: Previous or concurrent coursework taken by the students ($n = 42$).

Table 5.3 shows the academic background of these students, which is largely consistent with our expectations. Nearly 80 % of the students have taken a computing course, but only 17 % of them have taken a DS or statistics course. Even fewer had taken a formal course in SW and it is likely that their only prior exposure to technical writing was in the laboratory component to the gateway MSE course. MSE majors at our institution are required to take a 4-unit introductory computing course, but there is no strict requirement on the timing.

Notably, of the nine students who had not taken a computing course, all of them answered "No" to the other two questions, eight of them were male identifying, and none of them were sophomores. This suggests that there does not appear to be a gender disparity in terms of access to computing courses in this sample of students and that later class years may see a benefit to taking computing courses early.



(a) Data science (DS) **(b)** Scientific writing (SW)

Figure 5.3: Pre-course survey results for student sentiment on (a) data science and (b) scientific writing. Students ($n = 42$) indicated their level of agreement with five prompts on a Likert-type scale (1 to 7, with 7 being "Strongly agree"). Each square marker and associated error bars correspond to the mean (μ) and one standard deviation (σ), respectively, and each dot is an individual student response.

Supplementary Table C.2 details the programming languages used by students in their MSE courses *only*, from which it is clear that MATLAB is the most common required programming language (55 % of students were required to use it at least once) followed by Mathematica (31 %) and Python (26 %). It is interesting to note that Python is the only programming language where the most common reason students used it in MSE courses was by "personal choice" (40 %), although roughly the same proportion of students (38 %) had never used it in their MSE courses. These results support our hypothesis that students generally lack formal training in DS and SW skills and that appropriately scaffolded content in Python would be beneficial toward developing these capabilities.

The two sets of Likert-type scale questions on DS and SW gave similar results (Figure 5.3). Student confidence regarding four DS and communication tasks hovers around 5/7, indicating potential for improvement. In spite of this (or because of it), students wish to see more opportunities to practice DS and SW skills in their MSE courses as the current amount may not be enough (with respect to RQ1). They also tend to regard both DS and SW as important skills for materials scientists and their own careers (all greater than 6/7 in agreement, consistent with previous studies [257]), which is encouraging to see and signals to us that our learning modules may be appropriately targeted for this audience (RQ2). The

first two questions for each topic in Figure 5.3 have a larger spread in the responses, and we note that the two choices for "1" for the two DS questions are selected by the same student, but a different one from the single student who selected the two choices for "1" for the two SW questions.

We asked students to elaborate on their agreement with the importance of DS and SW (RQ2), and we report the coded results in Table 5.4. A majority of the students discussed the importance of these skills in the context of scientific research, such as using DS tools to analyze experimental results or using SW to communicate findings. The data-related topics that appeared in the responses to the DS question mirrored those in students' definition of DS, although a greater proportion of students mentioned the importance of handling large volumes of data. It is not surprising that a majority of students felt SW was important for communication and their future careers, although it is interesting to note that fewer students felt that SW was an important transferable skill (11 to 3), whereas for DS the order of career vs. transferable skills was reversed (3 to 7).

5.3.2.2 Post-course survey

We surveyed the students a second time towards the end of the course and show the results from the same set of Likert-type scale questions in Figure 5.4. Unfortunately, we were only able to obtain 29 student responses in this survey instead of the full population of 42 students. Nevertheless, it appears that there is a noticeable increase in average student

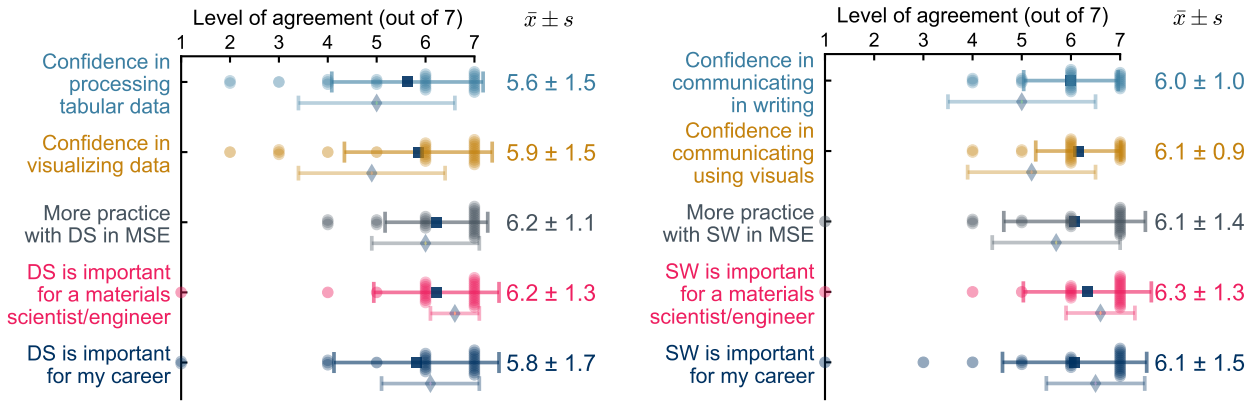
Theme	Topic	Counts
Academic (27)	Research/expt.	15
	MSE	8
	Courses	4
Data-related (24)	Understanding	10
	Volume	7
	Visualization	5
	Communication	2
Life (18)	Transferable skill	7
	Easier life	5
	Applications	3
	Career skill	3

(a) Data science (DS)

Theme	Topic	Counts
Academic (20)	Research/expt.	19
	MSE	3
	Courses	1
Actions (34)	Communication	27
	Understanding	6
	Instruction	1
Life (15)	Career skill	11
	Transferable skill	3
	Enjoyment	1

(b) Scientific writing (SW)

Table 5.4: Inductively coded student responses ($n = 42$) to the question, "Why are these skills important (or not important) to you?" for (a) Data science and (b) Scientific writing. Topics were first identified (possibly multiple in a single response) and then grouped into themes.



(a) Data science (DS) **(b)** Scientific writing (SW)

Figure 5.4: Post-course survey results ($n = 29$) for student sentiment on (a) data science and (b) scientific writing. The diamond marker and second set of error bars correspond to the mean and standard deviation in the pre-course survey results (Figure 5.3) for ease of comparison.

confidence in DS and SW skills with gains between 0.6 and 1.0 points out of 7. Moreover, the average scores for the questions asking for more practice with DS/SW effectively stayed the same, which suggests that more integration of DS/SW topics into the MSE curriculum would be welcomed by this audience (RQ1). Supplementary Figure C.1 stratifies the results in Figure 5.4 by those students who used and did not use the modules, which numbered 18 and 11, respectively. It is interesting to note that among these two groups there is a large difference in confidence when it comes to DS skills but almost equal confidence in SW skills. This provides some evidence that our modules were effective at promoting DS but less effective at promoting SW; however, because our experimental design did not link the survey responses, we caution against drawing stronger conclusions of causation. Additionally, some sampling bias may exist as we were unable to survey the entire class in the post-course survey, potentially missing out on capturing students who had an adverse experience with the modules (among other reasons). These shortcomings can be addressed by improving the experimental design in future iterations, which will also be informed by the following observations.

For the ($n_{\text{used}} = 18$) students who used the learning modules, six of them used all four modules and overall usage was higher for the first three labs (Supp. Table C.3), which aligned with our expectations. Thirteen of those students had never taken a DS course and were likely using Jupyter and Python for the first time. As shown in Supp. Table C.4, the top motivation for these 18 students was the belief that the modules would help with writing the reports and other primary motivations included the belief that they would support learning MSE and Python, and that the modules were easy to use. It is encouraging to see from another set of Likert-type scale questions (Figure 5.5) in the post-course survey that students had very positive experiences with the accessibility and utility of the programming-

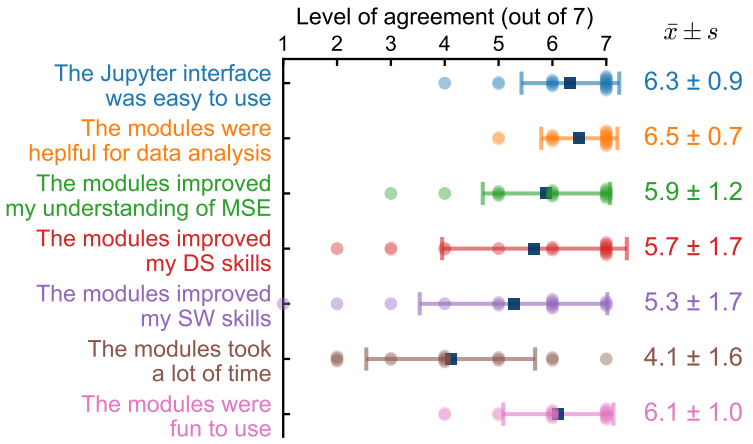


Figure 5.5: Post-course survey results for student sentiment on the learning modules ($n_{\text{used}} = 18$).

based learning modules. The questions explicitly asking for improvement in DS and SW skills had lower ratings and larger variations, which suggests more improvements could be made to support a greater range of learners, especially when it comes to written communication. Supp. Table C.5 reveals that many students felt that some sections were lacking in detail and we acknowledge the challenges of using a new UI like Jupyter Book/Hub for the first time. Students felt like the modules took a moderate amount of time, although a more informative assessment could be to compare the amount of time it takes to do data analysis with these modules versus their personal choice.

In fact, personal preference for another method of data analysis was the most common reason given among the ($n_{\text{not}} = 11$) students who did not use any of the modules, followed by the fear that the modules would be too time consuming (Supp. Table C.4). Most of them also didn't provide ways to encourage adoption (Supp. Table C.5), but those who did proposed giving a tutorial at the beginning of the course or making the modules a required component of the course. The first suggestion can be implemented by giving a demonstration during the lab sections or even embedding tutorial videos into the Jupyter Book for asynchronous viewing. The second suggestion may very well be a motivation issue [257], but in light of the positive impacts of our learning modules and the large-scale redesign as demonstrated at other institutions [258], there are several merits to including these modules into the course learning objectives more broadly. By introducing these skills earlier and more systematically, we can increase the impact of these tools and better quantify learning gains as we equip students with the necessary skills for professional success.

From this multi-methods study, we find that MSE students have a strong desire to learn DS and SW in the context of their work and the modules we designed are effective at guiding this development. All of the open-source learning modules are freely available online [251], which underscores the scalability and accessibility of this approach, and more exercises can be added in the future in a straightforward manner. The Jupyter ecosystem makes it easy

to integrate interactive learning experiences into existing MSE courses in a streamlined way, and the barriers for instructors to acquire new skills [259] may be mitigated by the growing adoption of open-source tools and cooperative efforts between instructors.

Chapter 6

Conclusion and outlook

I began this dissertation by asking "What will the future of MSE look like?" and proceeded to show how computational tools are important drivers of this transformation for both research and education. Specifically, I focused on two examples of atomistic simulation workflows for alloy design: High-throughput calculations of APB energies in Ni_3Al-based alloys (Chapter 3) and grand canonical optimization of GB phases in α-Ti (Chapter 4). These workflows are built on established theory and existing tools, and they allow us to generate large amounts of data on planar defects in metal alloys. To benefit the community, we have shared all of our data and tools open source, including our use of Jupyter books to transfer research tools into the classroom to modernize the curriculum, engage students, and improve learning outcomes (Chapter 5). I am excited by the future possibilities enabled by these tools, and below I discuss two possible extensions of the work presented in the previous chapters.

6.1 Open-access GB database

With increasingly available computational resources and a growing open-data movement [260], many researchers have shared public datasets from high-throughput atomistic simulations of GB structures [261–267]. Due to limitations in computing resources or methodologies, their scope has been restricted to a few elements or only minimum-energy structures, and all of them employ the γ-surface method, potentially deleting atoms only when they overlap [261–263]. Additionally, with the exception of a few cases [266, 267], all of the previous datasets are stored in static CSV or ZIP files, making them challenging to access and expand in the future. Nonetheless, these early contributions have proved to be extremely useful to the community, underscoring the need for more comprehensive databases of GB structures.

Using a framework similar to that presented in Chapter 4, there exists an opportunity to use the GRIP tool to systematically explore GB structures across a diverse set of elements and configurations. The resulting data can be filtered and uploaded automatically to one of the many databases that are targeted for MSE [268–270], facilitating access, maintenance, and growth of the data. Table 6.1 gives an example of the different parameters one could control

Independent parameters	Approx. number of options
Element	20
Interatomic potential	3
Tilt/twist boundary	2
Axes	3
θ or (hkl)	50
n values	30
Structures per n	20
Total	10 800 000

Table 6.1: Diversity of GB database built using GRIP. For each element, we could generate tilt and twist boundaries for multiple axes, misorientation angles, and atomic densities. The same procedure would be repeated for different interatomic potentials (IAPs). Note these are only estimates and assumes a symmetric GB structure in a unary system.

in building out the database, which can total millions of unique, grand-canonically optimized structures even when restricted to unary systems and symmetric GBs. This database would be significantly larger and more diverse than any existing database to date, and we can take advantage of mature APIs to allow users to easily select the subset of data most useful for them (similar to how we shared the APB energy data on MPContribs [129]). Longer term, the database should be flexible enough to handle interfaces in multicomponent systems—an obvious and highly desirable extension of GRIP—which present additional compositional and structural DOF that must be controlled for and quantified (e.g., potentially multiple n_i).

These grand-canonically optimized GB structures can be used for a variety of downstream tasks, such as studies of mechanical behavior [271] and GB segregation [272]. The structures would also be interesting to explore in their own right, as one can screen/mine for trends in excess properties and engineer new descriptors for characterizing GB structures and environments [273]. Along those lines, such structures may also be valuable training data for ML IAPs [202], as they provide a tremendously diverse set of local configurations. This could help improve the accuracy of ML IAPs for simulating extended defects, which traditionally perform poorly in these scenarios when the defective structures differ from the bulk structures that comprise a majority of training data. As a final thought, just as we leveraged the Materials Project database for our summer internship curriculum, I am also intrigued by the possibility of incorporating the database into other academic curricula and course-based research experiences.

6.2 Universal design for learning

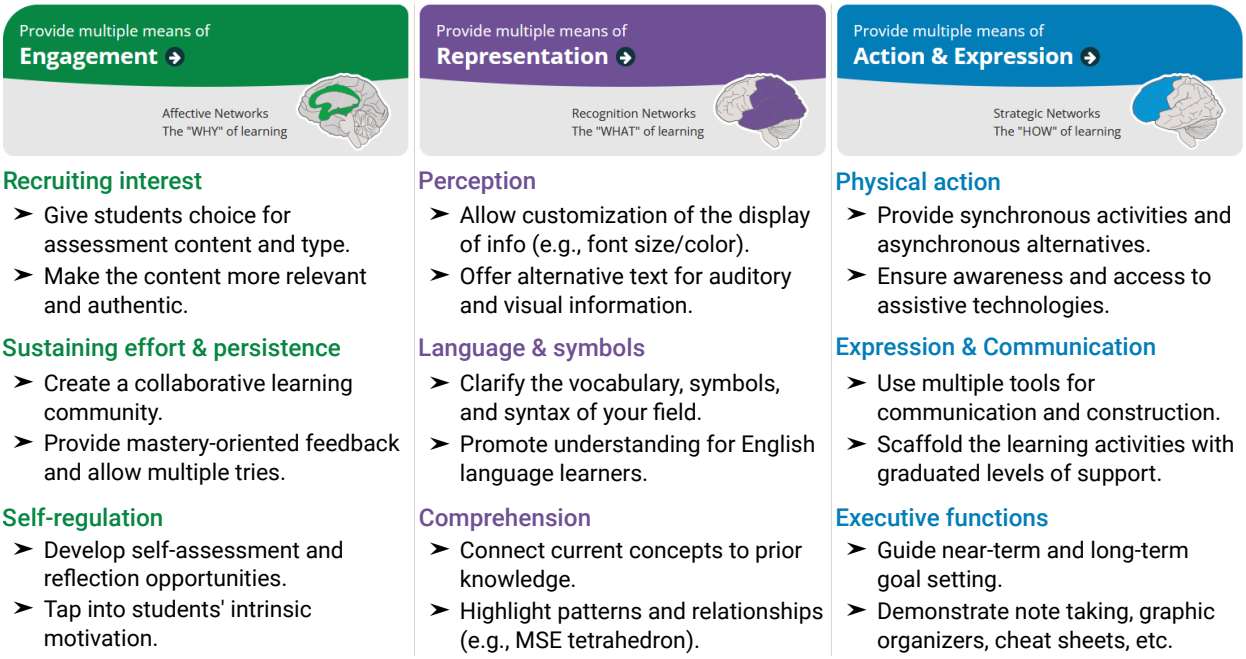


Figure 6.1: Universal design for learning (UDL) guidelines. UDL is based on the overarching principle of providing multiple ways of learning, with the three main guidelines of Engagement, Representation, and Action & Expression. The sub-topics and a few actionable examples are provided. The content is adapted from ref. [274].

While technology is not a panacea for education, and may in fact do as much harm as good [275], I believe an intentional implementation coupled with targeted learning outcomes will make our MSE curricula more authentic and accessible. In the equity sphere, social classes such as race and gender continue to be the primary focus (and for good reason!), but a less visible area that is gaining more attention recently is (dis)ability and neurodiversity [276]. Similar to other dimensions of diversity, neurodiverse students fall under a broad umbrella of abilities and conditions, some common ones being autism, dyslexia, and attention-deficit/hyperactivity disorder. These students can bring new perspectives and lived experiences into the learning environment, and it is important for instructors to understand how to tap into these strengths. Globally, the World Health Organization estimates that 1.3 billion people, or 16 % of the global population, live with some kind of disability [277], several of which may be invisible. Therefore, if we can make our curricula more accessible for students with physical and learning disabilities, then we can engage an even larger segment of the population and make our pedagogy more effective.

One framework for addressing this need is universal design for learning (UDL), which provides a set of guidelines for designing flexible curricula to accommodate individual learn-

ing differences [274, 278]. Instead of a traditional approach where the instructor creates a fixed curriculum that individual students are asked to adapt to (perhaps with academic accommodations from disability services on campus), UDL asks instructors to proactively design a flexible curriculum that gives students more options to learn in a way that best suits them. It has strong foundations in cognitive neuroscience [278] and relies on the overarching principle of "multiples," including providing multiple means of Engagement, Representation, and Action & Expression. Figure 6.1 shows the three core themes along with sub-topics and actionable examples for each one. With proper implementation, a UDL-based course will not only make neurodiverse students feel more included, but it would make all students perform better.

What may be less clear from Figure 6.1 is the central role technology plays in enabling many of the goals of UDL. In many respects, the Jupyter ecosystem is a natural fit for the UDL framework, as our previous work already aligns with several of these guidelines, including authenticity (e.g., data science) and communication (e.g., oral presentation). While that work pertained to new informatics curricula, I envision Jupyter and Python being useful tools for learning more traditional MSE content as well. For example, one can create interactive learning modules for studying different crystal structures [279], which is a topic that students are known to struggle with [280]. This and many other MSE topics naturally lend themselves to the Representation principle, where we can express the same idea using text, equations, visuals, and interactive (digital) demonstrations. Technology can also transform the way content is delivered and interacted with, such as enabling rapid auto-feedback mechanisms that are critical for learning, and yet difficult to achieve in MSE. Python-backed tools such as PrairieLearn [281] leverage random question generators to give students repeated attempts with different types of questions, allowing them to work on a concept until they achieve mastery [282]. While some of these implementations require significant time investment and changes up front, I believe they have the potential to dramatically change teaching and learning in MSE for the better.

Bibliography

[1] Merton C. Flemings. What next for departments of materials science and engineering? *Annual Review of Materials Science*, 29(1):1–23, 1999.

[2] Richard J. D. Tilley. *Defects in Solids*. Wiley, Hoboken, NJ, 1st edition, 2008.

[3] Adrian P. Sutton and Robert W. Balluffi. *Interfaces in Crystalline Materials*. Oxford University Press, New York, 1st edition, 1995.

[4] Roger C. Reed. *The Superalloys: Fundamentals and Applications*. Cambridge University Press, New York, NY, 1st edition, 2006.

[5] Eric O. Hall. The deformation and ageing of mild steel: III Discussion of results. *Proceedings of the Physical Society. Section B*, 64(9):747–753, 1951.

[6] Norman J. Petch. The cleavage strength of polycrystals. *J. Iron Steel Inst.*, 174(1):25–28, 1953.

[7] Yan Chong, Reza Gholizadeh, Tomohito Tsuru, et al. Grain refinement in titanium prevents low temperature oxygen embrittlement. *Nature Communications*, 14(1):404, 2023.

[8] Robert W. Balluffi. Grain boundary diffusion mechanisms in metals. *Metallurgical Transactions B*, 13(4):527–553, 1982.

[9] Ahmed A. Tiamiyu, Edward L. Pang, Xi Chen, James M. LeBeau, Keith A. Nelson, and Christopher A. Schuh. Nanotwinning-assisted dynamic recrystallization at high strains and strain rates. *Nature Materials*, 21(7):786–794, 2022.

[10] National Research Council. Integrated Computational Materials Engineering: A transformational discipline for improved competitiveness and national security. Technical report, National Academies Press, Washington, D.C., 2008.

[11] David S. Sholl and Janice A. Steckel. *Density Functional Theory: A Practical Introduction*. Wiley, Hoboken, NJ, 1st edition, 2009.

[12] Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications*. Academic Press, London, 2nd edition, 2001.

[13] Stefano Curtarolo, Gus L. W. Hart, Marco Buongiorno Nardelli, Natalio Mingo, Stefano Sanvito, and Ohad Levy. The high-throughput highway to computational materials design. *Nature Materials*, 12(3):191–201, 2013.

[14] Axel van de Walle and Mark Asta. High-throughput calculations in the context of alloy design. *MRS Bulletin*, 44(4):252–256, 2019.

[15] National Science and Technology Council. Materials Genome Initiative for global competitiveness. Technical report, Executive Office of the President, National Science and Technology Council, Washington, D.C., 2011. Accessed December 2021.

[16] Juan J. de Pablo, Barbara Jones, Cora Lind Kovacs, Vidvuds Ozolins, and Arthur P. Ramirez. The Materials Genome Initiative, the interplay of experiment, theory and computation. *Current Opinion in Solid State and Materials Science*, 18(2):99–117, 2014.

[17] Keith T. Butler, Daniel W. Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555, 2018.

[18] Taylor D. Sparks, Steven K. Kauwe, Marcus E. Parry, Aria Mansouri Tehrani, and Jakoah Brgoch. Machine learning for structural materials. *Annual Review of Materials Research*, 50(1):27–48, 2020.

[19] Gus L. W. Hart, Tim Mueller, Cormac Toher, and Stefano Curtarolo. Machine learning for alloys. *Nature Reviews Materials*, 6(8):730–755, 2021.

[20] Yuri Mishin, Mark Asta, and Ju Li. Atomistic modeling of interfaces and their impact on microstructure and properties. *Acta Materialia*, 58(4):1117–1151, 2010.

[21] Katsuyo Thornton, Samanthule Nola, R. Edwin Garcia, Mark Asta, and Gregory B. Olson. Computational materials science and engineering education: A survey of trends and needs. *JOM*, 61(10):12, 2009.

[22] Raúl A. Enrique, Mark Asta, and Katsuyo Thornton. Computational materials science and engineering education: An updated survey of trends and needs. *JOM*, 70(9):1644–1651, 2018.

[23] Rachael Mansbach, Andrew Ferguson, Kristopher Kilian, Jessica Krogstad, Cecilia Leal, Andre Schleife, Dallas R. Trinkle, Matthew West, and Geoffrey Herman. Reforming an undergraduate materials science curriculum with computational modules. *Journal of Materials Education*, 38(3-4):161–174, 2016.

[24] Timothy Chambers, Katsuyo Thornton, and Wenhao Sun. Teaching computational methods for materials discovery and design. *JOM*, 75(7):2086–2088, 2023.

[25] The Minerals Metals & Materials Society (TMS). Creating the next-generation materials genome initiative workforce. Technical report, TMS, Pittsburgh, PA, 2019.

[26] David L. McDowell. Gaps and barriers to successful integration and adoption of practical materials informatics tools and workflows. *JOM*, 73(1):138–148, 2021.

[27] Richard M. Martin. *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press, Cambridge, 2nd edition, 2020.

[28] Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical Review*, 136(3B):B864–B871, 1964.

[29] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133–A1138, 1965.

[30] John P. Perdew and Karla Schmidt. Jacob's ladder of density functional approximations for the exchange-correlation energy. *AIP Conference Proceedings*, 577(1):1–20, 2001.

[31] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical Review Letters*, 77(18):3865–3868, 1996.

[32] Peter E. Blöchl. Projector augmented-wave method. *Physical Review B*, 50(24):17953–17979, 1994.

[33] Georg Kresse and Daniel Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. *Physical Review B*, 59(3):1758–1775, 1999.

[34] Georg Kresse and Jürgen Hafner. *Ab initio* molecular dynamics for liquid metals. *Physical Review B*, 47(1):558–561, 1993.

[35] Georg Kresse and Jürgen Hafner. *Ab initio* molecular-dynamics simulation of the liquid-metal–amorphous-semiconductor transition in germanium. *Physical Review B*, 49(20):14251–14269, 1994.

[36] Georg Kresse and Jürgen Furthmüller. Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set. *Physical Review B*, 54(16):11169–11186, 1996.

[37] Georg Kresse and Jürgen Furthmüller. Efficiency of *ab initio* total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science*, 6(1):15–50, 1996.

[38] Luis A. Zepeda-Ruiz, Alexander Stukowski, Tomas Oppelstrup, Nicolas Bertin, Nathan R. Barton, Rodrigo Freitas, and Vasily V. Bulatov. Atomistic insights into metal hardening. *Nature Materials*, 20(3):315–320, 2021.

[39] Timofey Frolov, David L. Olmsted, Mark Asta, and Yuri Mishin. Structural phase transformations in metallic grain boundaries. *Nature Communications*, 4(1):1–7, 2013.

[40] Koenraad G. F. Janssens, David Olmsted, Elizabeth A. Holm, Stephen M. Foiles, Steven J. Plimpton, and Peter M. Derlet. Computing the mobility of grain boundaries. *Nature Materials*, 5(2):124, 2006.

[41] John E. Jones and Sydney Chapman. On the determination of molecular fields.—I. From the variation of the viscosity of a gas with temperature. *Proceedings of the Royal Society A*, 106(738):441–462, 1924.

[42] Jörg Behler. Four generations of high-dimensional neural network potentials. *Chemical Reviews*, 121(16):10037–10072, 2021.

[43] Murray S. Daw and Michael I. Baskes. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Physical Review B*, 29(12):6443–6453, 1984.

[44] Michael I. Baskes. Modified embedded-atom potentials for cubic materials and impurities. *Physical Review B*, 46(5):2727–2742, 1992.

[45] Michael P. Allen and Dominic J. Tildesley. *Computer Simulation of Liquids*. Oxford University Press, New York, 2nd edition, 2017.

[46] Hong Ding, Bharat Medasani, Wei Chen, Kristin A. Persson, Maciej Haranczyk, and Mark Asta. PyDII: A Python framework for computing equilibrium intrinsic point defect concentrations and extrinsic solute site preferences in intermetallic compounds. *Computer Physics Communications*, 193:118–123, 2015.

[47] Christopher Woodward, Mark Asta, Georg Kresse, and Jürgen Hafner. Density of constitutional and thermal point defects in $L1_2$ Al_3Sc. *Physical Review B*, 63(9):094103, 2001.

[48] Didier de Fontaine. *Principles of Classical Thermodynamics*. World Scientific, 1st edition, 2019.

[49] Mark Asta, Stephen M. Foiles, and Andrew A. Quong. First-principles calculations of bulk and interfacial thermodynamic properties for fcc-based Al-Sc alloys. *Physical Review B*, 57(18):11265–11275, 1998.

[50] Python framework for Defects In Intermetallics. *GitHub* `https://github.com/pydii/pydii`, 2015.

[51] Alex Zunger, Su-Huai Wei, Luiz G. Ferreira, and James E. Bernard. Special quasirandom structures. *Physical Review Letters*, 65(3):353–356, 1990.

[52] Axel van de Walle, Mark Asta, and Gerbrand Ceder. The alloy theoretic automated toolkit: A user guide. *Calphad*, 26(4):539–553, 2002.

[53] Axel van de Walle, Pratyush Tiwary, Maarten de Jong, David L. Olmsted, Mark Asta, A. Dick, Dongwon Shin, Y. Wang, Long-Qing Chen, and Zi-Kui Liu. Efficient stochastic generation of special quasirandom structures. *Calphad*, 42:13–18, 2013.

[54] Timofey Frolov, Wahyu Setyawan, Richard J. Kurtz, Jaime Marian, Artem E. Oganov, Robert E. Rudd, and Qiang Zhu. Grain boundary phases in bcc metals. *Nanoscale*, 10(17):8253–8268, 2018.

[55] Sebastian von Alfthan, Kimmo Kaski, and Adrian P. Sutton. Molecular dynamics simulations of temperature-induced structural transitions at twist boundaries in silicon. *Physical Review B*, 76(24):245317, 2007.

[56] Arash D. Banadaki, Mark A. Tschopp, and Srikanth Patala. An efficient Monte Carlo algorithm for determining the minimum energy structures of metallic grain boundaries. *Computational Materials Science*, 155:466–475, 2018.

[57] Matthew Guziewski, Arash D. Banadaki, Srikanth Patala, and Shawn P. Coleman. Application of Monte Carlo techniques to grain boundary structure optimization in silicon and silicon-carbide. *Computational Materials Science*, 182:109771, 2020.

[58] Shin Kiyohara and Teruyasu Mizoguchi. Searching the stable segregation configuration at the grain boundary by a Monte Carlo tree search. *The Journal of Chemical Physics*, 148(24):241741, 2018.

[59] Shengfeng Yang, Naixie Zhou, Hui Zheng, Shyue Ping Ong, and Jian Luo. First-order interfacial transformations with a critical point: Breaking the symmetry at a symmetric tilt grain boundary. *Physical Review Letters*, 120(8):085702, 2018.

[60] Alvin L.-S. Chua, Nicole A. Benedek, Lin Chen, Mike W. Finnis, and Adrian P. Sutton. A genetic algorithm for predicting the structures of interfaces in multicomponent systems. *Nature Materials*, 9(5):418–422, 2010.

[61] Sebastián Echeverri Restrepo, Simón Tamayo Giraldo, and Barend J. Thijsse. A genetic algorithm for generating grain boundaries. *Modelling and Simulation in Materials Science and Engineering*, 21(5):055017, 2013.

[62] Qiang Zhu, Amit Samanta, Bingxi Li, Robert E. Rudd, and Timofey Frolov. Predicting phase behavior of grain boundaries with evolutionary search and machine learning. *Nature Communications*, 9(1):467, 2018.

[63] Chaoming Yang, Mingfei Zhang, and Liang Qi. Grain boundary structure search by using an evolutionary algorithm with effective mutation methods. *Computational Materials Science*, 184:109812, 2020.

[64] Yehui Zhang, Bing Wang, Yixin Ouyang, Yipeng Zhou, Qiang Li, and Jinlan Wang. Toward low-symmetry systems: An adaptive differential evolution algorithm for global structure searching. *The Journal of Physical Chemistry Letters*, 13(13):2986–2993, 2022.

[65] Artem R. Oganov and Colin W. Glass. Crystal structure prediction using *ab initio* evolutionary techniques: Principles and applications. *The Journal of Chemical Physics*, 124(24):244704, 2006.

[66] Andriy O. Lyakhov, Artem R. Oganov, Harold T. Stokes, and Qiang Zhu. New developments in evolutionary structure prediction algorithm USPEX. *Computer Physics Communications*, 184(4):1172–1182, 2013.

[67] Mohammad S. Hooshmand, Ruopeng Zhang, Yan Chong, Enze Chen, Timofey Frolov, David L. Olmsted, Andrew M. Minor, and Mark Asta. Twin-boundary structural phase transitions in elemental titanium. *arXiv*, 2103.06194, 2021.

[68] Yan Chong, Max Poschmann, Ruopeng Zhang, et al. Mechanistic basis of oxygen sensitivity in titanium. *Science Advances*, 6(43):eabc4060, 2020.

[69] Nikolai A. Zarkevich and Duane D. Johnson. Titanium α-ω phase transformation pathway and a predicted metastable structure. *Physical Review B*, 93(2):020104, 2016.

[70] Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, et al. The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter*, 29(27):273002, 2017.

[71] Yuri Mishin, Michael J. Mehl, Dimitrios A. Papaconstantopoulos, Arthur F. Voter, and Joel D. Kress. Structural stability and lattice defects in copper: *Ab initio*, tight-binding, and embedded-atom calculations. *Physical Review B*, 63(22):224106, 2001.

[72] Xiaowang Zhou, R. A. Johnson, and Haydn N. G. Wadley. Misfit-energy-increasing dislocations in vapor-deposited CoFe/NiFe multilayers. *Physical Review B*, 69(14):144113, 2004.

[73] Frank H. Stillinger and Thomas A. Weber. Computer simulation of local order in condensed phases of silicon. *Physical Review B*, 31(8):5262–5271, 1985.

[74] Timofey Frolov, Qiang Zhu, Tomas Oppelstrup, Jaime Marian, and Robert E. Rudd. Structures and transitions in bcc tungsten grain boundaries and their role in the absorption of point defects. *Acta Materialia*, 159:123–134, 2018.

[75] Sebastian von Alfthan, Peter D. Haynes, Kimmo Kaski, and Adrian P. Sutton. Are the structures of twist grain boundaries in silicon ordered at 0 K? *Physical Review Letters*, 96(5):055505, 2006.

[76] Alexander Stukowski. Structure identification methods for atomistic simulations of crystalline materials. *Modelling and Simulation in Materials Science and Engineering*, 20(4):045021, 2012.

[77] Aidan P. Thompson, H. Metin Aktulga, Richard Berger, et al. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, 271:108171, 2022.

[78] Bernard D. Cullity and Stuart R. Stock. *Elements of X-Ray Diffraction*. Pearson, 3rd edition, 2001.

[79] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, et al. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020.

[80] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, et al. Jupyter notebooks - a publishing format for reproducible computational workflows. In Fernando Loizides and Birgit Schmidt, editors, *Positioning and power in academic publishing: Players, agents and agendas*, pages 87–90, 2016.

[81] Charles J. Weiss. A Creative Commons textbook for teaching scientific computing to chemistry students with Python and Jupyter notebooks. *Journal of Chemical Education*, 98(2):489–494, 2021.

[82] Erik J. Menke. Series of Jupyter notebooks using Python for an analytical chemistry course. *Journal of Chemical Education*, 97(10):3899–3903, 2020.

[83] David O. De Haan, Julia A. Schafer, and Eleanor I. Gillette. Using a modular approach to introduce Python coding to support existing course learning outcomes in a lower division analytical chemistry course. *Journal of Chemical Education*, 98(10):3245–3250, 2021.

[84] Michelene T. H. Chi and Ruth Wylie. The ICAP framework: Linking cognitive engagement to active learning outcomes. *Educational Psychologist*, 49(4):219–243, 2014.

[85] Scott Freeman, Sarah Eddy, Miles McDonough, Michelle Smith, Nnadozie Okoroafor, Hannah Jordt, and Mary Wenderoth. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415, 2014.

[86] Elli J. Theobald, Mariah J. Hill, Elisa Tran, et al. Active learning narrows achievement gaps for underrepresented students in undergraduate science, technology, engineering, and math. *Proceedings of the National Academy of Sciences*, 117(12):6476–6483, 2020.

[87] Executable Books Project. Jupyter Book. Zenodo, 2020. Accessed Apr 2022.

[88] Executable Books Project. MyST - Markedly Structured Text, 2022.

[89] Ani Adhikari, John DeNero, and David Wagner. *Computational and Inferential Thinking: The Foundations of Data Science*. GitHub, 2nd edition, 2021.

[90] John Sweller. Cognitive load theory, learning difficulty, and instructional design. *Learning and Instruction*, 4(4):295–312, 1994.

[91] Project Jupyter. JupyterHub, 2021. Accessed Jan. 2023.

[92] Google Research. Google Colaboratory, 2018. Accessed Jan. 2023.

[93] Carl E. Wieman, Wendy K. Adams, and Katherine K. Perkins. PhET: Simulations that enhance learning. *Science*, 322(5902):682–683, 2008.

[94] Paulo Blikstein and Uri Wilensky. An atom is known by the company it keeps: A constructionist learning environment for materials science using agent-based modeling. *International Journal of Computers for Mathematical Learning*, 14(2):81–119, 2009.

[95] Charles Xie, Robert Tinker, Barbara Tinker, Amy Pallant, Daniel Damelin, and Boris Berenfeld. Computational experiments for science education. *Science*, 332(6037):1516–1517, 2011.

[96] Enze Chen, Artur Tamm, Tao Wang, Mario E. Epler, Mark Asta, and Timofey Frolov. Modeling antiphase boundary energies of Ni_3Al-based alloys using automated density functional theory and machine learning. *npj Computational Materials*, 8(80), 2022.

[97] Tresa M. Pollock and Sammy Tin. Nickel-based superalloys for advanced turbine engines: Chemistry, microstructure and properties. *Journal of Propulsion and Power*, 22(2):361–374, 2006.

[98] Haibo Long, Shengcheng Mao, Yinong Liu, Ze Zhang, and Xiaodong Han. Microstructural and compositional design of Ni-based single crystalline superalloys — A review. *Journal of Alloys and Compounds*, 743:203–220, 2018.

[99] Ram Darolia. Development of strong, oxidation and corrosion resistant nickel-based superalloys: Critical review of challenges, progress and prospects. *International Materials Reviews*, 64(6):355–380, 2019.

[100] Eckhard Nembach and Günter Neite. Precipitation hardening of superalloys by ordered γ'-particles. *Progress in Materials Science*, 29(3):177–319, 1985.

[101] Bernard H. Kear and Heinz G. F. Wilsdorf. Dislocation configurations in plastically deformed polycrystalline Cu_3Au alloys. *Transactions of the Metallurgical Society of AIME*, 224:382–386, 1962.

[102] Bernd Reppich. Some new aspects concerning particle hardening mechanisms in γ' precipitating Ni-base alloys—I. Theoretical concept. *Acta Metallurgica*, 30(1):87–94, 1982.

[103] Vaclav Paidar, David P. Pope, and Vaclav Vitek. A theory of the anomalous yield behavior in $L1_2$ ordered alloys. *Acta Metallurgica*, 32(3):435–448, 1984.

[104] Timothy M. Smith et al. Phase transformation strengthening of high-temperature superalloys. *Nature Communications*, 7:13434, 2016.

[105] Xiaoxiang Wu, Surendra Kumar Makineni, Christian H. Liebscher, et al. Unveiling the Re effect in Ni-based single crystal superalloys. *Nature Communications*, 11(1):389, 2020.

[106] T. Yang, Y. L. Zhao, W. P. Li, et al. Ultrahigh-strength and ductile superlattice alloys with nanoscale disordered interfaces. *Science*, 369(6502):427–432, 2020.

[107] Nadine Baluc, Robin Schäublin, and Kevin J. Hemker. Methods for determining precise values of antiphase boundary energies in Ni_3Al. *Philosophical Magazine Letters*, 64(5):327–334, 1991.

[108] Dietmar Baither, Christian Rentenberger, Hans-Peter Karnthaler, and Eckhard Nembach. Three alternative experimental methods to determine the antiphase-boundary energies of the γ' precipitates in superalloys. *Philosophical Magazine A*, 82(9):1795–1805, 2002.

[109] Venkateswara Rao Manga, James E. Saal, Yi Wang, Vincent H. Crespi, and Zi-Kui Liu. Magnetic perturbation and associated energies of the antiphase boundaries in ordered Ni_3Al. *Journal of Applied Physics*, 108(10):103509, 2010.

[110] Mahesh Chandran and Sanjay K. Sondhi. First-principle calculation of APB energy in Ni-based binary and ternary alloys. *Modelling and Simulation in Materials Science and Engineering*, 19(2):025008, 2011.

[111] Xiao-Xiang Yu and Chong-Yu Wang. Effect of alloying element on dislocation cross-slip in γ'-Ni_3Al: A first-principles study. *Philosophical Magazine*, 92(32):4028–4039, 2012.

[112] K. V. Vamsi and S. Karthikeyan. Effect of off-stoichiometry and ternary additions on planar fault energies in Ni_3Al. In *Proceedings of the Twelfth International Symposium on Superalloys*, pages 521–530, Champion, PA, 2012. TMS.

[113] David J. Crudden, Alessandro Mottura, Nils Warnken, Babak Raeisinia, and Roger C. Reed. Modelling of the influence of alloy composition on flow stress in high-strength nickel-based superalloys. *Acta Materialia*, 75:356–370, 2014.

[114] Kaushlendra Kumar, Ramachandran Sankarasubramanian, and Umesh V. Waghmare. Tuning planar fault energies of Ni_3Al with substitutional alloying: First-principles description for guiding rational alloy design. *Scripta Materialia*, 142:74–78, 2018.

[115] Marcel Sluiter, Y. Hashi, and Yoshiyuki Kawazoe. The effect of segregation and partial order on the thermodynamics of (111) antiphase boundaries in Ni_3Al. *Computational Materials Science*, 14(1):283–290, 1999.

[116] Hai-Ping Wang, Marcel Sluiter, and Yoshiyuki Kawazoe. Prediction of the effect of Ti on the (111) and (100) antiphase boundary energy in Ni_3Al. *Materials Transactions JIM*, 40(11):1301–1305, 1999.

[117] Jian-Bo Liu and Duane D. Johnson. First principle predictions of anomalous yield strength in $L1_2$ materials. *Materials Research Innovations*, 18(sup4):S4–1021–S4–1025, 2014.

[118] Oleg I. Gorbatov, Ilya L. Lomaev, Yuri N. Gornostyrev, Andrei V. Ruban, David Furrer, Vasisht Venkatesh, Dmitri L. Novikov, and Sergei F. Burlatsky. Effect of composition on antiphase boundary energy in Ni_3Al-based alloys: *Ab initio* calculations. *Physical Review B*, 93(22):224106, 2016.

[119] Ruoshi Sun, Christopher Woodward, and Axel van de Walle. First-principles study on Ni_3Al antiphase boundary with Ti and Hf impurities. *Physical Review B*, 95(21):214121, 2017.

[120] Mohammad S. Dodaran, A. Hemmasian Ettefagh, Shengmin Guo, Michael M. Khonsari, Wen Jin Meng, Nima Shamsaei, and Shuai Shao. Effect of alloying elements on the γ' antiphase boundary energy in Ni-base superalloys. *Intermetallics*, 117:106670, 2020.

[121] Yuri Mishin. Atomistic modeling of the γ and γ'-phases of the Ni-Al system. *Acta Materialia*, 52(6):1451–1467, 2004.

[122] Chandler A. Becker, Yuri Mishin, and William J. Boettinger. The pre-wetting transition at antiphase boundaries: An atomistic modeling study of Ni_3Al. *Journal of Materials Science*, 43(11):3873–3880, 2008.

[123] A. P. Miodownik and N. Saunders. The calculation of APB energies in $L1_2$ compounds using a thermodynamic database. In P. Nash and B. Sundman, editors, *Applications of Thermodynamics in the Synthesis and Processing of Materials*, pages 91–104, Warrendale, PA, 1995. The Minerals, Metals & Materials Society.

[124] Xiwen Jia, Allyson Lynch, Yuheng Huang, et al. Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis. *Nature*, 573(7773):251–255, 2019.

[125] Hendrik J. Monkhorst and James D. Pack. Special points for Brillouin-zone integrations. *Physical Review B*, 13(12):5188–5192, 1976.

[126] Michael Methfessel and Anthony T. Paxton. High-precision sampling for Brillouin-zone integration in metals. *Physical Review B*, 40(6):3616–3621, 1989.

[127] Dennis M. Dimiduk, Anthony W. Thompson, and James C. Williams. The compositional dependence of antiphase-boundary energies and the mechanism of anomalous flow in Ni_3Al alloys. *Philosophical Magazine A*, 67(3):675–698, 1993.

[128] Tomas Kruml, Egle Conforto, Birgit Lo Piccolo, Daniel Caillard, and Jean-Luc Martin. From dislocation cores to strength and work-hardening: A study of binary Ni_3Al. *Acta Materialia*, 50(20):5091–5101, 2002.

[129] Enze Chen, Artur Tamm, Tao Wang, Mario E. Epler, Mark Asta, and Timofey Frolov. Antiphase boundary energies of Ni_3Al-based alloys. *MPContribs* `https://contribs.materialsproject.org/projects/apbe_Ni3Al`, 2022.

[130] Logan Ward, Alexander Dunn, Alireza Faghaninia, et al. Matminer: An open source toolkit for materials data mining. *Computational Materials Science*, 152:60–69, 2018.

[131] Julia Ling, Erin Antono, Saurabh Bajaj, Sean Paradiso, Maxwell Hutchinson, Bryce Meredig, and Brenna M. Gibbons. Machine learning for alloy composition and process optimization. In *ASME Turbo Expo 2018*, 2018.

[132] Logan Ward, Ankit Agrawal, Alok Choudhary, and Christopher Wolverton. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Computational Materials*, 2:16028, 2016.

[133] David G. Pettifor. A chemical scale for crystal-structure maps. *Solid State Communications*, 51(1):31–34, 1984.

[134] Charles Spearman. The proof and measurement of association between two things. *American Journal of Psychology*, 15(1):72–101, 1904.

[135] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[136] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[137] Hai-Jin Lu, Nan Zou, Ryan Jacobs, Ben Afflerbach, Xiao-Gang Lu, and Dane Morgan. Error assessment and optimal cross-validation approaches in machine learning applied to impurity diffusion. *Computational Materials Science*, 169:109075, 2019.

[138] Rees D. Rawlings and Anne E. Staton-Bevan. The alloying behaviour and mechanical properties of polycrystalline Ni_3Al (γ' phase) with ternary additions. *Journal of Materials Science*, 10(3):505–514, 1975.

[139] Navdeep Singh, Anjana Talapatra, Anchalee Junkaew, Thien Duong, Sean Gibbons, Shengyen Li, Hassan Thawabi, Emmi Olivos, and Raymundo Arróyave. Effect of ternary additions to structural properties of NiTi alloys. *Computational Materials Science*, 112:347–355, 2016.

[140] Rasim Eriş, M. Vedat Akdeniz, and Amdulla O. Mekhrabov. Atomic size effect of alloying elements on the formation, evolution and strengthening of γ'-Ni_3Al precipitates in Ni-based superalloys. *Intermetallics*, 109:37–47, 2019.

[141] Dobrina Iotova, Nicholas Kioussis, and Say Peng Lim. Electronic structure and elastic properties of the Ni_3X (X=Mn, Al, Ga, Si, Ge) intermetallics. *Physical Review B*, 54(20):14413–14422, 1996.

[142] Marcel H. F. Sluiter and Y. Kawazoe. Site preference of ternary additions in Ni_3Al. *Physical Review B*, 51(7):4062–4073, 1995.

[143] Andrei V. Ruban and Hans L. Skriver. Calculated site substitution in ternary γ'-Ni_3Al: Temperature and composition effects. *Physical Review B*, 55(2):856–874, 1997.

[144] Chao Jiang, Dan J. Sordelet, and Brian Gleeson. Site preference of ternary alloying elements in Ni_3Al: A first-principles study. *Acta Materialia*, 54(4):1147–1154, 2006.

[145] Qiong Wu and Shusuo Li. Alloying element additions to Ni_3Al: Site preferences and effects on elastic properties from first-principles calculations. *Computational Materials Science*, 53(1):436–443, 2012.

[146] Shaohua Liu, Minru Wen, Zi Li, Wenqing Liu, Ping Yan, and Chongyu Wang. Partitioning and diffusion of transition metal solutes in ternary model Ni-based single crystal superalloys. *Materials & Design*, 130:157–165, 2017.

[147] Baokun Lu, Chong-Yu Wang, and Zhihui Du. Site preferences of alloying transition metal elements in Ni-based superalloy: A first-principles study. *Chinese Physics B*, 27(9):097102, 2018.

[148] Aakash Kumar, Aleksandr Chernatynskiy, Minki Hong, Simon R. Phillpot, and Susan B. Sinnott. An *ab initio* investigation of the effect of alloying elements on the elastic properties and magnetic behavior of Ni_3Al. *Computational Materials Science*, 101:39–46, 2015.

[149] K. V. Vamsi and S. Karthikeyan. Modeling APB energies in multicomponent Ni-base superalloys. *Intermetallics*, 132:107124, 2021.

[150] A. Korner. Weak-beam study of superlattice dislocations moving on cube planes in Ni_3(Al, Ti) deformed at room temperature. *Philosophical Magazine A*, 58(3):507–522, 1988.

[151] Kevin J. Hemker and Michael J. Mills. Measurements of antiphase boundary and complex stacking fault energies in binary and B-doped Ni_3Al using TEM. *Philosophical Magazine A*, 68(2):305–324, 1993.

[152] Samer S. Ezz and Peter B. Hirsch. The strain rate sensitivity of the flow stress and the mechanism of deformation of single crystals of Ni_3(Al Hf)B. *Philosophical Magazine A*, 69(1):105–127, 1994.

[153] Jie Sun, Chun-Sing Lee, Joseph K. L. Lai, and Jian-Sheng Wu. Dislocation dissociations and fault energies in Ni_3Al alloys doped with palladium. *Intermetallics*, 7(12):1329–1335, 1999.

[154] Yoshiyuki Saito and Hiroshi Harada. The Monte Carlo simulation of ordering kinetics in Ni-base superalloys. *Materials Science and Engineering: A*, 223(1):1–9, 1997.

[155] Paul A. J. Bagot, Owen B. W. Silk, James O. Douglas, Stella Pedrazzini, David J. Crudden, Tomas L. Martin, Mark C. Hardy, Michael P. Moody, and Roger C. Reed. An atom probe tomography study of site preference and partitioning in a nickel-based superalloy. *Acta Materialia*, 125:156–165, 2017.

[156] Christopher Booth-Morrison, Zugang Mao, Ronald D. Noebe, and David N. Seidman. Chromium and tantalum site substitution patterns in $Ni_3Al(L1_2)$ γ'-precipitates. *Applied Physics Letters*, 93(3):033103, 2008.

[157] Akane Suzuki, Haruyuki Inui, and Tresa M. Pollock. $L1_2$-strengthened cobalt-base superalloys. *Annual Review of Materials Research*, 45(1):345–368, 2015.

[158] Z. M. Li, Xiaona Li, Y. L. Hu, et al. Cuboidal γ' phase coherent precipitation-strengthened Cu–Ni–Al alloys with high softening temperature. *Acta Materialia*, 203:116458, 2021.

[159] Roger C. Reed, T. Tao, and Nils Warnken. Alloys-By-Design: Application to nickel-based single crystal superalloys. *Acta Materialia*, 57(19):5898–5913, 2009.

[160] Edern Menou, Gérard Ramstein, Emmanuel Bertrand, and Franck Tancret. Multi-objective constrained design of nickel-base superalloys using data mining- and thermodynamics-driven genetic algorithms. *Modelling and Simulation in Materials Science and Engineering*, 24(5):055001, 2016.

[161] Timofey Frolov and Yuri Mishin. Phases, phase equilibria, and phase rules in low-dimensional systems. *The Journal of Chemical Physics*, 143(4):044706, 2015.

[162] Patrick R. Cantwell, Ming Tang, Shen J. Dillon, Jian Luo, Gregory S. Rohrer, and Martin P. Harmer. Grain boundary complexions. *Acta Materialia*, 62:1–48, 2014.

[163] Ming Tang, W. Craig Carter, and Rowland M. Cannon. Grain boundary transitions in binary alloys. *Physical Review Letters*, 97(7):075502, 2006.

[164] Patrick R. Cantwell, Timofey Frolov, Timothy J. Rupert, Amanda R. Krause, Christopher J. Marvel, Gregory S. Rohrer, Jeffrey M. Rickman, and Martin P. Harmer. Grain boundary complexion transitions. *Annual Review of Materials Research*, 50(1):465–492, 2020.

[165] Timfoey Frolov, Sergiy V. Divinski, Mark Asta, and Yuri Mishin. Effect of interface phase transformations on diffusion and segregation in high-angle grain boundaries. *Physical Review Letters*, 110(25):255502, 2013.

[166] Jiake Wei, Bin Feng, Ryo Ishikawa, Tatsuya Yokoi, Katsuyuki Matsunaga, Naoya Shibata, and Yuichi Ikuhara. Direct imaging of atomistic grain boundary migration. *Nature Materials*, 20(7):951–955, 2021.

[167] Frederic Sansoz and Jean-François Molinari. Mechanical behavior of Σ tilt grain boundaries in nanoscale Cu and Al: A quasicontinuum study. *Acta Materialia*, 53(7):1931–1944, 2005.

[168] Thorsten Meiners, Timofey Frolov, Robert E. Rudd, Gerhard Dehm, and Christian H. Liebscher. Observations of grain-boundary phase transformations in an elemental metal. *Nature*, 579(7799):375–378, 2020.

[169] Sai Rajeshwari K., Shanmugam Sankaran, K. C. Hari Kumar, Harald Rösner, Martin Peterlechner, Vladimir A. Esin, Sergiy Divinski, and Gerhard Wilde. Grain boundary diffusion and grain boundary structures of a Ni–Cr–Fe-alloy: Evidences for grain boundary phase transitions. *Acta Materialia*, 195:501–518, 2020.

[170] Timofey Frolov, Mark Asta, and Yuri Mishin. Phase transformations at interfaces: Observations from atomistic modeling. *Current Opinion in Solid State and Materials Science*, 20(5):308–315, 2016.

[171] Tobias Brink, Lena Langenohl, Hanna Bishara, and Gerhard Dehm. Universality of grain boundary phases in fcc metals: Case study on high-angle [111] symmetric tilt grain boundaries. *Physical Review B*, 107(5):054103, 2023.

[172] Jian Zhang, Cai-Zhuang Wang, and Kai-Ming Ho. Finding the low-energy structures of Si[001] symmetric tilted grain boundaries with a genetic algorithm. *Physical Review B*, 80(17):174102, 2009.

[173] Lin Sun, Miguel A. L. Marques, and Silvana Botti. Direct insight into the structure-property relation of interfaces from constrained crystal structure prediction. *Nature Communications*, 12(1):811, 2021.

[174] Arslan B. Mazitov and Artem R. Oganov. Grain boundaries in minerals: Atomic structure, phase transitions, and effect on strength of polycrystals. *Zapiski Rmo (proceedings of the Russian Mineralogical Society)*, 150(5):92–102, 2021.

[175] Yuri Mishin and Diana Farkas. Atomistic simulation of [001] symmetrical tilt grain boundaries in NiAl. *Philosophical Magazine A*, 78(1):29–56, 1998.

[176] Jian Han, Vaclav Vitek, and David J. Srolovitz. Grain-boundary metastability and its statistical properties. *Acta Materialia*, 104:259–273, 2016.

[177] Gerd Lütjering and James C. Williams. *Titanium*. Springer, Berlin, 2nd edition, 2007.

[178] Madeleine N. Kelly, Krzysztof Glowinski, Noel T. Nuhfer, and Gregory S. Rohrer. The five parameter grain boundary character distribution of α-Ti determined from three-dimensional orientation data. *Acta Materialia*, 111:22–30, 2016.

[179] Yue Chao Wang and Heng Qiang Ye. On the tilt grain boundaries in hcp Ti with [0001] orientation. *Philosophical Magazine A*, 75(1):261–272, 1997.

[180] Thomas Hammerschmidt, Alfred Kersch, and Peter Vogl. Embedded atom simulations of titanium systems with grain boundaries. *Physical Review B*, 71(20):205409, 2005.

[181] Zebang Zheng, Daniel S. Balint, and Fionn P. E. Dunne. Investigation of slip transfer across HCP grain boundaries with application to cold dwell facet fatigue. *Acta Materialia*, 127:43–53, 2017.

[182] Diana Farkas. Grain-boundary structures in hexagonal materials: Coincident and near coincident grain boundaries. *Metallurgical and Materials Transactions A*, 25(7):1337–1346, 1994.

[183] Jian Wang and Irene J. Beyerlein. Atomic structures of $[0\bar{1}10]$ symmetric tilt grain boundaries in hexagonal close-packed (hcp) crystals. *Metallurgical and Materials Transactions A*, 43(10):3556–3569, 2012.

[184] Chang Ni, Hong Ding, Mark Asta, and Xuejun Jin. Computational study of $\langle 1\bar{1}00\rangle$ symmetric tilt grain boundaries in Mg and Ti. *Scripta Materialia*, 109:94–99, 2015.

[185] Mehul A. Bhatia and Kiran N. Solanki. Energetics of vacancy segregation to symmetric tilt grain boundaries in hexagonal closed pack materials. *Journal of Applied Physics*, 114(24):244309, 2013.

[186] Jian Wang and Irene J. Beyerlein. Atomic structures of symmetric tilt grain boundaries in hexagonal close packed (hcp) crystals. *Modelling and Simulation in Materials Science and Engineering*, 20(2):024002, 2012.

[187] Shyue Ping Ong, William Davidson Richards, Anubhav Jain, et al. Python Materials Genomics (pymatgen): A robust, open-source Python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

[188] Rajendra R. Zope and Yuri Mishin. Interatomic potentials for atomistic simulations of the Ti–Al system. *Physical Review B*, 68(2):024102, 2003.

[189] Richard G. Hennig, Thomas J. Lenosky, Dallas R. Trinkle, Sven P. Rudin, and John W. Wilkins. Classical potential describes martensitic phase transformations between the α, β, and ω titanium phases. *Physical Review B*, 78(5):054121, 2008.

[190] Alexander Stukowski. Visualization and analysis of atomistic simulation data with OVITO—the Open Visualization Tool. *Modelling and Simulation in Materials Science and Engineering*, 18(1):015012, 2009.

[191] Alexander Stukowski, Vasily V. Bulatov, and Athanasios Arsenlis. Automated identification and indexing of dislocations in crystal interfaces. *Modelling and Simulation in Materials Science and Engineering*, 20(8):085007, 2012.

[192] Garritt J. Tucker and David L. McDowell. Non-equilibrium grain boundary structure and inelastic deformation using atomistic simulations. *International Journal of Plasticity*, 27(6):841–857, 2011.

[193] P. G. Partridge. The crystallography and deformation modes of hexagonal close-packed metals. *Metallurgical Reviews*, 12(1):169–194, 1967.

[194] John Wyrill Christian and Subhash Mahajan. Deformation twinning. *Progress in Materials Science*, 39(1):1–157, 1995.

[195] William T. Read and William Shockley. Dislocation models of crystal grain boundaries. *Physical Review*, 78(3):275–289, 1950.

[196] David L. Olmsted, Dorel Buta, Ari Adland, Stephen M. Foiles, Mark Asta, and Alain Karma. Dislocation-pairing transitions in hot grain boundaries. *Physical Review Letters*, 106(4):046101, 2011.

[197] Ian S. Winter, Robert E. Rudd, Tomas Oppelstrup, and Timofey Frolov. Nucleation of grain boundary phases. *Physical Review Letters*, 128(3):035701, 2022.

[198] Yigal Komem, Pierre Pétroff, and Robert W. Balluffi. Direct observation of grain boundary dislocation climb in ion-irradiated gold bicrystals. *The Philosophical Magazine: A Journal of Theoretical Experimental and Applied Physics*, 26(1):239–252, 1972.

[199] Zhongxia Shang, C. Fan, S. Xue, Jie Ding, Jin Li, Thomas Voisin, Yinmin Morris Wang, H. Wang, and Xinghang Zhang. Response of solidification cellular structures in additively manufactured 316 stainless steel to heavy ion irradiation: an *in situ* study. *Materials Research Letters*, 7(7):290–297, 2019.

[200] Chunguang Yan, Rongshan Wang, Yanli Wang, Xitao Wang, and Guanghai Bai. Effects of ion irradiation on microstructure and properties of zirconium alloys—A review. *Nuclear Engineering and Technology*, 47(3):323–331, 2015.

[201] Edmanuel Torres. Atomistic study of the structure and deformation behavior of symmetric tilt grain boundaries in α-zirconium. *Computational Materials Science*, 197:110600, 2021.

[202] Rodrigo Freitas and Yifan Cao. Machine-learning potentials for crystal defects. *MRS Communications*, 12(5):510–520, 2022.

[203] Teng Long, Nuno M. Fortunato, Ingo Opahle, Yixuan Zhang, Ilias Samathrakis, Chen Shen, Oliver Gutfleisch, and Hongbin Zhang. Constrained crystals deep convolutional generative adversarial network for the inverse design of crystal structures. *npj Computational Materials*, 7(1):1–7, 2021.

[204] Enze Chen and Mark Asta. Using Jupyter tools to design an interactive textbook to guide undergraduate research in materials informatics. *Journal of Chemical Education*, 99(10):3601–3606, 2022.

[205] Enze Chen, Mark Asta, and Andrew M. Minor. Integrating programming-based modules into a materials characterization laboratory course to reinforce data science and scientific writing. In *2023 ASEE Annual Conference & Exposition*, 2023.

[206] Adam C. Mater and Michelle L. Coote. Deep learning in chemistry. *Journal of Chemical Information and Modeling*, 59(6):2545–2559, 2019.

[207] Connor W. Coley, Dale A. Thomas, Justin A. M. Lummiss, et al. A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science*, 365(6453):eaax1566, 2019.

[208] Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018.

[209] Hongming Chen, Ola Engkvist, Yinhai Wang, Marcus Olivecrona, and Thomas Blaschke. The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6):1241–1250, 2018.

[210] Sunghwan Kim, Ehren C. Bucholtz, Kristin Briney, et al. Teaching cheminformatics through a collaborative intercollegiate online chemistry course (OLCC). *Journal of Chemical Education*, 98(2):416–425, 2021.

[211] Johannes Pernaa. Possibilities and challenges of using educational cheminformatics for STEM education: A SWOT analysis of a molecular visualization engineering project. *Journal of Chemical Education*, 99(3):1190–1200, 2022.

[212] Kaitlin Tyler, Enze Chen, Bryce Meredig, and Taylor Sparks. Artificial intelligence in materials education: A roundtable discussion. *JOM*, 75(7):2083–2085, 2023.

[213] Carl Wieman, Argenta Price, and Candice Kim. Instructional design based on the problem-solving decisions of scientists and engineers. *ETH Learning and Teaching Journal*, 2(2):10–17, 2020.

[214] Jeffrey M. Perkel. Programming: Pick up Python. *Nature*, 518(7537):125–126, 2015.

[215] Marie van Staveren. Integrating Python into a physical chemistry lab. *Journal of Chemical Education*, 99(7):2604–2609, 2022.

[216] Charles J. Weiss. Scientific computing for chemists: An undergraduate course in simulations, data processing, and visualization. *Journal of Chemical Education*, 94(5):592–597, 2017.

[217] Deborah Lafuente, Brenda Cohen, Guillermo Fiorini, Agustín Alejo García, Mauro Bringas, Ezequiel Morzan, and Diego Onna. A gentle introduction to machine learning for chemists: An undergraduate workshop using Python notebooks for visualization, data processing, analysis, and modeling. *Journal of Chemical Education*, 98(9):2892–2898, 2021.

[218] Alejandro Strachan and Saaketh Desai. Hands-on Data Science and Machine Learning Training Series. nanoHUB, 2020. Accessed December 2021.

[219] National Institute of Standards and Technology. Machine Learning for Materials Research Bootcamp & Workshop on Machine Learning Microscopy Data, 2021. Accessed May 2022.

[220] Enze Chen, Oghoghosa Igbineweka, Lenore Kubie, and James S. Peerless. Educating current industrial workforce to embrace data-driven materials development. *MRS Bulletin*, 47(10):981–985, 2022.

[221] Randall Munroe. Python environment. `https://xkcd.com/1987/`, 2018. Accessed May 2022.

[222] Felipe Engelberger, Pablo Galaz-Davison, Graciela Bravo, Maira Rivera, and César A. Ramírez-Sarmiento. Developing and implementing cloud-based tutorials that combine bioinformatics software, interactive coding, and visualization exercises for distance learning on structural bioinformatics. *Journal of Chemical Education*, 98(5):1801–1807, 2021.

[223] Project Jupyter, Matthias Bussonnier, Jessica Forde, et al. Binder 2.0 - reproducible, interactive, sharable environments for science at scale. *Proceedings of the 17th Python in Science Conference*, pages 113–120, 2018.

[224] Kevin Markham. Six easy ways to run your Jupyter Notebook in the cloud. Data School, 2019.

[225] Geoffrey R. Hutchison. Integrating Python into an undergraduate mathematics for chemists course. In *Teaching Programming across the Chemistry Curriculum*, volume 1387 of *ACS Symposium Series*, chapter 9, pages 123–134. American Chemical Society, 2021.

[226] Enze Chen and Mark Asta. *Introduction to Materials Informatics*. GitHub Pages, 2021.

[227] Andrew D. White. *Deep Learning for Molecules and Materials*. GitHub, 2021.

[228] Bradley N. Miller and David L. Ranum. Beyond PDF and ePub: Toward an interactive textbook. In *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education*, ITiCSE '12, pages 150–155, New York, NY, USA, 2012.

[229] David H. Smith, Qiang Hao, Christopher D. Hundhausen, Filip Jagodzinski, Josh Myers-Dean, and Kira Jaeger. Towards modeling student engagement with interactive computing textbooks: An empirical study. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, SIGCSE '21, pages 914–920, New York, NY, USA, 2021.

[230] Elaine Seymour, Anne-Barrie Hunter, Sandra L. Laursen, and Tracee DeAntoni. Establishing the benefits of research experiences for undergraduates in the sciences: First findings from a three-year study. *Science Education*, 88(4):493–534, 2004.

[231] Marcia C. Linn, Erin Palmer, Anne Baranger, Elizabeth Gerard, and Elisa Stone. Undergraduate research experiences: Impacts and opportunities. *Science*, 347(6222):1261757, 2015.

[232] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, et al. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013.

[233] Ioannis Petousis, David Mrdjenovich, Eric Ballouz, et al. High-throughput screening of inorganic compounds for the discovery of novel dielectric and optical materials. *Scientific Data*, 4(1):160134, 2017.

[234] Kazuki Morita, Daniel W. Davies, Keith T. Butler, and Aron Walsh. Modeling the dielectric constants of crystals using machine learning. *The Journal of Chemical Physics*, 153(2):024503, 2020.

[235] Abhijith Gopakumar, Koushik Pal, and Chris Wolverton. Identification of high-dielectric constant compounds from statistical design. *npj Computational Materials*, 8(146):1–10, 2022.

[236] Angus I. Kingon, Jon-Paul Maria, and S. K. Streiffer. Alternative dielectrics to silicon dioxide for memory and logic devices. *Nature*, 406(6799):1032–1038, 2000.

[237] John Robertson. High dielectric constant oxides. *The European Physical Journal Applied Physics*, 28(3):265–291, 2004.

[238] Division of Computing, Data Science, and Society. DataHub, 2020.

[239] Laurie Williams, Robert R. Kessler, Ward Cunningham, and Ron Jeffries. Strengthening the case for pair programming. *IEEE Software*, 17(4):19–25, 2000.

[240] Hypothesis team. hypothes.is, 2022.

[241] Project Jupyter, Douglas Blank, David Bourgin, et al. nbgrader: A tool for creating and grading assignments in the Jupyter notebook. *Journal of Open Source Education*, 2(16):32, 2019.

[242] Christopher Pyles and UC Berkeley Data Science Education Program. Otter-Grader: A Python and R autograding solution. Zenodo, 2022.

[243] Santiago Vargas, Siavash Zamirpour, Shreya Menon, et al. Team-based learning for scientific computing and automated experimentation: Visualization of colored reactions. *Journal of Chemical Education*, 97(3):689–694, 2020.

[244] Katy Börner, Olga Scrivner, Mike Gallant, Shutian Ma, Xiaozhong Liu, Keith Chewning, Lingfei Wu, and James A. Evans. Skill discrepancies between research, education, and jobs reveal the critical need to supply soft skills for the data economy. *Proceedings of the National Academy of Sciences*, 115(50):12630–12637, 2018.

[245] National Academies of Sciences, Engineering, and Medicine. Communicating science effectively: A research agenda. Technical report, National Academies Press, Washington, D.C., 2017.

[246] National Academies of Sciences, Engineering, and Medicine. Data science for undergraduates: Opportunities and options. Technical report, National Academies Press, Washington, D.C., 2018.

[247] ABET. Criteria for accrediting engineering programs, 2022–2023, 2022. Accessed Jan. 2023.

[248] Eric Wiebe, Thomas Hare, Michael Carter, Yusef Fahmy, Roger Russell, and Miriam Ferzli. Supporting lab report writing in an introductory materials engineering lab. In *2001 ASEE Annual Conference*, pages 6.917.1–6.917.11, 2001.

[249] Amber Genau. Teaching report writing in undergraduate labs. In *2020 ASEE Annual Conference*, 2020.

[250] Benjamin Afflerbach, Nafsaniath Fathema, Anne Gillian-Daniel, Wendy Crone, and Dane Morgan. Authentic undergraduate research in machine learning with the Informatics Skunkworks: A strategy for scalable apprenticeship applied to materials informatics research. In *2022 ASEE Annual Conference & Exposition*, 2022.

[251] Enze Chen, Mark Asta, and Andrew M. Minor. *MSE 104L Data Analysis*. GitHub Pages, 2023.

[252] Chad M. Parish and Philip D. Edmondson. Data visualization heuristics for the physical sciences. *Materials & Design*, 179:107868, 2019.

[253] Gregg M. Janowski. Preparing students for communicating in the professional world. *JOM*, 74(8):2883–2884, 2022.

[254] John D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science Engineering*, 9(3):90–95, 2007.

[255] J. B. Nelson and Dennis P. Riley. An experimental investigation of extrapolation methods in the derivation of accurate unit-cell dimensions of crystals. *Proceedings of the Physical Society*, 57(3):160–177, 1945.

[256] Wes McKinney. Data structures for statistical computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56–61, 2010.

[257] Susan P. Gentry. Motivation on programming assignments in materials science and engineering. In *2019 ASEE Annual Conference & Exposition*, page 16, 2019.

[258] Grace M. Lu, Dallas R. Trinkle, Andre Schleife, et al. Impact of integrating computation into undergraduate curriculum: New modules and long-term trends. In *2020 ASEE Annual Conference*, 2020.

[259] Nathan C. Emery, Erika Crispo, Sarah R. Supp, Kaitlin J. Farrell, Andrew J. Kerkhoff, Ellen K. Bledsoe, Kelly L. O'Donnell, Andrew C. McCall, and Matthew E. Aiello-Lammens. Data science in undergraduate life science education: A need for instructor skills training. *BioScience*, 71(12):1274–1287, 2021.

[260] L. Catherine Brinson, Laura M. Bartolo, Benjamin Blaiszik, David Elbert, Ian Foster, Alejandro Strachan, and Peter W. Voorhees. Community action on FAIR data will fuel a revolution in materials research. *MRS Bulletin*, 2023.

[261] David L. Olmsted, Stephen M. Foiles, and Elizabeth A. Holm. Survey of computed grain boundary properties in face-centered cubic metals: I. Grain boundary energy. *Acta Materialia*, 57(13):3694–3703, 2009.

[262] Mark A. Tschopp, Shawn P. Coleman, and David L. McDowell. Symmetric and asymmetric tilt grain boundary structure and energy in Cu and Al (and transferability to other fcc metals). *Integrating Materials and Manufacturing Innovation*, 4(1):176–189, 2015.

[263] Eric R. Homer, Gus L. W. Hart, C. Braxton Owens, Derek M. Hensley, Jay C. Spendlove, and Lydia H. Serafin. Examination of computed aluminum grain boundary structures and energies that span the 5D space of crystallographic character. *Acta Materialia*, 234:118006, 2022.

[264] Hyun-Kyu Kim, Won-Seok Ko, Hyuk-Joong Lee, Seong Gyoon Kim, and Byeong-Joo Lee. An identification scheme of grain boundaries and construction of a grain boundary energy database. *Scripta Materialia*, 64(12):1152–1155, 2011.

[265] Sutatch Ratanaphan, David L. Olmsted, Vasily V. Bulatov, Elizabeth A. Holm, Anthony D. Rollett, and Gregory S. Rohrer. Grain boundary energies in body-centered cubic metals. *Acta Materialia*, 88:346–354, 2015.

[266] Hui Zheng, Xiang-Guo Li, Richard Tran, Chi Chen, Matthew Horton, Donald Winston, Kristin Aslaug Persson, and Shyue Ping Ong. Grain boundary properties of elemental metals. *Acta Materialia*, 186:40–49, 2020.

[267] Brendon Waters, Daniel S. Karls, Ilia Nikiforov, Ryan S. Elliott, Ellad B. Tadmor, and Brandon Runnels. Automated determination of grain boundary energy and potential-dependence using the OpenKIM framework. *Computational Materials Science*, 220:112057, 2023.

[268] Patrick Huck, Anubhav Jain, Dan Gunter, Donald Winston, and Kristin Persson. A community contribution framework for sharing materials data with materials project. In *2015 IEEE 11th International Conference on e-Science*, pages 535–541, 2015.

[269] Sebastiaan P. Huber, Spyros Zoupanos, Martin Uhrin, et al. AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. *Scientific Data*, 7(1):300, 2020.

[270] Benjamin Blaiszik, Kyle Chard, Jim Pruyne, Rachana Ananthakrishnan, Steven Tuecke, and Ian Foster. The materials data facility: Data services to advance materials science research. *JOM*, 68(8):2045–2052, 2016.

[271] Gerhard Dehm and Julie Cairney. Implication of grain-boundary structure and chemistry on plasticity and failure. *MRS Bulletin*, 47(8):800–807, 2022.

[272] Malik Wagih, Peter M. Larsen, and Christopher A. Schuh. Learning grain boundary segregation energy spectra in polycrystals. *Nature Communications*, 11(1):6376, 2020.

[273] Jonathan L. Priedeman, Conrad W. Rosenbrock, Oliver K. Johnson, and Eric R. Homer. Quantifying and connecting atomic and crystallographic grain boundary structure using local environment representation and dimensionality reduction techniques. *Acta Materialia*, 161:431–443, 2018.

[274] CAST. Universal design for learning guidelines version 2.2. https://udlguidelines.cast.org/, 2018.

[275] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, et al. On the opportunities and risks of foundation models. *arXiv*, 2108.07258, 2022.

[276] Lawrence K. Fung, Tashiema L. Ulrich, Kiyo T. Fujimoto, and Mitra Taheri. Neurodiversity: An invisible strength? *JOM*, 74(9):3200–3202, 2022.

[277] World Health Organization. Disability fact sheet. `https://www.who.int/news-room/fact-sheets/detail/disability-and-health`, 2023.

[278] David H. Rose, Wendy S. Harbour, Catherine Sam Johnston, Samantha G. Daley, and Linda Abarbanell. Universal design for learning in postsecondary education: Reflections on principles and their application. *Journal of Postsecondary Education and Disability*, 19(2):135–151, 2006.

[279] Corbin Gruber, Alec James, Jeremy T. Berchtold, Zoe J. Wood, Gregory E. Scott, and Zahra Alghoul. Interactive unit cell visualization tool for crystal lattice structures. *Journal of Chemical Education*, 97(7):2020–2024, 2020.

[280] Susan P. Gentry, Tanya Faltens, Ashwin Wheeler, and Andre Schleife. Measuring student learning of crystal structures using computer-based visualizations. In *2018 ASEE Annual Conference & Exposition*, page 27, 2018.

[281] Matthew West, Geoffrey L. Herman, and Craig Zilles. PrairieLearn: Mastery-based online problem solving with adaptive scoring and recommendations driven by machine learning. In *2015 ASEE Annual Conference & Exposition*, pages 26.1238.1–26.1238.14, 2015.

[282] Benjamin S. Bloom. Learning for mastery. *Evaluation Comment*, 1(2):1–12, 1968.

[283] Łukasz Mentel. mendeleev - A Python resource for properties of chemical elements, ions and isotopes. *GitHub* `https://github.com/lmmentel/mendeleev`, 2014.

[284] Andries R. Miedema. A simple model for alloys. I. rules for the alloying behaviour of transition metals. *Philips Technical Review*, 33:149–160, 1973.

[285] X. Yang and Yong Zhang. Prediction of high-entropy stabilized solid-solution in multi-component alloys. *Materials Chemistry and Physics*, 132(2):233–238, 2012.

[286] Ann M. Deml, Ryan O'Hayre, Chris Wolverton, and Vladan Stevanović. Predicting density functional theory total energies and enthalpies of formation of metal-nonmetal compounds by linear regression. *Physical Review B*, 93(8):085142, 2016.

Appendix A

APBs in Ni_3Al

A.1 ML model without ordering energy

While including the ordering energy (E_{ord}) as a feature for modeling the APB energy leverages our domain knowledge [113, 118], there may be interest in an ML model without E_{ord}, particularly for multicomponent compositions where this information may not be readily available (see Subsection 3.3.4 in the main text) or vary in estimated values depending on the database used. We use the same procedure outlined in the manuscript to train an ML model without E_{ord} and we obtain a 5-fold grouped CV error of $0.035\,\text{J}\,\text{m}^{-2}$ (compared to $0.033\,\text{J}\,\text{m}^{-2}$ with the feature included). Therefore, one might not *need* to calculate E_{ord} in order to predict the APB energies in the model systems used in this study, and we showed in Subsection 3.3.4 how a model trained without E_{ord} can extrapolate to multicomponent systems. However, we feel it is an important contribution to include, when available, for several reasons:

- E_{ord} is a physically-motivated correlation that has been proposed in the literature, which we discussed in Section 3.1. As discussed there, we hope to use ML and data-driven techniques to build on top of this domain knowledge.
- E_{ord} remains after recursive feature elimination, which is a data-driven approach to feature selection as opposed to physical intuition.
- When included, E_{ord} is the fourth-most important feature using the random forest model's feature importance ranking.
- The performance is still marginally better with E_{ord} included. In addition, the good performance achieved with the other, auto-generated features helps support one of our central conclusions that informatics approaches can outperform methods that rely on physical intuition or analytical models alone.

Feature	Method
Ordering energy (E_{ord})	DFT [34–37]
Change in lattice parameter (Δa) and density	DFT
Sublattice preference	PyDII [46, 50]
Pettifor number [133]	mendeleev package [283]
Magpie elemental properties [132]	matminer package [130]
Pymatgen elemental properties [187]	matminer package
Intermetallic formation enthalpies [284]	matminer package
Mixing thermochemistry and size mismatch [285]	matminer package
Fraction of magnetic transition metals [286]	matminer package

Table A.1: All of the features that we initially generate for our ML models and their corresponding citations.

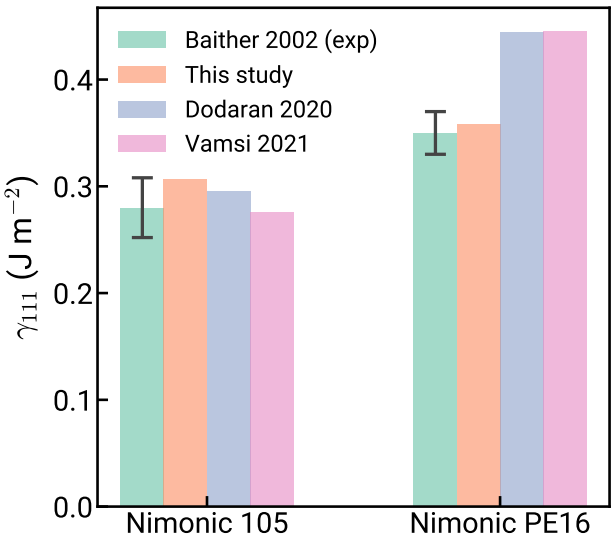


Figure A.1: We compare the performance of our ML model against other models in the literature in predicting the APB energy of two commercial superalloys, Nimonic 105 and Nimonic PE16. The green bars are experimental measurements from Baither et al. [108], who report values for γ_{111} between 0.252–0.308 J m^{-2} and 0.330–0.370 J m^{-2} for the two alloys, respectively. The orange, lavender, and pink bars correspond to the predicted APB energies from our ML model, the model by Dodaran et al. [120], and the model by Vamsi and Karthikeyan [149], respectively.

Solute	Site pref.	Low (1.39 at.%)	Medium (5.56 at.%)	High (9.72 at.%)
Ag	Ni	0.187(01)	0.193(15)	0.210(13)
Au	Ni	0.191(02)	0.205(06)	0.233(22)
Cd	mixed	0.192(07)	0.183(21)	0.218(27)
Ce	Al	0.227(14)	0.335(10)	0.374(31)
Co	Ni	0.189(02)	0.190(07)	0.184(13)
Cr	Al	0.217(09)	0.281(04)	0.282(05)
Cu	Ni	0.186(02)	0.197(13)	0.229(19)
Fe	mixed	0.203(07)	0.185(16)	0.198(33)
Ga	Al	0.185(02)	0.173(02)	0.163(03)
Ge	Al	0.189(03)	0.204(03)	0.223(05)
Hf	Al	0.205(29)	0.320(11)	0.390(28)
Hg	mixed	0.193(07)	0.190(20)	0.228(22)
Ir	mixed	0.177(05)	0.164(06)	0.158(24)
La	Al	0.223(10)	0.291(22)	0.262(80)
Mg	Al	0.186(04)	0.169(02)	0.148(07)
Mn	Al	0.187(07)	0.187(07)	0.202(21)
Mo	Al	0.224(14)	0.344(12)	0.252(33)
Nb	Al	0.221(13)	0.340(13)	0.428(29)
Os	Al	0.216(08)	0.206(15)	0.096(28)
Pb	Al	0.188(03)	0.199(05)	0.216(13)
Pd	Ni	0.185(04)	0.179(12)	0.166(10)
Pt	Ni	0.187(03)	0.189(13)	0.180(10)
Re	Al	0.227(15)	0.304(14)	0.122(38)
Rh	Ni	0.179(05)	0.146(17)	0.107(25)
Ru	Al	0.202(08)	0.184(09)	0.120(20)
Sc	Al	0.208(05)	0.269(06)	0.293(20)
Si	Al	0.189(04)	0.201(02)	0.219(07)
Sn	Al	0.189(04)	0.207(05)	0.232(12)
Ta	Al	0.225(23)	0.353(12)	0.464(30)
Tc	Al	0.221(10)	0.256(13)	0.142(27)
Ti	Al	0.211(08)	0.292(07)	0.344(16)
Tl	Al	0.181(04)	0.151(06)	0.069(08)
V	Al	0.215(09)	0.308(08)	0.382(14)
W	Al	0.225(16)	0.369(11)	0.295(41)
Y	Al	0.215(08)	0.247(07)	0.245(38)
Zn	mixed	0.197(07)	0.194(15)	0.247(33)
Zr	Al	0.219(11)	0.318(11)	0.374(27)

Table A.2: γ_{111} (in $\mathrm{J\,m^{-2}}$) of ternary Ni_3Al-based alloys. The APB energy is calculated for three different concentrations of the ternary species. The uncertainty in the final two decimal places is the standard deviation of five different DFT calculations of γ_{111} for each composition. Each row also contains that solute's PyDII-calculated sublattice preference.

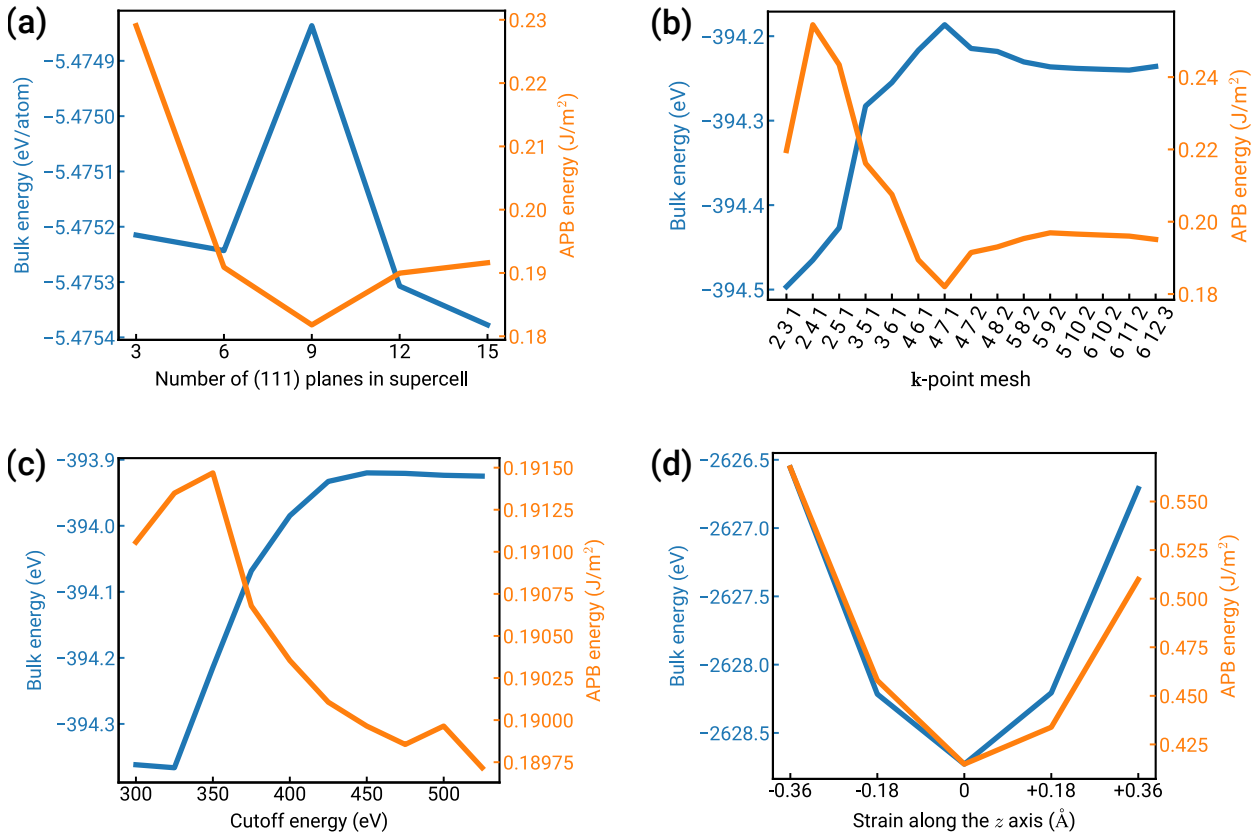


Figure A.2: VASP convergence tests. Convergence of the total energy in the bulk $L1_2$ structure of Ni_3Al in a $1 \times 1 \times 3$ supercell oriented with (111) planes normal to the z-axis. Convergence of γ_{111} with respect to (a) the number of (111) planes separating the APB in Ni_3Al, (b) the **k**-point mesh, and (c) the plane wave cutoff energy. (d) Convergence of the two quantities with respect to the cell height (measured as strain in Å following volume relaxation) in a 288-atom cell with 9.72 at.% Nb.

Appendix B

GBs in α-Ti

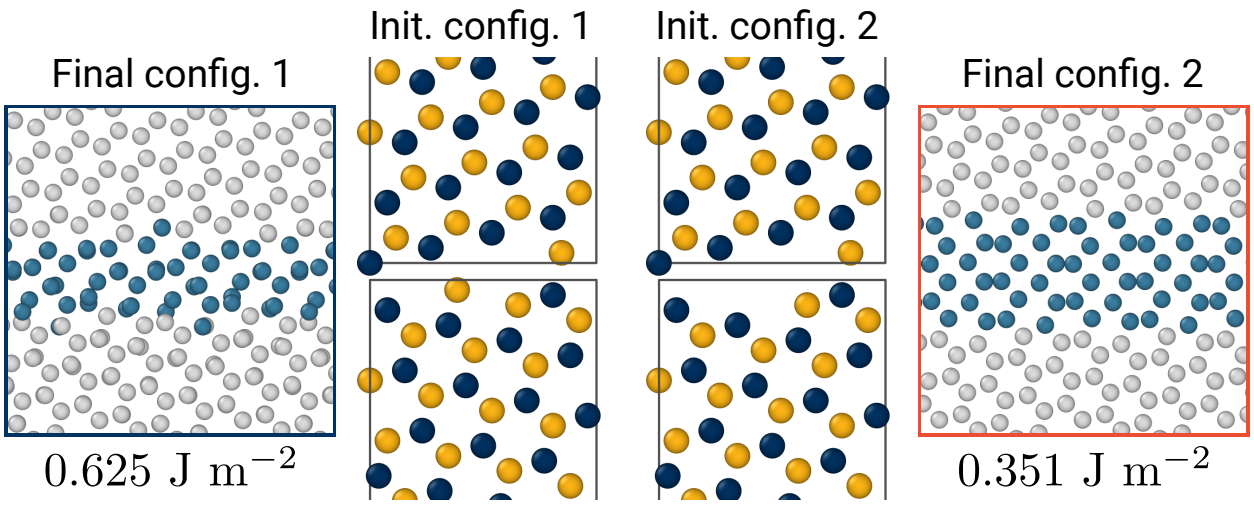


Figure B.1: Different planar terminations and corresponding GB structures. $\{31\bar{4}0\}[0001]$ is a Type I GB where the two basis atoms have different z coordinates normal to the GB plane, and two initial configurations are shown where the lower slabs terminate at different basis atoms (gold for the first, blue for the second). We use the γ-surface method to optimize the GB structure in both cases. The first configuration ($n = 0$) produces a higher-energy structure shown on the left ($E_{\mathrm{gb}} = 0.625\,\mathrm{J\,m^{-2}}$), while the second configuration ($n = 0.5$) produces the lower-energy structure shown on the right ($E_{\mathrm{gb}} = 0.351\,\mathrm{J\,m^{-2}}$), which matches the ground-state structure from GRIP. The final structures and colors correspond to those in Figure 4.1 in Chapter 4.

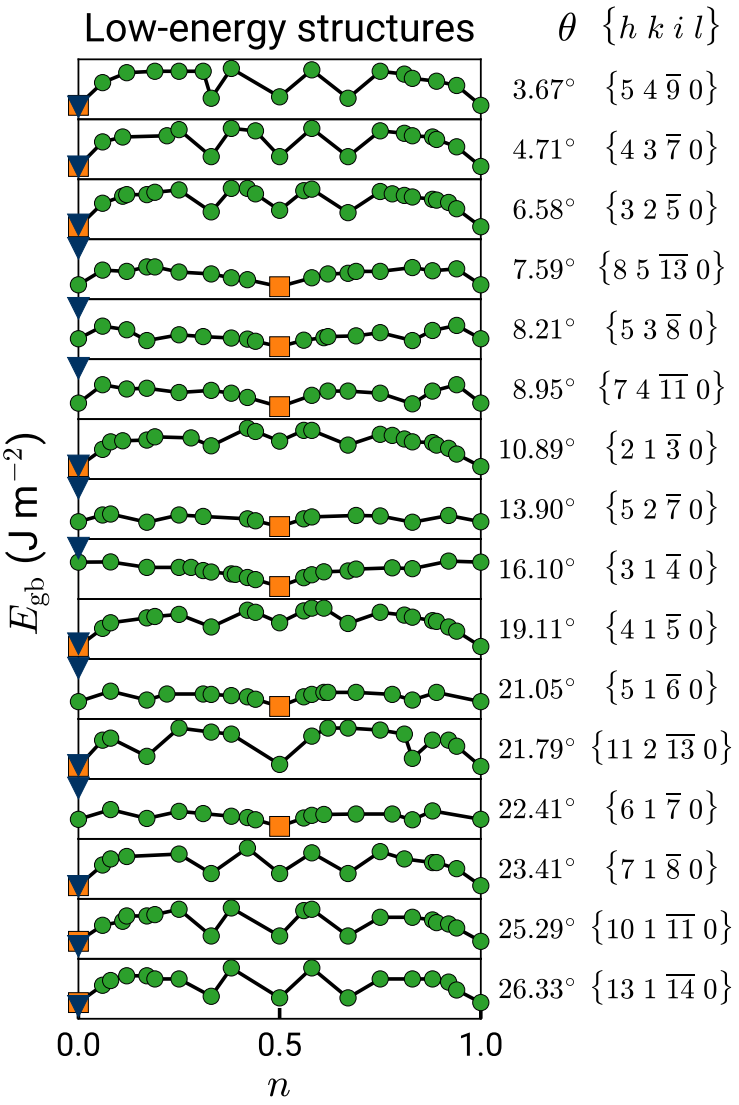


Figure B.2: Energy map of [0001] **STGBs simulated using an EAM potential** [188]. The profiles are qualitatively similar to those in Figure 4.2 in the main text, which was generated using a MEAM potential [189]. See Figure 4.2 for additional descriptions of features.

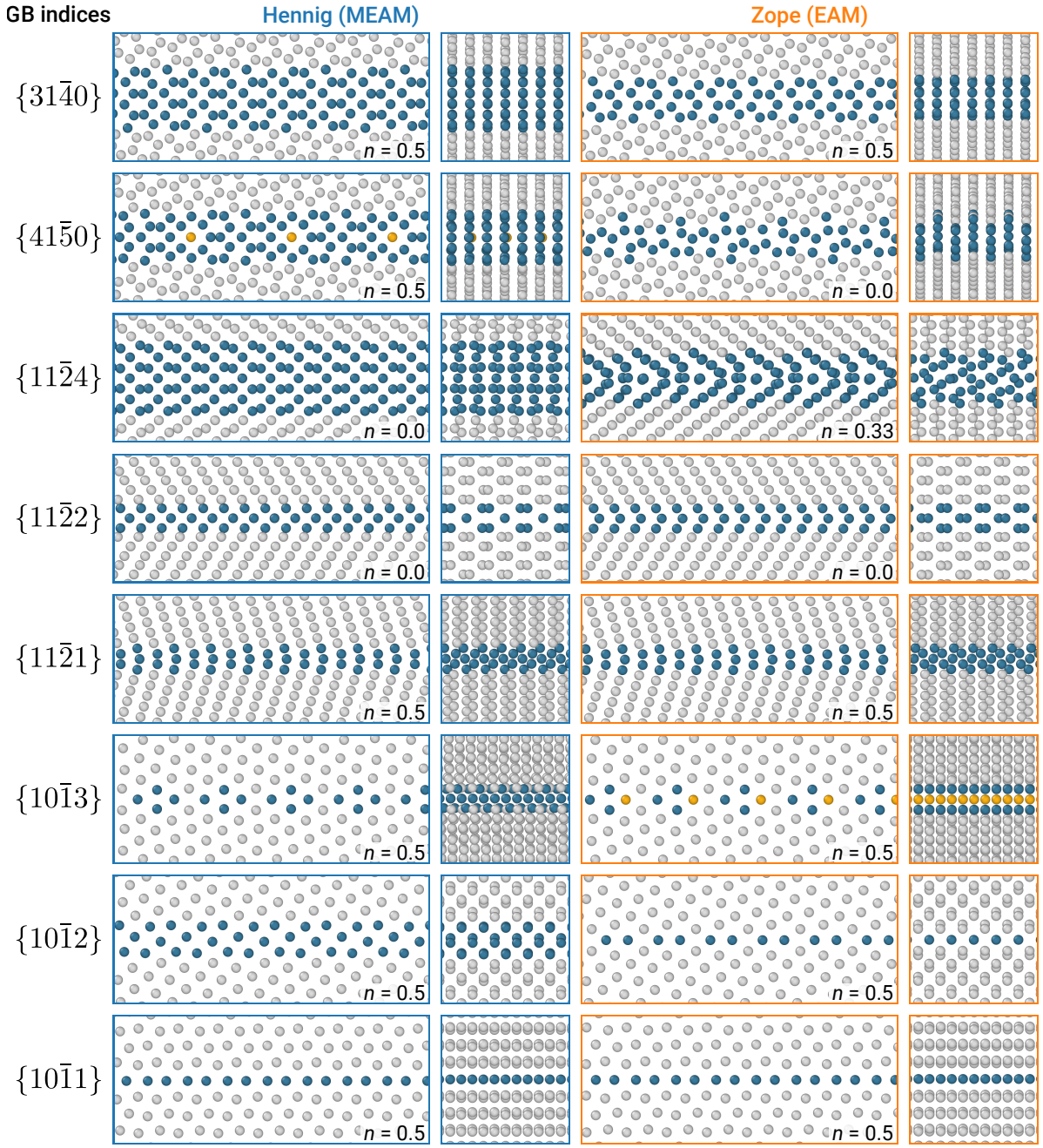


Figure B.3: Comparison of optimized GB structures. A few misorientations are provided for each tilt axis and two projections are shown for each ground state. The GB atomic density is shown in the bottom right corner and may not be equal for both potentials. The atoms are colored according to the common neighbor analysis (CNA) in OVITO [76], where gray are HCP-coordinated atoms, gold are FCC-coordinated atoms, and blue have a different coordination (in the GB).

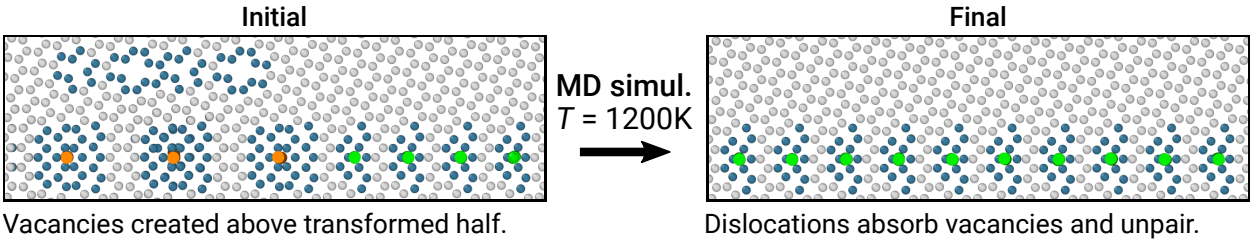


Figure B.4: (Reverse) phase transformation through vacancy absorption. In Figure 4.5, we demonstrate interstitial-induced phase transformation and coexistence in $\{21\bar{3}0\}[0001]$, where every two $\mathbf{b}_{\mathrm{I}}$ dislocations pair up to form one $\mathbf{b}_{\mathrm{II}}$ dislocation. Here, by injecting vacancies near the transformed region and performing MD simulations at 1200 K, we reverse the transformation, whereby the $\mathbf{b}_{\mathrm{II}}$ dislocations absorb the vacancies and unpair. The dislocations are identified using the dislocation extraction algorithm (DXA) in OVITO [191] and the colors are consistent with those in Figure 4.4.

Appendix C

MSE education

Major	Responses
MSE	32 (76 %)
Other	5 (12 %)
MSE & Other	5 (12 %)

(a)

Class	Responses
Sophomore	5 (12 %)
Junior	27 (64 %)
Senior & older	10 (24 %)

(b)

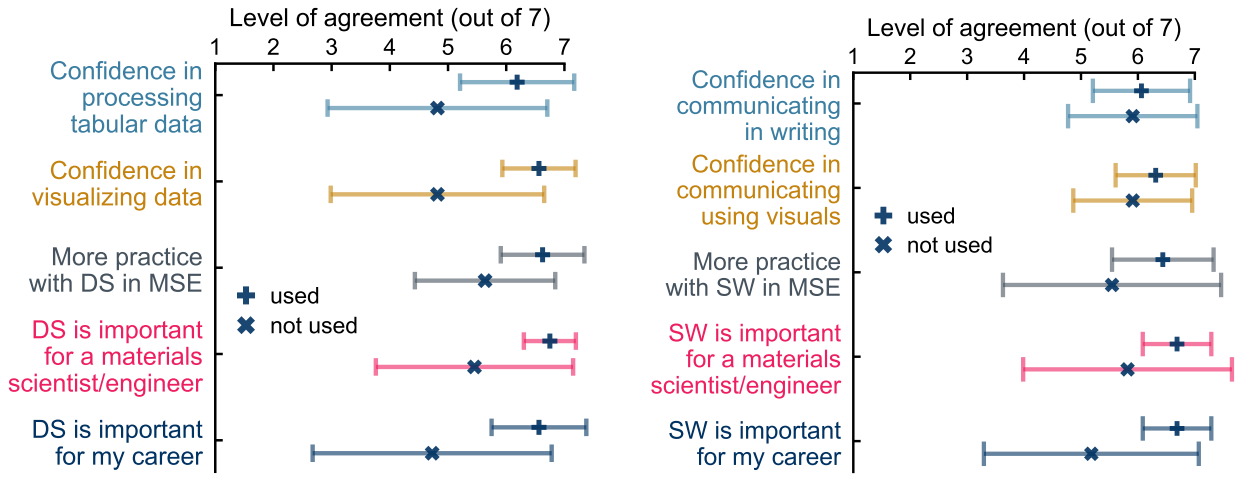
Gender	Responses
Male	25 (60 %)
Female	15 (36 %)
Non-binary	2 (5 %)

(c)

Table C.1: Student demographics ($n = 42$) in terms of (a) Major, (b) Class year, and (c) Gender. Note that "Junior" includes 1st-year transfer students and "Senior" includes 2nd-year transfer students, etc.

Language	Reason		
	Required	Choice	Not used
C/C++	2 (5 %)	4 (10 %)	**36 (86%)**
Mathematica	13 (31 %)	8 (19 %)	**21 (50%)**
MATLAB	**23 (55%)**	14 (33 %)	10 (24 %)
Python	11 (26 %)	**17 (40%)**	16 (38 %)
R	2 (5 %)	1 (2 %)	**40 (95%)**
Other	5 (12 %)	8 (19 %)	**29 (69%)**

Table C.2: Previous programming languages used by students ($n = 42$) *in their MSE courses only* and whether it was a course requirement, personal choice, or not used at all. The most common reason for each language is in boldface to guide the eye. Note that multiple choices were allowed, so rows may not sum to 100 %.



(a) Data science (DS) **(b)** Scientific writing (SW)

Figure C.1: Post-course survey results for student sentiment on (a) data science and (b) scientific writing, where the results from Fig. 5.4 are stratified by whether students used the modules ($n_{\text{used}} = 18$, upper bars) or not ($n_{\text{not}} = 11$, lower bars).

Lab topic	Responses
Lab 1: X-ray emission	12
Lab 2: Powder XRD	11
Lab 3: Precision XRD	13
Lab 4: EDS in the SEM	7

Table C.3: Student responses to "Which labs did you use the modules for?"

Motivation	**%**
Helpful for the reports	78 %
Easy to use	56 %
Supported learning MSE concepts	56 %
Improve Python skills	56 %
Already planned to use Python	33 %
Improve DS skills	33 %

(a) Used the modules, $n_{\text{used}} = 18$.

Reason	**%**
Preferred own method	39 %
Too time consuming	22 %
Adapted to MATLAB	11 %
Would distract from MSE	11 %
Too difficult to understand	11 %
Not helpful for the reports	6 %

(b) Did not use the modules, $n_{\text{not}} = 11$.

Table C.4: Top reasons for (a) using and (b) not using the Python-based online modules. Percentage is given out of the number of students who identified with each group. Multiple selections were allowed and thus percentages may not sum to 100 %.

Question	Topic	Counts
Most helpful about the notebooks?	Plotting w/ Python	7
	Data processing w/ Python	6
	Connections to lab	4
Least helpful about the notebooks?	Nothing/ I don't know	8
	Need more explanation	7
	Jupyter environment confusing	5
What could we have done differently to encourage use?	Nothing/ I don't know	6
	Required course/assignment	2
	Tutorial videos/slides	2
How have your perspectives on DS/SW changed?	No change	3
	Should learn more	3
	DS/SW should be required	3

Table C.5: Inductively coded student responses to free-response questions in the post-course survey. The first two questions are only asked to those who used the notebooks ($n_{\text{used}} = 18$), the third is only asked to those who didn't use the notebooks ($n_{\text{not}} = 11$), and the last question is asked to everyone. Only the most common topics are shown.