TECHNISCHE
UNIVERSITÄT
WIEN

DIPLOMARBEIT

Machine Learning Force Field for Dynamic Oxidation of Silicon

ausgeführt am

Institut für
Mikroelektronik
TU Wien

unter der Anleitung von

Univ.Prof. Dipl.-Ing. Dr.techn. Tibor Grasser
Univ.Ass. Dipl.-Ing. Lukas Cvitkovich
Univ.Ass. Dipl.-Ing. Christoph Wilhelmer

durch

Ing. Franz Fehringer BSc.
[redacted]

Wien, am 28. August 2024

Abstract

This master's thesis is dedicated to the development of an interatomic potential based on machine learning (ML) for the dynamic simulation of the thermal oxidation of silicon. This potential is based on the *Gaussian Approximation Potential* (GAP), a successful ML algorithm specifically designed to develop interatomic potentials for molecular dynamics (MD) simulations.

Silicon (Si) in combination with its native oxide SiO_2 has been of central importance in semiconductor technology for decades. The progressive miniaturization of the devices used in this field requires a precise understanding of the complex oxidation process at the atomic level. Since other methods suffer from shortcomings in accuracy (classical force fields) or computational feasibility (*ab-initio* MD) when simulating the thermal oxidation of Si, ML interatomic potentials offer a promising alternative to overcome these limitations.

The use of ML adds a highly efficient variant to conventional simulation methods. The GAP model is trained with data from the highly complex and accurate density functional theory (DFT). Therefore, the employment of a GAP trained on DFT data in MD simulations promises enormous savings in the required computing power while maintaining almost the same accuracy.

The main goal of this thesis is to develop a reliable GAP that allows MD simulations of larger systems on longer time scales. The results allow the generation of realistic Si/SiO_2 interfaces and provide insights into the oxidation kinetics as well as the atomic structure of this material system.

Kurzfassung

Die vorliegende Diplomarbeit widmet sich der Entwicklung eines interatomaren Potentials auf Basis von Machine Learning (ML) zur dynamischen Simulation der thermischen Oxidation von Silizium. Als Grundlage für dieses Potential dient das *Gaussian Approximation Potential* (GAP), ein erfolgreicher ML Algorithmus, der speziell zur Entwicklung von interatomaren Potentialen für Molekulardynamik-Simulationen (MD) entworfen wurde.

Silizium (Si) in Verbindung mit seinem nativen Oxid SiO_2 hat seit Jahrzehnten eine zentrale Bedeutung in der Halbleitertechnologie. Die fortschreitende Miniaturisierung der dort eingesetzten Bauelemente erfordert genaues Verständnis des komplexen Oxidationsprozesses bis hin zur atomaren Ebene. Da andere Methoden zur Simulation der thermischen Oxidation von Silizium entweder an Genauigkeitsmängeln (klassische Kraftfelder) oder an der rechnerischen Durchführbarkeit (*ab-initio* MD) scheitern, bieten interatomare maschinell gelernte Potenziale eine vielversprechende Alternative, um diese Einschränkungen zu überwinden.

Durch den Einsatz von ML werden die herkömmlichen Simulationsmethoden um eine höchst effiziente Variante erweitert. Das GAP-Modell wird mit Daten aus der hochkomplexen und genauen Dichtefunktionaltheorie (DFT) trainiert. Daher verspricht der Einsatz eines auf DFT-Daten trainierten GAPs in MD-Simulationen enorme Einsparungen bei der erforderlichen Rechenleistung bei nahezu gleicher Genauigkeit.

Das Hauptziel dieser Arbeit besteht darin, ein zuverlässiges GAP zu entwickeln, welches MD Simulationen von größeren Systemen auf längeren Zeitskalen ermöglicht. Die Ergebnisse dieser Arbeit erlauben das Generieren von realistischen Si/SiO_2 Grenzflächen und geben Einblicke in die Oxidationskinetik sowie den atomaren Aufbau dieses Materialsystems.

Acknowledgement

I want to express my sincere appreciation to my supervisor Prof. Tibor Grasser for the opportunity to write my master's thesis at his Institute of Microelectronics. I also express my deepest gratitude to Univ.Ass. Lukas Cvitkovich for his invaluable guidance, support, and encouragement throughout this research. His expertise and dedication have been instrumental in shaping this thesis. Additionally, I would like to thank Christoph Wilhelmer for his continuous support, insightful feedback, and valuable contributions, which have significantly enhanced the quality of this work.

I especially thank Dominic Waldhör and Diego Milardovich for their countless tips and helpful discussions. They contributed to a deeper comprehension of my research topic.

Furthermore, I express my gratitude to my colleagues Robert Stella, Anna Benzer, Angus Gentles, and Martin Loesener for their social support during difficult times and for their thoughtful conversations and feedback that contributed to the development of this thesis.

In addition, I am very grateful to my family for their encouragement, understanding, and financial support throughout my academic journey. All of this would not have been possible without them.

Finally, I would like to thank my beloved girlfriend Daphne from the bottom of my heart for her incredible patience and amazing support over the past few months as I worked on my thesis.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am 28. August 2024

Franz Fehringer

Contents

1 Introduction

SiO_2, the native oxide of silicon, still plays a crucial role in the performance and reliability of today's semiconductor devices such as metal oxide semiconductor field effect transistors (MOSFETs) [1]. Silicon, a cornerstone material in the electronics industry, is subjected to complex oxidation processes to generate Si/SiO_2 structures as employed in a wide range of innovative applications including semiconductor spin qubits [2], spintronics [3, 4], and single-electron devices [5]. The significant advantage of silicon lies in its ability to produce high-quality semiconductor/insulator interfaces, characterized by low defect densities [6] and the ease of forming oxide directly on the substrate through thermal oxidation [7]. The understanding of chemical reactions at the atomic scale is essential for advancing materials science, particularly in the field of semiconductor technology.

Molecular Dynamics (MD) simulations have proven invaluable for studying these reactions [8], providing insights into the intricate details of silicon oxidation [9]. Traditional MD simulations, while powerful, often face challenges in accurately capturing the dynamics of large-scale systems due to an inadequate description of quantum-mechanical effects. Therefore, machine learning (ML) techniques have emerged as promising tools to augment traditional simulation methods [10].

Density Functional Theory (DFT) is a widely used quantum mechanical method that provides a good balance between accuracy and computational feasibility for many systems. While DFT offers a significant improvement in accuracy compared to simpler methods, it can become computationally expensive, particularly for large or complex systems, due to the detailed treatment of electron correlation effects [11, 12].

This master thesis focuses on the development and training of a machine learning potential tailored specifically for molecular dynamics simulations of silicon oxidation. The training of the ML potential was carried out within the Gaussian approximation potential (GAP) method[13]. Unlike classical force fields, which are often parameterized based on empirical data or simplified models [8, 14], the machine learning potential in this thesis was trained using high-fidelity density functional theory data. This training method allows the ML potential to capture complex electronic interactions and subtleties in the oxidation process [15], leading to much higher accuracy in the simulations. As a result, the ML potential offers significantly improved precision in modeling the behavior of the system compared to traditional force fields, thus

providing a more accurate and reliable representation of the material's properties [16].

The oxidation of silicon is a dynamic process [9, 17], making it a compelling but intricate subject of study. Silicon dioxide layers with thicknesses on the order of nanometers are usually created by thermal oxidation of silicon. The fundamental mechanisms driving this process have been thoroughly investigated over many years, utilizing both experimental and theoretical approaches [18–26]. Traditional model approaches, such as the well-known Deal-Grove model [18], are effective in describing oxidation processes when the oxide layer has reached substantial thicknesses greater than 15 nm. However, they do not accurately capture the dynamics of the initial oxidation phase [27, 28], which is crucial for applications involving very thin oxide layers in the nanometer range.

The machine-learned interatomic potential presented in this thesis enables the thermal oxidation process to be modeled inside dynamic simulations, beginning with silicon surface structures that are completely oxygen-free. In addition to examining flat silicon surfaces, the investigation are extended to encompass more complex surface geometries, such as cylindrical silicon nanowires. These structures are subjected to an O_2 gas in the MD simulation, which reacts with the surface to form an amorphous SiO_2 coating layer. The ML potential aims to capture the intricate energy landscapes and reactive pathways associated with silicon oxidation. The main goal of this thesis is to bridge the gap between accuracy and computational efficiency by leveraging the capabilities of machine learning algorithms. The trained model is expected to provide a reliable representation of the potential energy surface (PES) and enable accelerated simulations without compromising accuracy. The overall goal is to improve our understanding of silicon oxidation kinetics and thermodynamics.

The thesis is structured as follows: Chapter 2 describes the properties of Si and SiO_2, followed by the theoretical background to density functional theory (DFT) and an introductory description of the CP2K software in Chapter 3. Chapter 4 is dedicated to the theory of molecular dynamics, with reference to the MD software LAMMPS. The theoretical part is concluded in Chapter 5 focusing on details about the gap algorithm and its training. The practical part of this thesis starts with the creation of the training structures, which will be described in Chapter 6. Chapter 7 explains how the various GAPs were trained. The methods for validating the trained GAPs are found in Chapter 8. Chapter 9 summarizes the results of the practical part and shows, that the trained ML model is broadly applicable to gaseous oxygen, crystalline Si, and amorphous SiO_2. Compared to the the reactive force field (ReaxFF) the ML potential produces much more realistic interface structures. Finally, an outlook for future investigations is given in Chapter 10.

2 Si and SiO_2

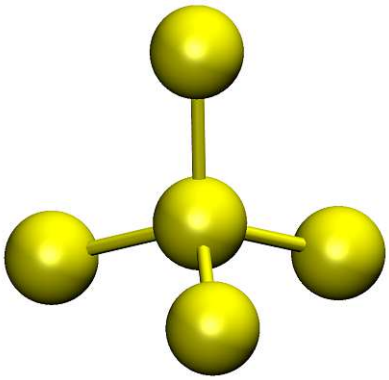
Silicon (Si) and its natural oxide SiO_2 are still essential for the development of cutting-edge device technologies today and playing an exceptional role in the semiconductor industry [29]. Usually, Si is thermally oxidized to create SiO_2 layers, whereby today's layer thicknesses of dielectric amorphous SiO_2 (a-SiO_2) are in the order of a few nanometers. For example, in modern semiconductors, the SiO_2 gate of metal oxide semiconductor field effect transistors (MOSFETs) is increasingly being replaced by so-called high-κ dielectrics with high permittivity [30]. However, a thin layer of SiO_2 is still required to ensure good interface quality with the high-κ dielectric.

The entire mechanism of Si oxidation and the formation of an oxide layer is far from completely understood. Cvitkovich et al. [9] have demonstrated that most Si oxidation processes may be explained by a variety of theoretical models. The initial oxidation process can be described by chemisorption, where O_2 molecules dissociate and are adsorbed on the Si surface. As a result, the oxide layer grows rapidly at the beginning. The subsequent layer growth is then increasingly determined by the diffusion of oxygen through the growing oxide layer. This stage of the oxidation process can be described by the Deal and Grove model [18]. This severely limits the velocity at which the oxide layer forms.

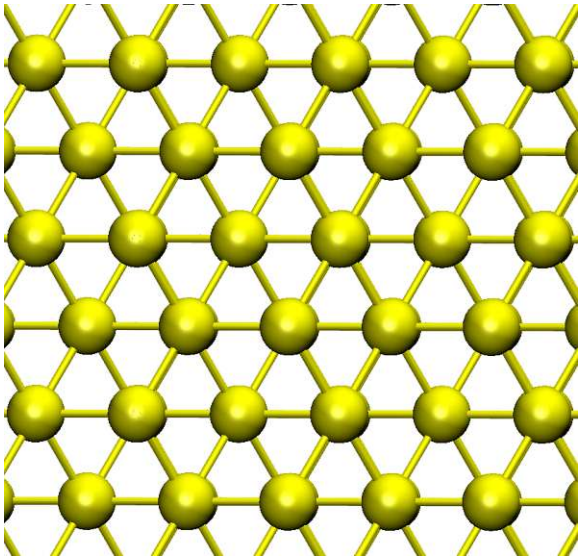
In the following, some basic properties of Si and SiO_2 will be discussed. The properties of the oxide depend strongly on its growth kinetics. For modern MOSFETs, it is essential, that these oxide layers are defect-free and avoid charge trapping at the interface.

2.1 Properties of Si

The electron configuration of Si is $1s^2 2s^2 2p^6 3s^2 3p^2$ whereas the $3s$ and the three $3p$ orbitals hybridize resulting in four equivalent sp^3 orbitals, which are tetrahedrally arranged. This is illustrated in Fig. 2.1, where panel (a) shows a silicon atom with its four neighbor silicon atoms positioned in the tetrahedron corners. As a result of the sp^3 hybridization, crystalline Si forms a diamond structure, which consists of two face-centered cubics (fcc), shifted against each other by 1/4 of the space diagonal of the cubic crystal. In Fig. 2.1 (b), a silicon crystal in the (100) plane is shown. Here, a denotes the length of the diamond unit cell. Consequently, the two neighboring Si



(a)



(b)

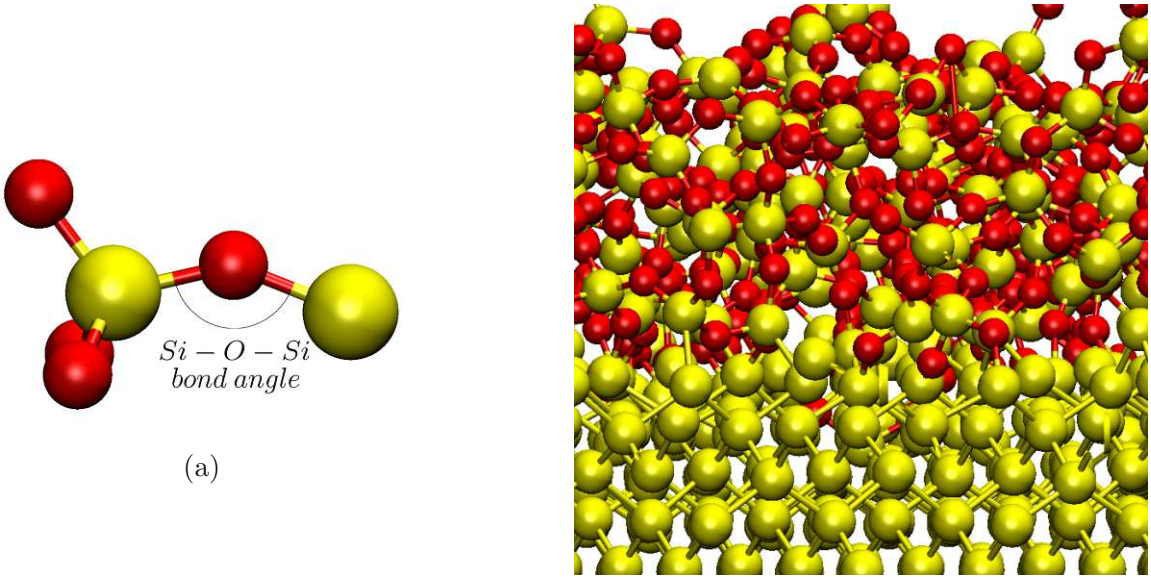
Figure 2.1: Crystalline Silicon. Panel (a) shows a silicon atom with its four tetrahedrally arranged neighbors due to the sp^3 orbitals. Panel (b) shows crystalline silicon in the diamond structure in the (100) plane.

atoms have a distance of $a(\sqrt{3}/4) = 2.35\,\text{Å}$ [31].

Any surface of a crystal causes a symmetry break, which can be quantified by a surface energy γ. To minimize the surface energy, the crystal undergoes relaxations and reconstructions (Wulff shapes [32]) and tries to reduce the unsaturated, dangling bonds. Since different crystallographic directions usually have different surfaces, the surface energy γ depends on the orientation of the surface. This affects how strongly other atoms (e.g. oxygen) are getting adsorbed and bonded to the surface atoms. When manufacturing silicon wafers for electronic devices, the wafers are usually orientated in the {100}, {111}, and {110} planes [33].

2.2 Properties of SiO_2

Oxygen reacts chemically with silicon when it comes into contact with its surface resulting in an amorphous SiO_2 layer. Silicon dioxide is an excellent insulator that is stable both electrically and mechanically. There are nine known allotropic forms of SiO_2, whereas eight of them have a tetrahedral structure [34]. In this modification, one Si atom and four O atoms are bound via the four sp^3 orbitals of Si and the $2p$ orbital of O. Consequently one silicon atom is coordinated by four oxygen atoms while two silicon atoms are connected by an oxygen atom. The tetrahedral form of



(a) (b)

Figure 2.2: SiO_2 structure. Panel (a) shows two silicon atoms (yellow) that are bonded by an oxygen atom (red). The four oxygen atoms of the left Si atom are tetrahedrally arranged. Panel (b) shows a section of an amorphous SiO_2 layer grown on an silicon surface.

SiO_2 is visualized in Fig. 2.1a by assuming one Si atom in the middle and four O atoms at the end of each sp^3 orbitals. The Si–O bond length ranges from 1.52 Å to 1.69 Å. The tetrahedral O–Si–O bond angle is 109.18° whereas the Si–O–Si bond angle varies from 120° to 180°[35]. Fig. 2.2a illustrates two silicon atoms, which are connected via an oxygen atom. The full amorphous structure of an SiO_2 layer is given in Fig. 2.2b.

Defects within the oxide layer can significantly degrade the performance of electronic devices, leading to increased leakage currents, reduced reliability, and compromised overall efficiency [36, 37]. Unsaturated dangling bonds, which are a source of charge trapping center in the oxide, determine the stability and reliability of semiconductor applications like modern MOSFETs [38, 39]. During operation, these defects can trap charges from the substrate or gate. A controlled SiO_2 oxidation process is therefore essential.

2.3 Oxidation of Si

Over the last decades, many different techniques to form a SiO_2 layer have been proposed [18, 40, 41]. The following two techniques will be described in more detail:

- Thermal oxidation
- Tetraethyl orthosilicate (TEOS) Oxide Deposition

2.3.1 Thermal oxidation

During thermal oxidation, which usually takes place at a temperature of about 1000 °C [9, 42, 43], O_2 gas is brought into contact with the silicon surface. Initially, the O_2 molecules dissociate near the clean and reconstructed Si surface and get adsorbed based on chemisorption events [9]. The SiO_2 layer is growing rather quickly at this stage. After the first oxide layer has formed, the growth of the oxide layer slows down [9]. This stage of the oxidation process is not yet fully understood. It turns out that an extended Deal-Grove model with non-physical exponential terms [44] and models that include oxidation reactions in the oxide layer agree with experimental data [45] and provide a partial explanation.

After reaching an oxide thickness > 30 Å, the oxidation process is governed purely by the oxygen diffusion through the SiO_2 layer. The oxide growth in this stage is accurately described by the Deal-Grove model. The oxidation reaction is assumed to occur at the interface between Si and SiO_2 after the O_2 molecule diffuses through the oxide layer. Because of the low diffusion coefficient of oxygen in SiO_2, growing a thick oxygen layer takes a long time.

2.3.2 TEOS oxide deposition

In contrast to thermal oxidation, where the silicon for the oxide layer originates only from the substrate, silicon is also present in the gas in the TEOS gas phase deposition processes.

TEOS is the acronym for tetraethyl orthosilicate $SiO_4C_8H_{20}$. During this procedure, liquid TEOS is evaporated in a vacuum chamber. In the next step, the ethyl groups are separated from the TEOS molecule at a temperature of $700 - 750$ °C and are removed. The oxide layer is formed by the deposition of SiO_2 on the Si substrate. The chemical reaction of this process is given by [42]:

$$SiO_4C_8\,H_{20} \longrightarrow SiO_2 + 2\,H_2O + \text{byproducts}$$

This low-pressure chemical vapor deposition (LPCVD) process creates an oxide layer with high electrical stability, whereby the pressure and the process temperature determine the final quality of the SiO_2 layer. Compared to thermal oxidation, the growth rate here is higher and can reach up to several nm per minute [46, 47].

3 Density functional theory - DFT

Density functional theory (DFT) is a useful tool and enabled technique for analyzing and investigating the electronic structure and properties of molecules and materials, especially of condensed matter. By solving approximated versions of the Schrödinger equation, DFT provides a practical and efficient approach to studying many-body problems at a fundamental level [48]. This chapter summarizes the theoretical foundations behind DFT and their application, focusing on the code utilized in the software CP2K [49].

3.1 From one particle to many particles

To understand the behavior of a quantum particle, one has to solve the Schrödinger equation [50] by determining the corresponding wavefunction $\psi(\mathbf{r})$. $\psi(\mathbf{r})$ denotes the one-body wavefunction with the position vector $\mathbf{r}$. The probability of finding the particle at position $\mathbf{r}$ is given by $|\psi(\mathbf{r})|^2$ [51].

With $\hat{H}$ as the Hamiltonian operator, the symbolic form of the time-independent, one-body Schrödinger equation can expressed as following Eigenvalue problem [52]:

$$(E_{\text{kin}} + E_{\text{pot}})\psi = \hat{H}\psi(\mathbf{r}) = E\psi(\mathbf{r}) \tag{3.1}$$

where E_{kin} denotes the kinetic energy, E_{pot} the potential energy of the system and E the eigenenergy corresponding to the eigenfunction $\psi(\mathbf{r})$. For an electron with mass m_{e} in an potential field $V(\mathbf{r})$ equation (3.1) becomes

$$\left[\frac{\mathbf{p}}{2m_e} + V(\mathbf{r})\right]\psi(\mathbf{r}) = E\psi(\mathbf{r}) \tag{3.2}$$

$\mathbf{p}$ is the quantum-mechanical momentum operator and is given by

$$\mathbf{p} = -i\hbar\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}\right) = -i\hbar\nabla \tag{3.3}$$

Here, $\hbar$ denotes the reduced Planck constant and ∇ represents the Nabla operator:

$$\nabla = \frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z} \tag{3.4}$$

We now add another electron to the system. The energy of the Coulomb repulsion of both particles with a distance d_{ee} is given by [53]:

$$E_{\mathrm{ee}} = \frac{e^2}{4\pi\epsilon_0 d_{\mathrm{ee}}} \tag{3.5}$$

Here, e is the electron charge, and ϵ_0 denotes the permittivity of the vacuum. This repulsive interaction changes the potential term $V(\mathbf{r})$ from the equation (3.2) and therefore the eigenvalues may differ compared to the one-electron Schrödinger equation.

If we additionally include nuclei in our analysis, we also need to take the repulsion energy between two nuclei, E_{nn}, and the attraction energy between an electron and a nuclei, E_{en}, into account:

$$E_{\mathrm{nn}} = \frac{Z^2 e^2}{4\pi\epsilon_0 d_{\mathrm{nn}}} \tag{3.6}$$

$$E_{\mathrm{en}} = -\frac{Z e^2}{4\pi\epsilon_0 d_{\mathrm{en}}} \tag{3.7}$$

Z stands for the atomic number of the nuclei, d_{nn} for the distance between the two nuclei, and d_{en} for the distance between the electron and the nuclei. Note the minus sign in equation (3.7), which indicates an attractive force between the electron and the nuclei. E_{nn} and E_{en} also may change $V(\mathbf{r})$.

A realistic and practical theory of materials requires a correct description of systems including numerous nuclei and electrons. We now considering N electrons at positions $\mathbf{r}_{\mathrm{N}}$ with $N \in 1..N$ and M nuclei at positions $\mathbf{R}_{\mathrm{M}}$ with $M \in 1..M$. We introduce the many-body wavefunction Ψ which is a function of all the coordinates of the electrons and the nuclei [52]:

$$\Psi = \Psi(\mathbf{r}_1..\mathbf{r}_{\mathrm{N}}; \mathbf{R}_1..\mathbf{R}_{\mathrm{M}}) \tag{3.8}$$

The probability of concurrently finding the first electron at position $\mathbf{r}_1$, the first nuclei at position $\mathbf{R}_1$ and so on, is given by $|\Psi(\mathbf{r}, \mathbf{R})|^2$.

Similar to eq. (3.1), the time-independent, many-body Schrödinger equation can now be written as:

$$(E_{\mathrm{kin}} + E_{\mathrm{pot}})\Psi = \hat{H}\Psi = E\Psi \tag{3.9}$$

The kinetic energy of the many-body system, $T_{e+n}(\mathbf{r},\mathbf{R})$, consists of a term for the kinetic energies of the N electrons, $T_e(\mathbf{r})$, and a term for the kinetic energies of the M nuclei, $T_n(\mathbf{R})$ [52]:

$$T_{e+n}(\mathbf{r},\mathbf{R}) = T_e(\mathbf{r}) + T_n(\mathbf{R}) = -\sum_{i=1}^{N} \frac{\hbar^2}{2m_{\mathrm{e}}}\Delta_{\mathrm{i}} - \sum_{I=1}^{M} \frac{\hbar^2}{2M_{\mathrm{I}}}\Delta_{\mathrm{I}} \tag{3.10}$$

Here, m_e denotes the electron masses and M_I denotes the nuclei's masses. Δ denotes the Laplace operator and is given by:

$$\Delta_{i;I} = \frac{\partial^2}{\partial x_{i;I}^2} + \frac{\partial^2}{\partial y_{i;I}^2} + \frac{\partial^2}{\partial z_{i;I}^2} \tag{3.11}$$

The potential energy terms of equations (3.5) - (3.7) can be extended to multiple electrons and nuclei. Therefore the Coulomb repulsion between electron pairs reads as:

$$V_{ee}(\mathbf{r}) = \frac{1}{2} \sum_{i \neq j} \frac{e^2}{4\pi\epsilon_0} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} \tag{3.12}$$

The indices i and j range from 1 to N and the factor $1/2$ is required to count each pair only once. Note that an electron can not repel itself, so $i \neq j$ is required. Similar to the equation (3.12) the Coulomb repulsion between pairs of nuclei leads to

$$V_{nn}(\mathbf{R}) = \frac{1}{2} \sum_{I \neq J} \frac{e^2}{4\pi\epsilon_0} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \tag{3.13}$$

Here, the indices I and J range from 1 to M and Z_I and Z_J denote the atomic number of nuclei I and nuclei J, respectively. With the same argumentation as before, the $1/2$ and also $I \neq J$ are required here. For the Coulomb attraction between electrons and nuclei, we obtain the following equation:

$$V_{en}(\mathbf{r},\mathbf{R}) = -\sum_{i,I} \frac{e^2}{4\pi\epsilon_0} \frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} \tag{3.14}$$

The indices i range from 1 to N, the indices J range from 1 to M, and Z_I denotes the atomic number of nuclei I. The minus indicates again an attractive force between the electrons and the nuclei.

With the equations (3.10) - (3.14) the time-independent, many-body Schrödinger equation is given by

$$\left[T_e(\mathbf{r}) + T_n(\mathbf{R}) + V_{ee}(\mathbf{r}) + V_{nn}(\mathbf{R}) + V_{en}(\mathbf{r},\mathbf{R})\right] \Psi = \hat{H}\Psi = E\Psi \tag{3.15}$$

3.2 Born-Oppenheimer approximation

The problem is now to find an eigenfunction Ψ that solves Equation (3.15). Finding such a wavefunction Ψ is analytical only possible for the H atom and the He^+ ion. With increasing electrons and nuclei, the number of different configurations exponentially increases with the number of electrons and nuclei (hitting the „exponential wall“ [54]) and an analytical solution is no longer feasible.

Born-Oppenheimer presented a highly efficient approximation (also called adiabatic approximation) in the 1920s [55], where the many-body Schrödinger equation (3.15) is divided into two parts. Due to the much higher mass of the nuclei compared to that of the electrons, the nuclei move much more slowly than the fast-moving electrons, allowing the nuclei to be treated as nearly constant while solving the electronic Schrödinger equation. Vice versa, the electrons can be treated as a constant electronic charge distribution while solving the Schrödinger equation for the nuclei [56]. This assumption leads to a many-body wavefunction Ψ that can be expressed as a product of a wavefunction for the electrons, ϕ, and a wavefunction for the nuclei, η: $\Psi = \phi \cdot \eta$

The time-independent, many-body Schrödinger equation for the electrons is then given by the following eigenvalue equation:

$$\hat{H}_e \phi_h(\mathbf{r},\mathbf{R}) = [T_e(\mathbf{r}) + V_{ee}(\mathbf{r}) + V_{en}(\mathbf{r})]\, \phi_h(\mathbf{r},\mathbf{R}) = E_h(\mathbf{R})\phi_h(\mathbf{r},\mathbf{R}) \tag{3.16}$$

where $\hat{H}_e$ is the Hamiltonian operator for the electrons and $\phi_h(\mathbf{r},\mathbf{R})$ are the eigenstates to the eigenvalues $E_h(\mathbf{R})$, whereas the positions of the electrons are collected by $\mathbf{r}$. Note that the positions of the nuclei, $\mathbf{R}$, are fixed and only appear here as a parameter [57].

The second part of the time-independent, many-body Schrödinger equation, which describes the motion of the nuclei in the field of electronic charge distribution, is given by:

$$\hat{H}_n \eta_{hk}(\mathbf{R}) = [T_n(\mathbf{R}) + V_{nn}(\mathbf{R}) + E_h(\mathbf{R})]\, \eta_{hk}(\mathbf{R}) = E_{hk}(\mathbf{R})\eta_{hk}(\mathbf{R}) \tag{3.17}$$

$\hat{H}_n$ denotes the Hamiltonian operator for the nuclei and $\eta_{hk}(\mathbf{R})$ gives the eigenstates to the eigenvalues $E_{hk}(\mathbf{R})$. Note that $\hat{H}_n$ includes the eigenvalues $E_h(\mathbf{R})$ of equation (3.16).

3.3 The Hohenberg-Kohn Theorems

DFT as a whole is based on two theorems established by Kohn and Hohenberg 1964 [58]. The Hohenberg-Kohn Theorems and the Kohn-Scham Equations, which will be discussed in the next section, transform the many-body problem of interacting electrons into a problem of non-interacting single electrons.

The Hamiltonian $\hat{H}$ of a many-electron system with energy $E = \langle\Psi|\hat{H}|\Psi\rangle$ is independent of the specific material being studied, therefore each modification in E is linked to modifications of Ψ [52]. This observation can be expressed in the form that the energy E is a functional of Ψ (represented by the square brackets):

$$E = \mathcal{F}[\Psi] \tag{3.18}$$

1. Hohenberg-Kohn Theorem

The first Hohenberg-Kohn theorem states that the ground-state energy E from Schrödinger's equation is a special functional of the electron density $\rho(\mathbf{r})$ [11, 58]:

$$E = \mathcal{F}[\rho(\mathbf{r})] = E[\rho] \tag{3.19}$$

This theorem implies that the total energy E is a functional of the electron density $\rho(\mathbf{r})$ and is all that is required to calculate the total energy in the ground state and $\rho(\mathbf{r})$ alone determines all the features of the electronic ground state. As a consequence, the many-electron Schrödinger equation is now a function of just 3 spatial variables instead of 3N spatial variables (N denotes the number of electrons).

2. Hohenberg-Kohn Theorem

The second Hohenberg-Kohn theorem states that that the energy functional $E[\rho]$ of the electron density $\rho(\mathbf{r})$ reaches its minimum value at the correct ground-state density $\rho_0(\mathbf{r})$ for a many-body system. This theorem implies that the true ground-state electron density is the one that minimizes the energy functional:

$$E[\rho] \geq E[\rho_0] \tag{3.20}$$

where $E[\rho]$ represents the energy as a functional of the electron density, and $E[\rho_0]$ is the actual ground-state energy. This principle allows the determination of the ground-state energy of a system by varying the electron density, without the need to know the full many-body wavefunction.

The total energy E can be expressed as a sum of two energy terms [51]:

$$E[\rho] = E_{\text{known}}[\rho] + E_{\text{XC}}[\rho] \tag{3.21}$$

The first term, $E_{\text{known}}[\rho]$, collects all known energy components whereas the second term, $E_{\text{XC}}[\rho]$, includes all quantum mechanical phenomena that are not covered by $E_{\text{known}}[\rho]$. XC is the acronym for exchange correlation. The determination of E_{XC} is the main issue in DFT as will be discussed in more detail in section 3.4.

3.3.1 The Kohn-Scham Equations

Kohn and Scham proved in 1965, that a set of equations, known as Kohn-Sham equations, can determine the minimal energy of a system [12], where each equation involves only one electron. The Kohn-Sham equations transform the many-body problem of interacting electrons into a system of non-interacting electrons moving in an effective potential. This effective potential includes contributions from the exchange-correlation effects, which account for the complex many-body interactions.

Each electron is represented by a single-electron wavefunctions $\psi_i(\mathbf{r})$, therefore each Kohn-Scham equation depends on only three spatial variables. The single-electron wavefunctions, which describe non-interacting Kohn-Scham orbitals, must be orthogonal and resemble the same electron density $\rho(\mathbf{r})$ as the original system [12, 52]:

$$\rho(\mathbf{r}) = 2\sum_i \psi_i^*(\mathbf{r})\psi_i(\mathbf{r}) \tag{3.22}$$

ψ_i^* is the conjugate complex wavefunction of ψ_i and the factor 2 is due the Pauli exclusion principle.

The Kohn-Scham equations have the following form:

$$\left[T_e(\mathbf{r}) + V_{en}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r})\right]\psi_i(\mathbf{r}) = \epsilon_i\psi_i(\mathbf{r}) \tag{3.23}$$

Here, T_e denotes the kinetic energy of the N non-interacting single electrons, see equation (3.10). V_{en} denotes the Coulomb attraction between electrons and nuclei, see equation (3.14). V_H denotes the so-called Hartree potential and describes the Coulomb repulsion between the one electron considered in one of the Kohn-Scham orbitals and the total electron density defined by the rest of the electrons of the system. V_{XC} denotes the exchange-correlation potential. $V_{en}(\mathbf{r})$, $V_H(\mathbf{r})$ and $V_{XC}(\mathbf{r})$), can be interpreted as an external potential $V_{eff}(\mathbf{r})$ for the electrons:

$$V_{eff}(\mathbf{r}) = V_{en}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r}) \tag{3.24}$$

whereas $V_{XC}(\mathbf{r})$ can be expressed as the functional derivative of $E_{XC}[\rho]$:

$$V_{XC}(\mathbf{r}) = \frac{\delta E_{XC}[\rho]}{\delta\rho} \tag{3.25}$$

The Hartree potential is given by:

$$V_H(\mathbf{r}) = e^2\int \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d^3r' \tag{3.26}$$

with e as the electron charge.

After specifying the exchange-correlation functional $E_{XC}[\rho]$, the total energy E and the electron density $\rho(\mathbf{r})$ of a system in the ground state can be determined self consistently by using the following procedure [11]:

(1) Define a initial electron density $\rho(\mathbf{r})$ to start with.

(2) Calculate the single-electron wavefunctions $\psi_i(\mathbf{r})$ by solving the Kohn-Scham equations with the initial defined electron density $\rho(\mathbf{r})$ from step (1).

(3) With the single-electron wavefunctions from step (2) calculate a new electron density $\rho(\mathbf{r})$ by using equation (3.22).

(4) Proceed with step (2) and step (3) as long as the difference between the electron density from step (2) and the electron density from step (3) is greater than a defined tolerance.

3.4 Exchange–correlation functional

The exchange-correlation functional $E_{XC}[\rho]$ describes the interaction and correlation of the electrons in a many-body system. There exist several approximations for $E_{XC}[\rho]$, which differ in terms of accuracy and computational cost ("Jacob's Ladder of Density Functional Theory" [59]). In this section, we will discuss some of the most common functionals used in practical DFT calculations.

3.4.1 Local density approximation LDA

In the local density approximation (LDA) the exchange-correlation functional $E_{XC}[\rho]$ depends only on the local electronic density $\rho(\mathbf{r})$ and was first introduced by Kohn and Scham themselves [12]. LDA works exactly for a homogenous electron gas (HEG) distribution and gives quite accurate results for some classes of solids [60], although it generally tends to overestimate binding energies. The exchange-correlation functional in LDA is given by:

$$E_{XC}^{\mathrm{LDA}}[\rho] = \int \rho(\mathbf{r})\epsilon_{XC}(\rho)d^3\mathbf{r} \tag{3.27}$$

ϵ_{XC} denotes the exchange-correlation energy per particle. While the computational expenses of LDA are comparatively modest, LDA is not appropriate for determining the band structure of semiconductors or insulators (as SiO_2) due to improper calculations [58].

3.4.2 Generalized gradient approximation GGA

The generalized gradient approximation (GGA) is obtained by including a dependency on the spatial variation of the electron density in the form of the electron density gradient $\nabla\rho$:

$$E_{XC}^{\mathrm{GGA}}[\rho] = \int \rho(\mathbf{r})\epsilon_{XC}(\rho, \nabla\rho)d^3\mathbf{r} \tag{3.28}$$

There exists a variety of different types of GGA, which can be categorized into one of the following two groups: empirical and non-empirical. While the empirical GGAs may include fitted parameters that have been changed by comparison with the experimental data, the non-empirical GGAs are derived straight from basic principles

and are not adapted to experimental data.

In this thesis and for the study on SiO_2, the non-empirical, semilocal GGA functional PBE was used. PBE is the acronym for Perdew, Burke, and Ernzerhof [61] and is widely used for solid-state calculations. The non-empirical parameters in PBE are designed to meet several known requirements for the precise functional and perform equally well for finite and infinite systems. Similar to LDA, GGA tends to underestimate the electronic bandgap [62].

3.4.3 Meta-GGA

The meta-GGA functionals contain in addition to the electron density $\rho(\mathbf{r})$ and the gradient of the electron density, $\nabla\rho(\mathbf{r})$, also information on the Laplacian of the electron density $\Delta\rho(\mathbf{r}) = \nabla^2\rho(\mathbf{r})$. Because the kinetic energy density of the Kohn-Sham orbitals, $\tau(\mathbf{r})$, carries the same physical information as the Laplacian of the electron density. Furthermore, due to a more precise information about the electronic structure, $\tau(\mathbf{r})$ rather than $\nabla^2\rho(\mathbf{r})$ is used in meta-GGA functionals [11]. The exchange-correlation functional can be read as:

$$E_{XC}^{\text{meta}-\text{GGA}} = \int \rho(\mathbf{r})\epsilon_{XC}(\rho, \nabla\rho, \tau)d^3\mathbf{r} \tag{3.29}$$

with the kinetic energy density of the Kohn-Sham orbitals $\tau(\mathbf{r})$:

$$\tau(\mathbf{r}) = \frac{1}{2}\sum_i |\nabla\rho(\mathbf{r})|^2 \tag{3.30}$$

whereas the sum runs over all occupied states i.

3.4.4 Hybrid-GGA

In addition to the GGA functional, the Hybrid-GGA contains also contributions of the exact exchange energy. The fundamental concept of hybrid-GGA is to use for one part of the exchange interaction LDA, GGA, or meta-GGA, and for the remaining part the Hartree-Fock method.

Becke suggested 1993 a hybrid exchange-correlation functional, where the correlation energy is treated entirely semilocal. Within this approach, the exchange energy is a linear combination of a semilocal exchange functional and the Hartree-Fock exchange functional [63]:

$$E_{XC}^{\text{hybrid}} = \alpha_X E_X^{\text{HF}} + (1-\alpha_X)E_X^{(\text{meta})-\text{GGA}} + E_C^{(\text{meta})-\text{GGA}} \tag{3.31}$$

with E_X^{HF} as the Hartree-Fock exchange functional and α_X as a parameter with $0 \leq \alpha_X \leq 1$. Two famous examples of hybrid-GGA functionals are the semiempirical B3LYP (acronym for Becke, three-parameter, Lee-Yang-Parr) functional [63, 64] and the PBE0 functional [65]. For instance, the PBE0 is given by:

$$E_{XC}^{\mathrm{PBE0}} = \frac{1}{4}E_X^{\mathrm{HF}} + \frac{3}{4}E_X^{\mathrm{PBE}} + E_C^{\mathrm{PBE}} \tag{3.32}$$

whereas 1/4 is the default value for α_X.

For the electrical structure (band gap) of insulators, semiconductors, and thermochemistry (atomization energy of molecules), hybrid functionals are more accurate than LDA, GGA, and meta-GGA. They have been shown to correctly (within 10% of the experimental value) predict the electronic structure of a variety of semiconductors and insulators [66]. However, compared to semilocal approaches, the calculations are significantly more costly. Furthermore, hybrid functionals give far too high magnetic moments for itinerant ferromagnets and are inaccurate for metals.

For strongly correlated systems there exist some even more advanced exchange-correlation functionals like the random phase approximation (RPA) [67] and the Hubbard-Corrected DFT energy functional DFT+U [68] which will not be discussed here further.

3.5 Basis sets

To solve the Kohn-Scham equations (3.23), basis sets are used to approximate the electron wavefunctions ψ_i of a system. Basis sets consist of a series of functions, φ_j, that represent the electron orbitals and transform the model's partial differential equations into algebraic equations that can be effectively solved by a computer. For efficient calculations, it is essential that the basis set is compact and at the same time accurate enough. With the basis functions φ_j of a chosen basis set, the electron wavefunctions ψ_i can be expressed as a linear combination in the form:

$$\psi_i = \sum_j c_{ij}\varphi_j \tag{3.33}$$

Here, c_{ij} denotes the coefficient of a basis function φ_j.

3.5.1 Double-ζ Gaussian basis set

All calculations in this work are carried out using a double-ζ Gaussian basis set. This basis set is an advancement of Slater-type atomic orbitals (STO) [69], with the radial term of STOs given by:

$$R_n^{\mathrm{STO}}(r) = N_n r^{n-1} e^{-\zeta r} \tag{3.34}$$

Here, $n \in \mathbb{N}$ stands for the principal quantum number and N is a normalizing constant. The distance between the electron and the atomic nucleus is represented by r, whereas the constant ζ is associated with the nucleus's effective charge.

By using two Slater-type orbitals with different ζ values for a single orbital, the atomic orbital can be described more accurately due to the increased flexibility. This leads to the double-ζ basis set:

$$R_{nl}^{\mathrm{STO}}(r) = C_1 r e^{-\zeta_1 r} + C_2 r e^{-\zeta_2 r} \tag{3.35}$$

Charge is accounted for near the nucleus by the function with a big ζ, and at increasing distances from the nucleus by the function with a smaller ζ.

However, STOs are computationally inefficient. Frank Boys subsequently discovered that these STOs may also be roughly represented as linear combinations of Gaussian-type orbitals (GTO) [70] by replacing each STO with several Gaussian functions with different values for the exponential parameter:

$$R_{n}^{\mathrm{GTO}}(r) = \sum_i c_{ni} e^{-\alpha_{ni} r^2} \tag{3.36}$$

Here, c_{ni} and α_{ni} are fitting parameter. This results in a significant reduction of computational costs since Gaussian basis functions make it simpler to compute overlap and other integrals. The Gaussian basis functions are given then by [71]

$$\varphi_{n,l,m}^{\mathrm{GTO}}(r,\theta,\phi) = r^{n-1} R_n(r) Y_{lm}(\theta,\phi) \tag{3.37}$$

where $Y_{lm}(\theta,\phi)$ are the normalized spherical harmonics, θ is the azimuth angle and ϕ is the polar angle. Each atomic orbital can then be described using the double zeta basis function:

$$\varphi_i = a_1 \varphi_{nl}^{\mathrm{GTO}}(r,\zeta_1) + a_2 \varphi_{nl}^{\mathrm{GTO}}(r,\zeta_2) \tag{3.38}$$

with a_1, a_2, ζ_1 and ζ_2 as fitting parameter.

3.6 CP2K software package

CP2K is an open-source software package for quantum chemistry and solid state physics [49]. For DFT calculations, CP2K is supported by QUICKSTEP [72], a freely available computer code, which allows accurate and efficient DFT computations.

All DFT data for the training and testing of machine learning force fields (MLFF) in this thesis were performed with CP2K by using single-point (energy) calculations, geometry optimizations, and *ab-initio* molecular dynamics (AIMD). In the following, some basic concepts of CP2K and the setting of the DFT calculation will be described.

3.6.1 Gaussian and plane wave (GPW) method

CP2K uses the Gaussian and plane waves (GPW) method, which is implemented in QUICKSTEP, to solve the Kohn-Sham equations efficiently. The GPW method uses two representations of the electron density, $\rho(\mathbf{r})$ and $\tilde{\rho}(\mathbf{r})$, and combines an atom-centered Gaussian-type basis and an additional plane wave basis [73].

The first representation of the electron density, $\rho(\mathbf{r})$, is used to represent the wave-functions and is given by an expansion in atom-centered, contracted Gaussian functions [72] :

$$\rho(\mathbf{r}) = \sum_{\mu,\nu} P_{\mu\nu}\varphi_\mu(\mathbf{r})\varphi_\nu(\mathbf{r}) \tag{3.39}$$

Here, $P_{\mu\nu}$ denotes an element of the density matrix $\underline{\mathbf{P}}$ and $\varphi_\mu(\mathbf{r})$ denotes the contracted Gaussian functions given by:

$$\varphi_\mu(\mathbf{r}) = \sum_i d_{i\mu} g_i(\mathbf{r}) \tag{3.40}$$

with the corresponding contraction coefficients $d_{i\mu}$ and the primitive Gaussian functions $g(\mathbf{r})$ [49]:

$$g(\mathbf{r}) = r^l \exp\left[-\alpha(\mathbf{r}-\mathbf{r}_0)^2\right] Y_{lm}(\mathbf{r}-\mathbf{r}_0) \tag{3.41}$$

The primitive Gaussian functions $g(\mathbf{r})$ are centered at atomic positions $\mathbf{r}_0$. These functions are determined by the exponent α, the coordinates of its center $\mathbf{r}_0$, and the spherical harmonics Y_{lm} with angular momentum (l, m).

The second representation of the electronic density, $\tilde{\rho}(\mathbf{r})$, is used to represent the actual electronic density and is approximated by an auxiliary plane waves basis set:

$$\tilde{\rho}(\mathbf{r}) = \frac{1}{\Omega}\sum_{\mathbf{G}} \tilde{\rho}(\mathbf{G})\exp(i\mathbf{G}\cdot\mathbf{r}) \tag{3.42}$$

Ω denotes the volume of the unit cell and $\mathbf{G}$ denotes the reciprocal lattice vectors of the unit cell. $\tilde{\rho}(\mathbf{G})$ denotes the expansion coefficients, with which $\tilde{\rho}(\mathbf{r})$ matches $\rho(\mathbf{r})$ for a regular grid in the unit cell [72].

Within the GPW approach, the Kohn-Scham energy functional $E[\rho]$ is given by

$$E[\rho] = T_e[\rho] + V_{en}[\rho] + E_H[\rho] + E_{XC}[\rho] + E_{II}[\rho] \tag{3.43}$$

$T_e[\rho]$ is the electronic kinetic energy and $V_{en}[\rho]$ describes the electronic interaction with the ionic cores. $E_H[\rho]$ is the electronic Hartree energy and describes the total electrostatic (Coulomb) energy, which is usually expressed via a pseudopotential (see section 3.6.2). $E_{XC}[\rho]$ is the exchange-correlation energy and $E_{II}[\rho]$ the interaction energies of the ionic cores.

3.6.2 Pseudopotentials

Many chemical processes, such as the formation and breaking of bonds, only require a precise representation of the valence electrons. Therefore the core electrons can be approximated by a pseudopotential V^{PP} since the expansion of an all-electron density is computationally very expensive. In the GPW approach, the well-established Goedecker-Teter-Hutter (GTH) pseudopotentials are used to represent these closed-shell electrons [74, 75].

GTH seudopotential

The GTH pseudopotentials are given by an analytical form and can be separated into two parts: First, in a local part $V_{loc}^{\mathrm{PP}}(\mathbf{r})$, which consists of a short-ranged (SR) term $V_{loc}^{\mathrm{SR}}(\mathbf{r})$ and a long-ranged (LR) term $V_{loc}^{\mathrm{LR}}(\mathbf{r})$. Additionally, in a non-local part $V_{nl}^{\mathrm{PP}}(\mathbf{r},\mathbf{r}')$. The GTH pseudopotential V^{PP} is then given by:

$$V^{\mathrm{PP}} = V_{loc}^{\mathrm{PP}}(\mathbf{r}) + V_{nl}^{\mathrm{PP}}(\mathbf{r},\mathbf{r}') = V_{loc}^{\mathrm{SR}}(\mathbf{r}) + V_{loc}^{\mathrm{LR}}(\mathbf{r}) + V_{nl}^{\mathrm{PP}}(\mathbf{r},\mathbf{r}') \tag{3.44}$$

The local part $V_{loc}^{\mathrm{PP}}(\mathbf{r})$ is defined as [76]:

$$\begin{aligned} V_{loc}^{\mathrm{PP}}(\mathbf{r}) = & \frac{-Z_{ion}}{\mathbf{r}} erf\left(\frac{\mathbf{r}}{\sqrt{2}r_{loc}}\right) + exp\left[-\frac{1}{2}\left(\frac{\mathbf{r}}{r_{loc}}\right)^2\right] \\ & \times\left[C_1 + C_2\left(\frac{\mathbf{r}}{r_{loc}}\right)^2 + C_3\left(\frac{\mathbf{r}}{r_{loc}}\right)^4 + C_4\left(\frac{\mathbf{r}}{r_{loc}}\right)^6\right] \end{aligned} \tag{3.45}$$

Z_{ion} denotes an ionic charge (e.g. the difference between the charge of the core electrons and the charge of the nucleus), erf denotes the error function, and r_{loc} denotes a characteristic distance at which the pseudopotential acts. For distances r $> r_{loc}$ the pseudopotential begins to behave like the corresponding Coulomb potential. In QUICKSTEP, the short-range terms are computed as two- and three-center overlap integrals, while the long-range term is handled as part of the electrostatic energy [72].

It is necessary to optimize the GTH pseudopotential for the exchange-correlation functional that is being used. For QUICKSTEP, improved GTH pseudopotential parameters based on LDA are available [49]. Furthermore, the parameters for the common elements have been optimized also for the GGA exchange-correlation potentials of Becke and Perdew (BP) [77, 78], Perdew, Burke and Ernzerhof (PBE) [61], Becke, Lee, Yang, and Parr (BLYP) [77, 79, 80], and Hamprecht, Cohen, Tozer and Handy (HCTH/120, HCTH/407) [81].

3.6.3 Cutoff energy

To describe the behavior of the inner electrons, a huge number of basis functions would be needed because their wavefunctions fluctuate quickly in space. To keep

the computing costs within reasonable limits, the kinetic energy (given in Ry) of the plane wave with the highest plane-wave vector $\mathbf{G}$ has to be limited by a cutoff energy E_{cutoff} [72]:

$$\frac{1}{2}|\mathbf{G}|^2 \leq E_{\text{cutoff}} \tag{3.46}$$

E_{cutoff} therefore restricts the resolvable spatial frequency. When using Gaussian basis sets, the required cutoff is directly tied to the largest exponent. The Gaussian basis sets of different elements show that the value of the largest exponent rapidly increases with the atomic number. To address this challenge, the core electrons have to be approximated by a pseudopotential as described in the previous chapter.

3.6.4 Simulation settings

While CP2K provides a general framework for a variety of methods, this thesis focuses on single-point calculations, cell and geometry optimizations, and *ab-initio* molecular dynamics (AIMD). CP2K uses atomic units, so the energy is given in units of Hartree with 1 Hartree = 27.211 eV, and the forces in units of Hartree/a_0 with $a_0 = 0.52918$ Å as the Bohr radius. The analytical stress components are given in units of GPa. Each CP2K calculation requires one main input file to provide CP2K with the necessary system information and parameters. This file consists of keywords in ordered blocks. CP2K also requires two additional files. One file which contains the parameters for the selected basis set. And a second file containing the parameters for the selected GTH pseudopotential.

Following is a brief description of some important sections of the main input file:

- The section GLOBAL contains among other settings also the type of run, e.g. ENERGY_FORCE for single-point calculations, GEO_OPT or CELL_OPT for optimizations, and MD for *ab-initio* molecular dynamics.
- In the section FORCE_EVAL the method for evaluating the forces can be set. With the setting of the method to QUICKSTEP, CP2K uses the Gaussian and Planewaves method (section 3.6.1) for DFT calculations.
- Within the subsection SUBSYS the simulation unit cell and the initial coordinates of the atoms can be set.
- In the subsection KIND the basis set (e.g. double-ζ Gaussian basis, section 3.5.1) and the pseudopotential (e.g. GTH pseudopotential, section 3.6.2) for each element can be defined.
- The simulation unit cell used in the calculation is defined in the subsection CELL.

- In the subsection TOPOLOGY the initial atomic coordinates can be specified. The coordinates can also provided via an additional input file in the XYZ format.
- The subsection DFT contains all information for the self-consistent Kohn-Sham DFT calculation (section 3.3.1).
- The parameters for QUICKSTEP can be set in the subsection QS.
- QUICKSTEP uses a multi-grid representation of the Gaussian functions. The parameters of these multi-grids can be specified in the subsection MGRID. Next to the number of levels also the cutoff energy (section 3.6.3) of the Gaussian basis set can be set in this subsection.
- The exchange-correlation density functional (section 3.4) can be defined in the XC subsection.
- In the subsection SCF (self-consistent field) all parameters for the self-consistent solution of the Kohn-Sham equations (section 3.3.1) can be specified. The initial trial electron density function $\rho(\mathbf{r})$ can be set via SCF_GUESS and the maximum number of self-consistency loops for QUICKSTEP is set with MAX_SCF. Convergence criteria for the SCF calculations, such as energy and density matrix convergence thresholds, are specified to ensure the iterative solution achieves the desired accuracy before terminating the calculation.
- There are two options for CP2K finding the ground state Kohn-Sham energy and the electron density: The first option uses the traditional diagonalization method, where the parameter can be set in the subsection DIAGONALIZATION. The alternative to this method is the Orbital Transform (OT) method, which can be specified in the subsection OT.
- The subsection PRINT tells CP2K the properties that should written out in the output files.

All DFT calculations carried out for this thesis had the following CP2K setting: For all atoms, a double-ζ Gaussian basis set was used as the primary basis for the GPW method and the core electrons were represented by using the GTH pseudopotential. For the description of the exchange-correlation energy, the semilocal functional PBE as decribed in section 3.4.2 was employed. For the plane wave expansion of the electron density, an energy cutoff E_{cutoff} of 650 Ry was used.

Single-point calculations

Single-point calculations are static self-consistent Kohn-Sham calculations and are used to compute the total energy, the forces on each atom, and the analytical stress

components of the provided system. Single point energies are the lowest energy solution for the Schrödinger equation and the single-point calculations gives the energy of the ground state of the system [61].

Cell and geometry optimizations

There are two optimization schemes used in this thesis: Cell optimization and geometry optimization. In geometry optimization, the positions of the atoms in the system are changed iteratively in order to minimize the total energy. This is done by adjusting the atomic coordinates in order to find local or global minima of the energy. The optimization is continued until the change in total energy and the changes in atomic positions between iterations are smaller than specified threshold values. Cell optimization is the process of optimizing the lattice parameters (i.e. the dimensions and angles) of a crystal or periodic system to minimize the total energy of the system.

In CP2K, for both optimization schemes, the movement of the atoms is based on the computed forces of the previous step. The accuracy of the computed forces is determined by the convergence criteria of the self-consistent field (SCF) and a well-chosen cutoff energy.

The optimization calculations in CP2K were also used to calculate the total energy, the forces on each atom, and the analytical stress components to use as training data of the ML potential.

Ab-initio molecular dynamics (AIMD)

CP2K can also perform molecular dynamics calculations with DFT accuracy, called *ab-initio* molecular dynamics. Compared to classical molecular dynamics (MD), which is described in chapter 4, AIMDs have a much higher computational cost. In contrast to classical MD, which uses force fields (FF) to calculate the forces on each atom, AIMD computes the forces directly from electronic structure computations and uses them to generate the trajectory of the system. The forces on each atom and the total energy of the system were recorded in the output file to be used as input data for training the ML potential.

4 Molecular dynamics

Molecular Dynamics (MD) is a powerful computational technique used to determine and simulate the dynamic behavior of molecular systems in time. MD simulations offer important insights into the structure, thermodynamics, and kinetics of a wide range of systems in physics [82] and nanotechnology [83], by numerically solving the classical equations of motion for a set of interacting particles.
At its core, MD relies on the principles of statistical mechanics and Newtonian physics. It allows researchers to explore the time evolution of a molecular ensemble under the influence of intermolecular forces, providing a microscopic understanding of phenomena that are often challenging to investigate experimentally [84].

In the upcoming sections, the basics of MD will be explained and also the MD software Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS), with which all MD simulations in this thesis were carried out, will be described in more detail.

4.1 Concepts of MD

In contrast to quantum mechanics, molecular dynamics description uses particles to represent individual atoms or groups of atoms [85]. The two fundamental components that govern molecular dynamics simulations are (i) the equations of motion and (ii) the interatomic potential (i.e., potential energy) of the particles, from which then the forces can be calculated [86].

4.1.1 Equations of motion

Starting with classical mechanics, the force on an atom i in a system with N interacting atoms is given by

$$\mathbf{F}_i = m_i \mathbf{a}_i = m_i \frac{d\mathbf{v}_i}{dt} = m_i \frac{d^2\mathbf{r}_i}{dt^2} \tag{4.1}$$

Here, $\mathbf{a}_i$ denotes the acceleration of the atom i, $\mathbf{v}_i$ it's velocity, $\mathbf{r}_i$ it's position and m_i it's mass. The force on the atom i can also be expressed as the negative gradient of a potential V, which is also the negative derivative of the potential energy E regarding

the change in the atom's position $\mathbf{r}_i$:

$$\mathbf{F}_i = -\nabla_i V = -\frac{dE}{d\mathbf{r}_i} \tag{4.2}$$

The potential energy E of a system with N atoms with positions $\mathbf{r}_i$, is defined as

$$E = E(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_{N-1}, \mathbf{r}_N) \tag{4.3}$$

In this context, the right-hand side of equation (4.3) is also referred to as the system's potential energy surface (PES), a scalar field with a dimension of $3N$. The PES describes the distribution of the potential energy E over the space of possible conformations of the system and indicates how the potential energy changes when the particles change their positions. If the exact shape of the PES is known, the movement of the particles along the PES can be predicted by calculating the forces on each particle with equation (4.2). However, approximations are necessary since it is impossible to determine the precise form of the PES in complex systems. To approximate the PES one can make use of classical potentials, such as the Lennard-Jones potential or ReaxFF, which will be discussed in the following section.

4.1.2 Classical Potentials and Force Fields

An effective way to approximate the Born-Oppenheimer PES computationally is through the use of classical potentials or force fields (FF). The basis of classical potentials and FFs is a set of empirical energy functions, in which the parameters of these functions are based on experimental data. This enables the computation of the potential energy V of a system of particles as a function of the molecular coordinates. Two important representatives are the Lennard-Jones Potential and ReaxFF.

Lennard-Jones potential

The Lennard-Jones (LJ) potential is a widely used classical potential in MD simulations to model the van der Waals interactions between atoms or molecules. It is named after the physicist John Lennard-Jones, who first proposed it to describe the attraction and repulsion between neutral atoms [87]. The LJ potential between an atom i and an atom j has the following form:

$$V_{LJ}(r_{ij}) = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] \tag{4.4}$$

The distance between atom i and atom j is r_{ij}, ϵ is the potential energy well depth, and σ the distance at which $V_{LJ} = 0$.

The form of the LJ potential, which is graphically shown in Fig. 4.1, consists of 2 terms:

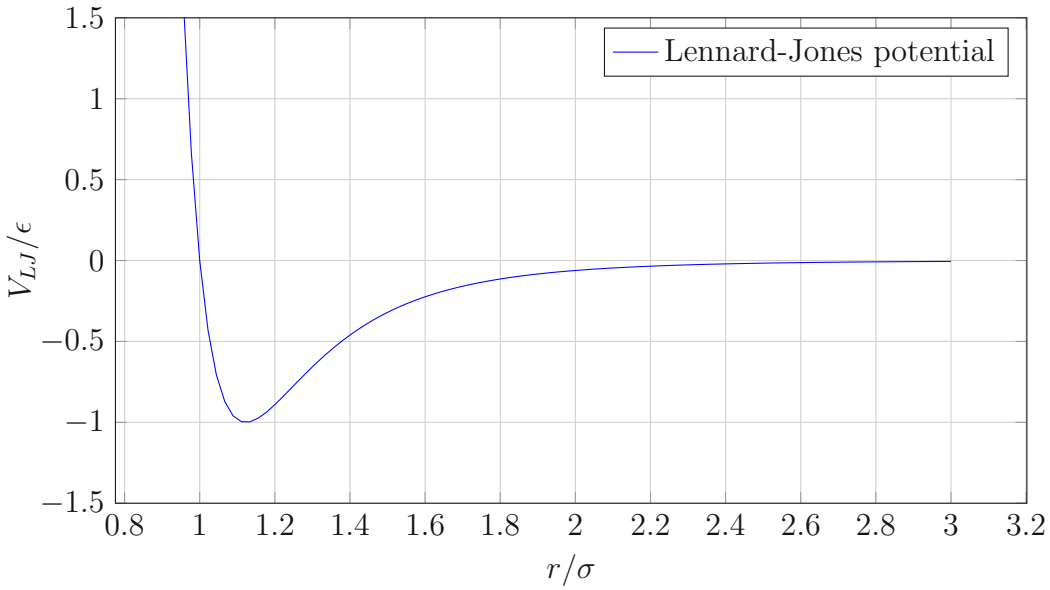


Figure 4.1: The Lennard-Jones potential V_{LJ} plotted against the particel distance r/σ. Repulsive forces act in the area of negative gradient, and attractive forces in the area of positive gradient.

1. The lefthanded term with the exponent 12 represents the Pauli repulsion between the electron shells of the atoms, which becomes dominant at very small distances. It causes a rapid increase in potential energy at small distances.

2. The righthanded term with the exponent 6 describes the Van der Waals attraction (dipole-dipole interactions due to fluctuating dipoles) which becomes dominant at larger distances.

Despite being a simplified model, the LJ potential captures the key elements of interactions between basic atoms and molecules: When two particles interact, they repel one another when they are extremely near, attract one another when they are somewhat apart, and do not interact when they are infinitely apart. The LJ potential only considers interactions between 2 particles, it does not account for interactions between three or more bodies.

ReaxFF

ReaxFF [88], which is the acronym for reactive force field, is a bond order-based interatomic potential that uses bond orders rather than fixed bond lists to enable continuous bond formation and/or breaking. ReaxFF links bond distance and bond order on the one hand, and bond order and bond energy on the other.

To describe the behavior of each atom within a system, one can determine the forces acting on each atom by deriving them from the following energy expression [89]:

$$E_{\text{System}} = E_{\text{bond}} + E_{\text{over}} + E_{\text{under}} + E_{\text{lp}} + E_{\text{val}} + E_{\text{tor}} + E_{\text{vdWaals}} + E_{\text{Coulomb}} \quad (4.5)$$

The individual energy terms denote bond energy, over-coordination penalty energy, under-coordination stability energy, lone-pair energy, valence angle energy, torsion energy, van der Waals energy, and Coulombic energy. A schematic illustration of how each computational iteration is carried out throughout a MD simulation employing ReaxFF is shown in Fig 4.2.

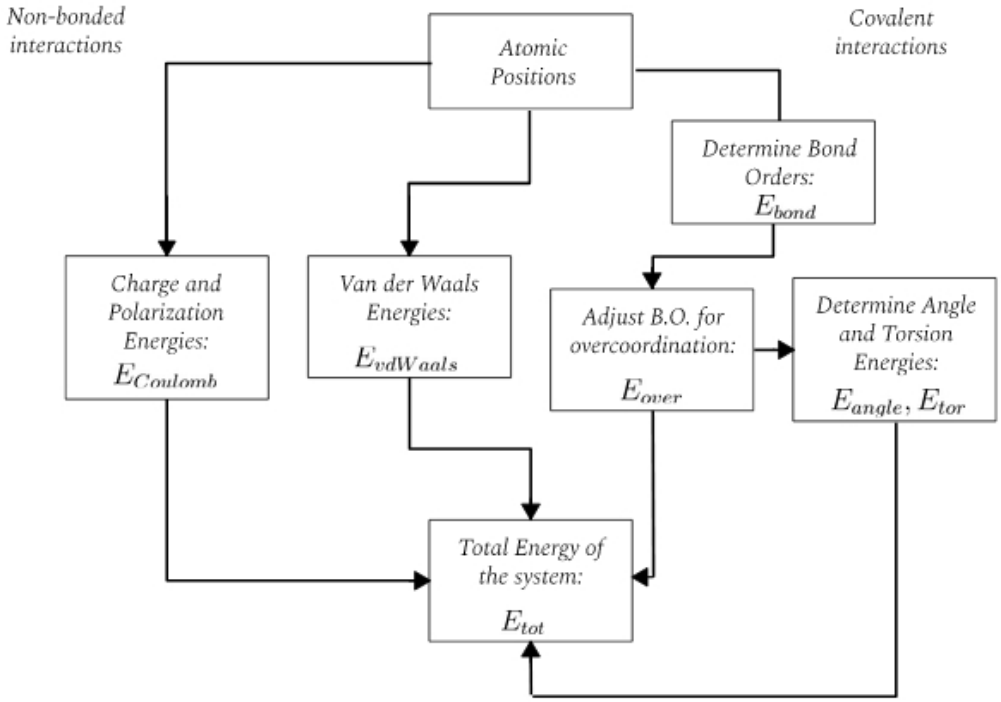


Figure 4.2: Schematic illustration of the ReaxFF simulation steps. The non-bonded interactions are on the left-hand side, the covalent and bonded interactions are on the right-hand side.

After recording each atom's position, the initial step is finding the bond order BO_{ij} between each pair of atoms to determine E_{bond}. A correction term E_{over} must be included in this bond to ensure accurate modeling. E_{angle} and E_{tors} are terms that result from the breaking of an atom bond; the bond order decreases and the angle and torsion forces acting on the atom decrease relative to the remaining atoms.

The non-bonded interactions consist of the terms E_{Coulomb} and E_{vdWaals}. ReaxFF is capable of determining how charges inside atom configurations are polarized by using

the electronegativity and hardness parameters of each element. The polarization can be calculated as followed [89]:

$$\frac{\partial E}{\partial q_n} = \chi_n + 2q_n\eta_n + C\sum_{j=1}^{n}\frac{q_j}{\{r_{n,j}^3 + (1/\gamma_{n,j}^3\}^{1/3}}, \qquad \sum_{i=1}^{n} q_i = 0 \tag{4.6}$$

Here, q_n denotes the charge of an element n and q_j the charge of an element j, χ_n denotes the electronegativity, and η_n the hardness of element n, constant C denotes a conversion factor, $r_{n,j}$ denotes the interatomic distance and $\gamma_{n,j}$ is the shielding parameter between atom n and j. During MD simulation, the charge values are computed for each time step.

Both the Coulomb interactions E_{Coulomb} and van der Waals interactions E_{vdWaals} are shielded in ReaxFF by using a shielding term γ to prevent too repulsive or attractive non-bonded interactions at short distances. E_{Coulomb} is given by :

$$E_{\text{Coulomb}} = C\left[\frac{q_i q_j}{\{r_{ij}^3 + (1/\gamma_{ij}^3\}^{1/3}}\right] \tag{4.7}$$

with γ_{ij} as shielding parameter between atom i and atom j and r_{ij} as interatomic distance. C denotes a possible conversion factor and q_i and q_j denotes the charges of atom i and atom j, respectively.

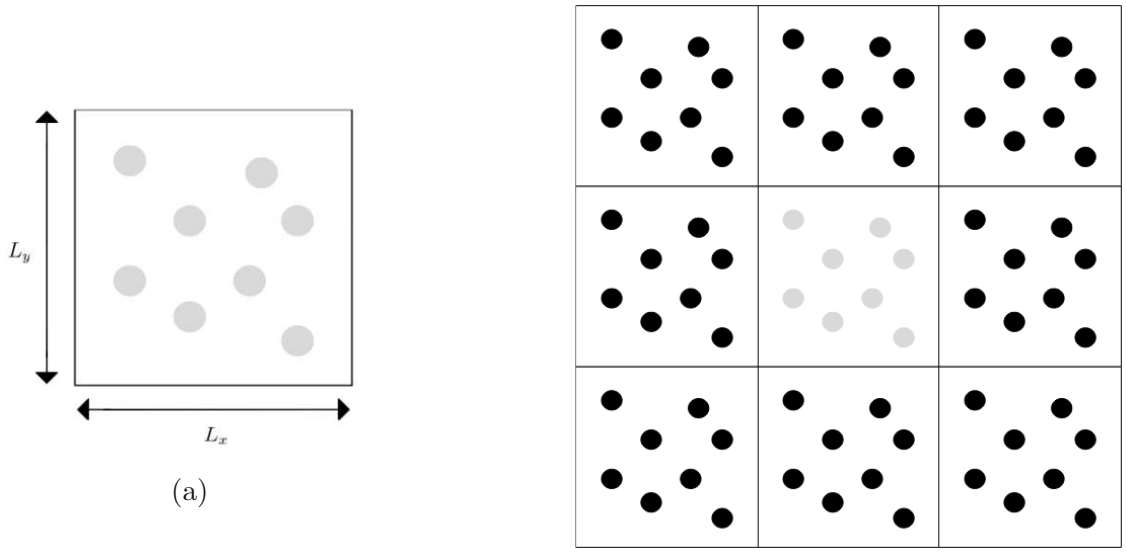
In this thesis, several MD simulations were carried out with ReaxFF. The results serve as a comparison to the MD simulations carried out with machine-learning force fields.

4.1.3 Boundary conditions

MD calculations are performed within predefined simulation boxes, whereas boundary conditions (BC) play an important role in treating various atom configurations differently and also in reducing computational costs. The most common BCs for MD simulations are the periodic boundary conditions (PBC) and the fixed boundary conditions (FBC), which are not suitable for non-equilibrium situations. In this case, for example, the Lees-Edwards boundary conditions [90] are needed.

PBC

PBCs are a collection of boundary conditions that are frequently used to approximate a vast (infinite) system by using just a small component of the system by repeating it numerous times in each direction of consideration. Therefore it is possible to describe systems that are spatially homogenous concerning their boundaries [91].



(a) (b)

Figure 4.3: MD simulation box and the PBC. Panel (a) shows a two-dimensional simulation box with atoms in it. Panel (b) indicates the replication of the simulation box in the x- and y-direction for the PBC

Fig. 4.3 illustrates this for the two-dimensional case: The left picture (Fig. 4.3a) shows the simulation box with size L_x and L_y, whereas the gray dots represent atoms. With PBCs, this simulation box and also all the atoms within will be replicated in every direction, in our case in the x-direction and the y-direction. This is indicated in Fig. 4.3b. If an atom i has the coordinates (x_i, y_i), then the replicas of atom i have the coordinates $(x_i \pm nL_x, y_i \pm nL_y)$ with $n \in \mathbb{Z}$ ranging from $-\infty$ to $+\infty$.

PBCs are used together with the so-called minimum image convention. That means that each particle in the simulation interacts with the closest image of the other particles in the system. When integrating the equations of motion (equation (4.1)), the wraparound impact of the periodic boundaries must be considered. This effect must also be accounted for when calculating the interactions. PBCs are typically used for simulations of bulk systems.

FBC

In contrast to PBC, in FBC the boundary conditions are fixed. This indicates that the simulation box is in the considered directions non-periodic, meaning that particles do not travel from one side of the box to the other or interact across the boundary. In this thesis, FBCs were mainly used for the oxidation of silicon surfaces to fix the boundary conditions in the z-direction.

4.1.4 Ensembles

In MD simulations often only the average properties of the individual particles in a system are of interest. To describe such thermodynamic systems, J. Willard Gibbs introduced the concept of an ensemble in 1902 [92]. An ensemble is a set of similarly prepared systems of particles that are in thermodynamic equilibrium. Gibbs defined three ensembles [93], which are illustrated in Fig. 4.4:

- **Microcanonical (NVE) ensemble**: In the microcanonical ensemble, the total number of particles N, the volume V, and the total energy of the system E are assumed to be constant. To maintain statistical equilibrium, the system needs to be completely isolated and not exchange any particles or energy with its surroundings. Microcanonical ensembles are used to describe closed systems that do not interact with their environment, i.e. do not exchange energy and/or matter with their surroundings.
- **Canonical (NVT) ensemble**: A set of microstates with fixed total number of particles N, constant volume V, fixed temperature T but variable total energy E is called a canonical ensemble. The system can form weak thermal contact with other systems but it must stay completely closed (unable to exchange particles with its surroundings) to maintain statistical equilibrium.
- **Grand canonical (μVT) ensemble**: The grand canonical ensemble describes a statistical ensemble, where neither the total energy E nor the particle number N are fixed. This ensemble is appropriate for describing an open system in a heat bath, where the chemical potential μ, the volume V, and the temperature T are fixed.

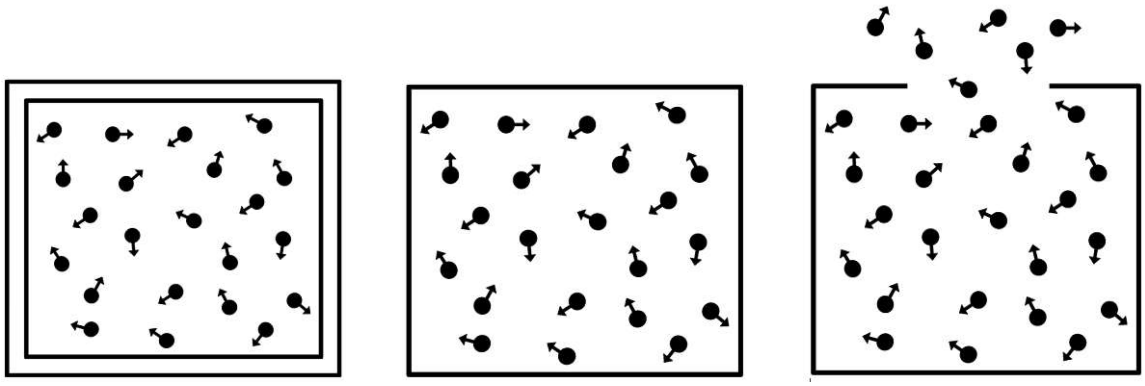
4.1.5 Thermostats

Thermostats in MD simulations are methods to control the temperature of particles by controlling their velocities. There are several thermostat techniques that add and remove energy from the boundaries of an MD system in a realistic way like the Berendsen Thermostat, the Langevin Thermostat, and the Nosé-Hoover Thermostat.

Berendsen thermostat

The Berendsen thermostat is used to simulate an MD system that is (weakly) coupled to an external bath with a desired temperature T_{set} [94]. In statistical mechanics, the time average of the kinetic energy $\langle K \rangle$ and the temperature T of an unconstrained MD system with N particles are related as follows:

$$\langle K \rangle = \frac{3}{2} N k_B T \tag{4.8}$$



(a) (b) (c)

Figure 4.4: Three different ensembles. Panel (a) shows the NVE (microcanonical) ensemble, where N, V, and E are constant. There is no particle or energy exchange with the surroundings, which is indicated by an isolated box. Panel (b) shows the NVT (canonical) ensemble with constant N, V and T. There is no particle exchange with the surroundings but heat exchange is allowed. Panel (c) shows the μVT (grand canonical) ensemble with fixed μ, V and T. This system can exchange heat and particles with the surroundings.

Here, k_B denotes the Boltzmann constant. The kinetic energy K of a system is also given by:

$$K = \frac{1}{2}\sum_i m_i v_i^2 \tag{4.9}$$

with m_i as the mass and v_i as the velocity of the ith atom. Therefore the temperature T of an MD system and the velocities v_i of the particles within the system are correlated.

The Berendsen Thermostat achieves control from a temperature T to a desired temperature T_0 by multiplying the velocities v_i with a scaling factor λ:

$$v_i^{\text{new}} = v_i^{\text{old}}\lambda \tag{4.10}$$

The velocities are scaled at every time step in an MD simulation in a way that the temperature difference is proportionate to the rate of temperature change [94]:

$$\frac{dT}{dt} = \frac{1}{\tau}(T_0 - T) \tag{4.11}$$

Here, τ denotes a time constant, which is defined:

$$\tau = 2\frac{C_V \tau_T}{N_f k_B} \tag{4.12}$$

C_V denotes the constant volume heat capacity of the MD system, N_f represents the number of degrees of freedom of the MD system, and τ_T denotes the coupling strength between the heat bath and the MD system.

The solution of equation (4.11) describes an exponential decay of the system towards T_0:

$$T = T_0 - Ce^{-t/T} \tag{4.13}$$

where C denotes a constant factor and t the time. The scaling factor λ in the Berendsen approach is given by

$$\lambda^2 = 1 + \frac{\Delta t}{\tau}\left(\frac{T_0}{T} - 1\right) \tag{4.14}$$

The Berendsen thermostat provides provides good approximate results for the majority of the calculated properties for large systems, even though the thermostat does not provide a correct canonical (NVT) ensemble, especially for small systems [95]. For MD simulations, the Berendsen thermostat is a very effective way of relaxing the MD system to a target temperature T_0 after an energy minimization step.

Langevin thermostat

For the simulation of a canonical NVT ensemble, the Langevin thermostat can be used to control the temperature of a system. Essentially, the idea is to sample the statistical ensemble by supposing that the dynamics of each particle in the system is governed by the Langevin equation [96]. The Langevin equation of each particle in the system is given by [97]:

$$m\ddot{r} = f - \alpha v + \beta(t) \tag{4.15}$$

The position and the velocity of the particles are denoted by r and $v = \dot{r}$, respectively. f denotes a force, and α denotes a friction coefficient. β denotes a Gaussian white noise with zero mean ($\langle\beta(t)\rangle = 0$) and its delta-function auto-correlation [98] is given by:

$$\langle\beta(t)\beta(t')\rangle = 2k_B T\alpha\delta(t - t') \tag{4.16}$$

Here, t and t' represent 2 different times, k_B represents the Boltzmann's constant and T the temperature. Equation (4.15) describes the motion of a particle which is under the influence of three different forces: a deterministic force f, a friction force $-\alpha v$ and a stochastic random force β.

The Langevin thermostat is now obtained by integrating the equation (4.15) over a discrete time interval Δt, whereas Δt must be small enough to ensure an exact solution for the integral. Due to the complex nature of solving the integral, a broad variety of algorithms exist to overcome this problem [99].

Nosé-Hoover thermostat

The Nosé-Hoover thermostat is based on an extended Lagrangian method by including an additional degree of freedom in the Hamiltonian operator [100]. In terms of the virtual variables, the Hamiltonian of the extended system H_{NH} with N particles and variable s for the additional degree of freedom is given by [101]

$$H_{\mathrm{NH}} = \sum_{i}^{N} \frac{\mathbf{p}_i^2}{2m_i s^2} + V(R_{ij}, P_{ij}) + \frac{p_s^2}{2Q} + g k_B T ln(s) \tag{4.17}$$

The first term is the sum of all kinetic energies of atoms i, which are represented by their momentum $\mathbf{p}_i$. $V(R_{ij}, P_{ij})$ denotes a potential which depends on both all positions R_{ij} and momenta P_{ij}. p_s denotes the momentum conjugate to s, Q is an effective mass associated with s and the parameter g represents the number of degrees of freedom of the system. Equation(4.17) leads to the following equations of motion:

$$\frac{d\mathbf{r}_i}{dt} = \frac{\partial H_{\mathrm{NH}}}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i s^2} \tag{4.18}$$

$$\frac{d\mathbf{p}_i}{dt} = -\frac{\partial H_{\mathrm{NH}}}{\partial \mathbf{q}_i} = -\frac{\partial V}{\partial \mathbf{q}_i} \tag{4.19}$$

$$\frac{ds}{dt} = \frac{\partial H_{\mathrm{NH}}}{\partial p_s} = \frac{p_s}{Q} \tag{4.20}$$

$$\frac{dp_s}{dt} = -\frac{\partial H_{\mathrm{NH}}}{\partial s} = \frac{\sum \frac{\mathbf{p}_i^2}{2m_i s^2} - g k_B T}{s} \tag{4.21}$$

The effective mass Q represents the coupling between the system and the heat bath, whereas a low coupling means a high Q. For an extended system the Nosé-Hoover method produces a microcanonical ensemble. In classical molecular dynamics, the Nosé-Hoover thermostat is currently one of the most popular thermostats for MD simulations [102].

4.2 MD simulations

To run MD simulation, one has to compute the equation of motion (4.1) and (4.2)). Finding a solution for these equations is not straightforward and several strategies exist to cope with this challenge, which will discussed in the following. There will also be a more thorough discussion of the MD simulation program LAMMPS.

4.2.1 Velocity Verlet algorithm

The equations of motion for a system with more than two atoms cannot be solved analytically, but they can be addressed numerically using the finite difference method.

The basic idea behind this method is an initial discretization of the equation of motion and then solving them by integrating over small time steps Δt to calculate the new positions $\mathbf{r}_i$ and the forces $\mathbf{F}_i$ acting on each atom i at the timestep $t + \Delta t$.

A widely used algorithm that uses the finite difference method is the Velocity Verlet algorithm [103]. This algorithm expresses the positions $\mathbf{r}_i$ and the velocities $\mathbf{v}_i$ at time $t+\Delta t$ of atom i as a Taylor series expansion of positions and velocities at time t. The Taylor series expansion of positions and velocities at time $t + \Delta t$ of the Velocity Verlet algorithm is given by

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \mathbf{a}_i(t)\frac{\Delta t^2}{2} + \mathcal{O}(\Delta t)^3 \tag{4.22}$$

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{1}{2}\Delta t[\mathbf{a}_i(t) + a(t + \Delta t)] \tag{4.23}$$

with $\dot{\mathbf{r}}_i = d\mathbf{r}_i/dt = \mathbf{v}_i$ and $\ddot{\mathbf{r}}_i = d^2\mathbf{r}_i/dt^2 = \mathbf{a}_i$. A MD simulation with the velocity Verlet algorithm is then carried out according to the following steps:

(1) Get the initial positions $\mathbf{r}_i(t = 0)$, velocities $\mathbf{v}_i(t = 0)$ and accelerations $\mathbf{a}_i(t = 0)$ from all atoms in the system

(2) Compute velocities $\mathbf{v}_i$ for $t = t + \Delta t/2$:

$$\mathbf{v}_i(t + \Delta t/2) = \mathbf{v}_i(t) + \frac{1}{2}\mathbf{a}_i(t)\Delta t \tag{4.24}$$

(3) Compute new positions $\mathbf{r}_i$ for $t = t + \Delta t/2$:

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \Delta t/2)\Delta t \tag{4.25}$$

(4) Compute new accelerations $\mathbf{a}_i$ for $t = t + \Delta t$ using the negative gradient of the potential V_i, which is a function over all atom positions r:

$$\mathbf{a}_i(t + \Delta t) = -\frac{1}{m_i}\nabla V_i(\mathbf{r}(t + \Delta t)) \tag{4.26}$$

(5) Get the new velocities $\mathbf{v}_i$ for $t = t + \Delta t$ with the following equation:

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t + \Delta t/2) + \frac{1}{2}\mathbf{a}_i(t + \Delta t)\Delta t \tag{4.27}$$

(6) Repeat steps (2) to (5) as long as necessary

To calculate the accelerations $\mathbf{a}_i$ for each atom i in step (4), it is assumed that the forces acting on the atomic nuclei remain constant during a time step Δt.

4.2.2 LAMMPS

LAMMPS is short for Large-scale Atomic/Molecular Massively Parallel Simulator [104] and is a powerful code for classical molecular dynamics simulations. This software is designed for parallel computation using MPI. The architecture of LAMMPS is based on a modular design which makes it simple to modify or add additional features. With the input script, which consists of a sequence of commands and parameters, MD simulations with different settings are possible. Tab. 4.1 lists and briefly describes frequently used commands of the MD simulation settings in this thesis.

4.2.3 MD simulation settings

Each MD simulation in this thesis is performed either with a QUIP potential or a ReaxFF potential.

Simulations with QUIP

A QUIP potential has been used to define *pair_style* in almost all MD simulations. The pair coefficients (via the command *pair_coeff*) are set by external XML files. These XML files and their values originate from a machine learning potential, which will be discussed in more detail in section 5.2. The *units* are set to *metal*, so the distances in the MD simulations are given in Å, the time in ps, the energy in eV, and the temperature in K. By setting *atom_style* to *atomic*, only the default values of each atom are getting stored or are getting read from an input file via the *read_data* command. The boundary conditions are set to *p p f* for MD simulations with *Si*-surfaces and to *p p p* for MD simulations with *Si*-nanowires. The parameters for the *neighbor* command are set to 0.3 *bin*, which creates a neighbor list with a skin distance of 0.3 Å by an *bin* algorithm. With the setting of *neigh_modify* to *delay* 10, the building of the neighbor list is delayed by 10 steps after the last build. For the employed ensemble either a *nve* ensemble or an *nvt* ensemble is assigned to the describe the system. The velocities for the atoms are set to an ensemble of generated velocities with a Gaussian distribution. These velocities are rescaled every timestep using a *Berendsen* or a *Langevin* thermostat. The timestep is usually set to 1 fs and the *thermo* command to 1000 timesteps, meaning that every 1000^{th} timestep the positions of the atoms are written out. The output information is written to one or more output files in XYZ format via the *dump* command. The main output file consists of information about the actual MD simulation, which is initialized by the *run* command. For subsequent geometry optimization via the *minimize* command, an additional output file was created. The stopping criteria for the energy minimization are set to 10^{-12} eV for the energy and 10^{-6} eV/Å for the forces.

units	Sets the style of units used for a simulation e.g. *real*, *metal*, *si*, *cgs* and *electron*.
atom_style	Specifies the style of atoms to be used in a simulation. Every style stores atom IDs, types, velocities coordinates, and possibly additional attributes. Examples are *atomic*, *charge* and *electron*.
pair_style pair_coeff	*pair_style* defines the interaction between pairs of atoms within a cutoff distance by specifying a force field (e.g. ReaxxFF, QUIP, section 4.1.2) and a neighbor list. The $pair\_coeff$ command sets the coefficients associated with a chosen *pair_style*.
boundary	This command sets the boundary conditions individually for each dimension of the simulation box: p denotes periodic, f denotes fixed and s denotes shrink-wrapping (section 4.1.3).
neighbor neigh_modify	The settings that are specified by the *neighbor* command impact how the pairwise neighbor lists are constructed. With the $neigh\_modify$ command additional arguments can be specified to regulate which atom pairs should be stored and how often the neighbor lists are built.
fix	This command fixes an attribute to a group of atoms e.g. assigning a group of atoms to an ensemble (section 4.1.4).
velocity	Assigns a velocity to a group of atoms, which can be specified with arguments, keywords, and values in a wide range. An additional fix command can be used to control the velocity distribution via a selected thermostat (section 4.1.5).
timestep	This command sets the discrete time interval over which the equations of motion are numerically integrated (section 4.2.1). The specific value depends on the selected units (*real*, *metal*,..)
thermo	Controls the output of thermodynamic information during an MD simulation by setting an interval of N timesteps. The thermodynamic quantities are printed to the output file after timesteps that are a multiple of N.
run	This command is used to perform the main MD simulation, allowing the system to evolve over a specified number of timesteps.
minimize	*minimize* adjusts the atomic positions of an MD simulation to find a local energy minimum. This energy minimization stops when one of the stopping criteria is satisfied.
dump	This command writes the output information about the system during an MD simulation to an output file. The output information, the number of timesteps after which the information is written to the output file, the output file format, and additional settings can be modified over a wide range.

Table 4.1: List of frequently used LAMMPS commands

Simulations with ReaxFF

For reasons of comparability, some MD calculations were carried out with the ReaxFF potential as *pair_style*. With this setting, LAMMPS receives the pair coefficients from an extern *.reac file via the *pair_coeff* command. The parameters, such as for the relaxations or the convergence tolerance, are defined within the *fix* command. The *units* command is set to *real*, so the distances in the MD simulations are measured in Å, the time in fs, the energy in kcal/mol, and the temperature in K. For *atom_style*, *charge* is chosen, which means, that also the charges are an additional attribute of the atoms. All other commands remain unchanged from those with the QUIP potential.

5 The Gaussian Approximation Potential (GAP) method

Accurate and computationally efficient interatomic potentials have long been a challenge in the field of atomistic simulations. The demand for predictive models that can represent complex quantum mechanical interactions between atoms has grown as computational capacity has increased. Machine Learning Interatomic Potentials (MLIPs) are a novel approach that provide an innovative method to overcome the drawbacks of conventional force fields and quantum mechanical techniques. These enable MD simulations to be performed with accuracy on par with *ab-initio* techniques, such as DFT, but at a much lower computational cost.

In this thesis, the Gaussian approximation potential method (GAP) was used to train MLIPs. By using a training dataset as a basis, this method creates a force field that is specifically designed for the dry oxidation of silicon. GAP demonstrated remarkable performance in capturing the intricacies of interatomic interactions. The GAP framework utilizes a Gaussian process regression model to predict potential energy surfaces and forces, achieving a high level of accuracy while maintaining computational efficiency. Significant contributions to this approach include the work of Bartók et al., who demonstrated its application to a variety of materials [105]. GAP has also been successfully applied to more complex systems [10, 106, 107], highlighting its potential to bridge the gap between the accuracy of *ab-initio* methods and the efficiency required for large-scale simulations. The theory behind GAP and how MLIPs are getting trained will be described in the following.

5.1 Theoretical background

The Gaussian Approximation Potential (GAP) method was originally proposed by Bartók et al. 2010 [108]. The GAP approach uses a training dataset of atomic configurations with their properties (e.g. their total energy, forces, and virial stress components) derived from *ab-initio* calculations, to design a MLIP than can predict the energy and forces of interacting atoms in an unknown system. The electrical structure of the atomic configurations is completely neglected here.

For a particular atomic structure, GAP computes the energy of the structure and

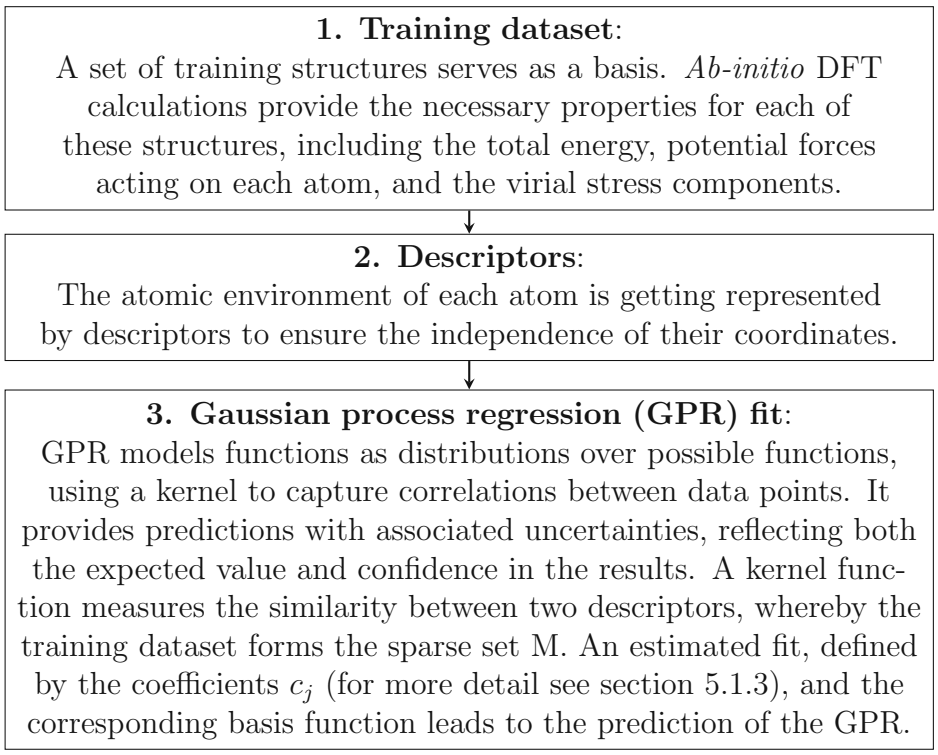


Figure 5.1: Workflow of the GAP fit process explained in three steps.

the forces acting on each atom. To do so, GAP proceeds according to a workflow which is represented in Fig. 5.1.

5.1.1 Total potential energy

The total potential energy E of an atomic configuration is the sum of all individual atomic energies ϵ_i, which also can be expressed as a combination of a short-range interatomic interaction term E_{short} and a long-range interatomic interaction term E_{long}:

$$E = \sum_i \epsilon(\{\mathbf{r}_{ij}\}) = E_{\text{short}} + E_{\text{long}} \tag{5.1}$$

Here, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ denotes the relative position between atom i and atom j. E_{short} provides nearly all of the total energy E of the system, therefore can E be restricted to E_{short}, which consists of atoms j that are within a cutoff distance r_{cut} with respect to atom i: $|\mathbf{r}_{ij}| < r_{\text{cut}}$.

5.1.2 Descriptors

To make the atomic configurations independent of their coordinates, GAP uses so-called descriptors. This significantly reduces the number of training data structures. A descriptor maps the atomic configuration to a vector and ensures invariance to rotation, translation, or permutation of identical atoms [109]. In addition to the conventional 2-body and 3-body descriptors, where either two or three atoms are related to each other, GAP also makes use of the smooth overlap of atomic positions (SOAP) descriptor. With SOAP, the representation of the neighbor atoms j of the considered atom i of element a is given by a set of neighbor densities [110]:

$$\rho^{i,a}(\mathbf{r}) = \sum_j \delta_{aa_j} \exp\left[\frac{-|\mathbf{r}-\mathbf{r}_{ij}|^2}{2\sigma_a^2}\right] f_{\text{cut}}(r_{ij}) \tag{5.2}$$

The neighbor atoms j that are considered for the description, which are of an element a, are within a cutoff distance r_{cut}. f_{cut} represents a cutoff function, that gradually approaches zero at r_{cut}. The smoothness of the representation is given by the hyper-parameter σ_a, which has length units.

The whole set of elemental neighbor densities is created for each atom i. To ensure rotational invariance, the neighbor density is expanded into a local basis of orthogonal radial functions $R_n(r)$ and spherical harmonics $Y_l^m(\hat{r})$:

$$\rho^{i,a}(\mathbf{r}) = \sum_{nlm} c_{nlm}^{i,a} R_n(r) Y_l^m(\hat{\mathbf{r}}) \tag{5.3}$$

$$c_{nlm}^{i,a} = \int d\mathbf{r} R_n(r)^* Y_l^m(\hat{\mathbf{r}})^* \rho^{i,a}(\mathbf{r}) \tag{5.4}$$

with the expansion coefficients $c_{nlm}^{i,a}$. $R_n(r)^*$ and $Y_l^m(\hat{\mathbf{r}})^*$ denotes the complex conjugate of $R_n(r)$ and $Y_l^m(\hat{\mathbf{r}})$ respectively. The rotational invariant power spectrum is then given by

$$p_{nn'l}^{i,aa'} = \frac{1}{\sqrt{2l+1}} \sum_m \left(c_{nlm}^{i,a}\right)^* c_{n'lm}^{i,a'} \tag{5.5}$$

whereas $\left(c_{nlm}^{i,a}\right)^*$ denotes the complex conjugate of $c_{n'lm}^{i,a'}$. This power spectrum, which gives a concise representation of atomic neighbor environments, is frequently referred to as the SOAP descriptor or SOAP vector [111]. The SOAP descriptor is a function of the following five indices: the angular channel l, the radial channels n and n', and the neighbor-element channels a and a'. The only free parameters of SOAP are the cutoff distance r_{cut} and the length scale σ_a.

5.1.3 Gaussian process regression (GPR) method

Only the energies of the training structures are precisely known on the descriptor-generated representation. To calculate the energy of an unknown atomic system, GAP uses the Gaussian process regression (GPR) method. In GPR, the energy E_A of the unknown atomic configuration A is given by [112, 113]:

$$E_A = \sum_{i \in A} \sum_{j}^{M} c_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{5.6}$$

The left sum covers all the descriptor vectors $\mathbf{x}_i$ in the unknown atomic configuration A. The second sum is over a collection of representative descriptor vectors $\mathbf{x}_j$ from the provided dataset (sparse points M). It describes a linear combination of M basis functions of $k(\mathbf{x}_i, \mathbf{x}_j)$, which are weighted with the regression weights c_j. k is a Kernel function and measures in our case the similarity between the two descriptors $\mathbf{x}_i$ and $\mathbf{x}_j$. The kernel functions have Gaussian forms, from which the GPR is derived. The regression weights c_j have to be fitted to match the *ab-initio* values of the sparse points M as closely as possible. This can be expressed as a regression problem by minimizing the loss function $\mathcal{L}$ with respect to the weights $\mathbf{c} = \{c_j\}$. In matrix notation, $\mathcal{L}$ is given by

$$\mathcal{L} = (\mathbf{y} - \hat{\mathbf{y}})^T \Sigma^{-1} (\mathbf{y} - \hat{\mathbf{y}}) + \mathbf{c}^T \mathbf{K}_{MM} \mathbf{c} \tag{5.7}$$

The N *ab-initio* reference properties are represented by $\mathbf{y}$, while $\hat{\mathbf{y}}$ represents the predicted properties. Σ denotes a diagonal matrix with elements inversely linked to the significance of each data point, containing the regularisation strength parameters σ_{energy}, σ_{force} and σ_{virials}. $\mathbf{K}_{MM}$ is the kernel matrix and its elements is derived from the kernel function values $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, which are calculated between the sparse point set $\{\mathbf{x}_i\}_i^M$.

The first term in the equation (5.7) controls the fit to the data points, while the second term attempts to avoid overfitting via a Tikhonov regularisation [114]. The minimization of the equation (5.7) leads to

$$\mathbf{c} = (\mathbf{K}_{MM} + \mathbf{K}_{MN} \Sigma^{-1} \mathbf{K}_{NM})^{-1} \mathbf{K}_{MN} \Sigma^{-1} \mathbf{y} \tag{5.8}$$

with $\mathbf{K}_{NM} = \mathbf{K}_{MN}^T$, whereas the components of $\mathbf{K}_{MN}$ are defined as

$$K_{ij} = \sum_{\alpha \in j} k(\mathbf{x}_i, \mathbf{x}_\alpha) \tag{5.9}$$

The summation goes over all descriptors $\mathbf{x}_\alpha$, which contribute to y_j, while the sparse set's descriptor is given by $\mathbf{x}_i$.

With the coefficients $\mathbf{c}$ and equation (5.6) the total energy E_A of the atomic configuration A can be evaluated. By differentiating E_A with respect to atomic coordinates or lattice deformations one can determine the forces and the virial stress components of A.

5.2 GAP training

The training of the MLIPs is carried out with the *gap_fit* program, which is part of the software package Quantum Mechanics and Interatomic Potentials (QUIP) [115]. The source code can be found on GitHub and is publicly available [116]. QUIP also comes with a Phyton interface, namely quippy, which provides access to a range of functions and converts all atomic potentials into Atomic Simulation Environment (ASE) calculators. The GAP models can also be easily integrated into MD simulation programs such as LAMMPS.

The training of a new GAP consists of the following 3 steps:

(1) Reading input data

(2) Choosing the descriptors and kernels that serve as basis functions

(3) Setting the parameters to control the least squares fit

5.2.1 Input data

The input data has to be provided for the *gap_fit* program in the extended XYZ format. Each atom is represented in this format by its atomic number, cartesian coordinates, and, if desired, additional forces and virial stresses. The position of the atoms has to be provided in the units of Å, the energy in the units of eV, the forces in units of eV/Å, and the stresses in units of eV (this is the volume times the normal stress).

The input XYZ file is a sequence of atomic configurations, whereas a periodic lattice unit cell is required for each structure. To obtain optimal fitting, consistency in the calculation of forces and energies is very important. Including not only the energies but also the forces improves the fits tremendously. For determining elastic constants for periodic solids, virial stresses are crucial.

In addition to the actual structures which serve as a training set for the GAP, each training set must also contain the isolated atoms of each atom type that occur in the training set. To improve the GAP fit, it is highly recommended, that the training set also include dimers of every combination of atom types of interest.

5.2.2 Specifying descriptors and kernels

The next step is defining and specifying the descriptors and the kernels. As mentioned in the previous section, GAP makes use of the 2-body, the 3-body, and the SOAP descriptor. The 2-body and the 3-body descriptors can be specified via the following values in the *gap_fit* program:

- *order* defines the order of the N-body descriptor (e.g. $order = 2$ or $order = 3$)
- The cutoff radius $cutoff$, expressed in units of Å, specifies the maximum interatomic interaction distance that describe interactions.
- The number of the sparse points is set by n_sparse.
- *covariance_type* defines the form of the kernel and *delta* the scaling of the kernel (in eV). *delta* indicates how significant this description is to the total potential and adjusts the influence of each descriptor on the overall potential. A smaller *delta* value results in a more localized kernel, giving greater weight to descriptors that closely match the training data.
- $theta\_uniform$, given in units of Å, defines the length scale of the Gaussian Kernel and specifies, how fast the Gaussian kernel decays.
- The choice of representative points is set with $sparse\_method$.
- *compact_cluster* determines how the cutoff is applied (e.g. T describes a sphere around the central atom)

In contrast to the 2-body and the 3-body descriptors, the SOAP descriptor uses a basis set of spherical harmonic functions. A single real number is obtained by projecting the atomic density onto these basis functions, which results in the SOAP vector as a collection of these numbers. The higher the number of basis functions in the basis set, the more accurately the chemical environment is mapped. On the downside, this also comes with increased computational costs. The SOAP descriptor has the following additional parameters compared to the N-body descriptors:

- l_max and n_max set the number of angular and radial basis functions.
- The Gaussian smearing width of the atom density is defined with $atom\_sigma$, in units of Å.
- $cutoff\_transition\_width$ sets the distance at which the kernel is gradually reduced to zero.
- *zeta* specifies the power of the polynomial kernel exponent.

5.2.3 Regularisation strength parameters

Finally, the regularisation strength parameters σ_{energy}, σ_{force}, σ_{virials} and σ_{hessians} have to be set in the *gap_fit* program. It is very useful to divide the training set into subcategories (e.g. single atoms, dimers, bulk, etc.), each with its own specific sigma values. σ determines the convergence criteria for the energy, the forces, the virials, and the Hessians, respectively. The lower this value, the more accurately the corresponding structure is represented in the MLIP.

5.2.4 Output data

The resulting GAP model is stored in several output files, which consist of one $*.xml$ file and a set of text files. These files can easily be integrated into the MD simulation software LAMMPS by using the QUIP plugin. The setting for an MD simulation with GAP in LAMMPS can be found in section 4.2.3.

To increase the accuracy of a trained GAP, active-learning techniques are appropriated, as proven by Deringer et al. [111]. The process begins with the use of a newly trained GAP as the potential in a MD simulation. During this simulation, atomic trajectories are generated, providing insights into the system's behavior. From these trajectories, specific structures are selected for further analysis based on their relevance or deviations from the model's predictions. These selected structures are then recalculated using DFT, which serves as a more accurate reference for the potential energy and forces. The results from these DFT recalculations are combined with the original training set to create an updated dataset. This augmented dataset is then used to train a new GAP, which incorporates the additional, high-fidelity information from the recalculated structures.

The re-training process is iterative and can be repeated as needed to progressively refine the accuracy of the GAP. Each iteration aims to reduce discrepancies between the model predictions and DFT results, ultimately enhancing the GAP's performance. This approach ensures that the potential remains highly accurate and capable of representing the atomic interactions with the required precision.

6 Generating the GAP Training data

The previous chapter discussed the theory behind GAP and how to train a new GAP potential. This chapter looks at the role of the training data itself and what structures have been created to train a GAP specifically for the oxidation of silicon.

6.1 Selection of the GAP training structures

The quality of the obtained GAP potential relies heavily on the quality and diversity of its training data. The training dataset serves as an information repository containing the relationships between the atomic configurations and the corresponding potential energies. The completeness of these data has a direct influence on the accuracy and reliability of the trained GAP model. A well-constructed and comprehensive training dataset is mandatory for the model to generalize effectively over a wide range of material configurations.
There are two essential requirements a training data needs to meet. The first requirement is that the training set must be representative of the diverse range of atomic environments and configurations. Inaccuracies or biases in the training data may lead to a model that fails to capture the true complexity of the interactions within the material. The second requirement is that the training data should originate from high-fidelity quantum mechanical calculations, such as first-principles DFT simulations, to bridge the gap between quantum mechanical accuracy and computational efficiency.
The challenge in training a new GAP lies in the selection and preparation of the training structures. Issues such as the size of the dataset and the trade-off between computational cost and model accuracy must be carefully considered. Iterative refinement of the training data and continuous validation of the resulting models are integral parts of the GAP development process. Based on the performance of the previously trained GAP, the training data must be readjusted for each new training loop.

6.2 GAP training structures for SiO_2

For the training of a GAP, which is intended to physically reproduce the oxidation of silicon correctly, a variety of different structures are generated consisting of silicon

atoms, oxygen atoms, and/or hydrogen atoms. Hydrogen atoms are solely used to saturate the lowest Si atoms of the Si surface structures but do in general not influence oxidation. Additionaly, the information associated with each of the atomic structures (such as the total energy, the forces on each atom, and possibly the virial stress components) is provided by DFT calculations performed with the CP2K software. This information is stored as a sequence of atom configurations in an extended XYZ format, which serves as the input file for the GAP learning algorithm. In the following a more detailed description of the individual training structures is given. An overview of the employed training structures can be found at the end of this chapter.

Rather than using the complete set of generated DFT data for the training of the GAP, some of the initially created structures are excluded from the training datasest. After the training process, these structures are utilized as test cases for unknown structures. Energy and force predictions of both GAP and DFT can be compared based on this testing dataset. The number and type of structures against which the GAPs were tested can be found in more detail in Chapter 8.

6.2.1 Single atoms

As briefly mentioned in chapter 5.2.1, each training set must contain the isolated atoms of each atom type that occurs. This allows GAP to learn the correct free energies and ensure that it correctly predicts the energy of isolated atoms. For the training of a GAP for SiO_2 each dataset contains a single Si-, O-, and H-atom. The size of the simulation cell for the DFT calculations of the single atoms was set to $20\times20\times20$ Å for the x-, y-, and z-direction respectively. The total energy according to our DFT framework is -102.282 eV for a single Si-atom, for an O-atom -430.92 eV, and for an H atom -13.577 eV.

6.2.2 Dimers

To significantly improve the GAP fit, each training set should also include the dimers of every atom combination of interest, so the following dimers were added to each training set: 97 Si-Si dimers, 23 Si-O dimers, 52 Si-H dimers, and 57 O-O dimers, in total 229 dimers.

Small steps were taken to adjust the spacing between the two atoms under consideration for the DFT calculation of the dimers. As a result, a total energy curve that resembles the Lennard-Jones potential is produced as a function of distance. Fig. 6.1 displays this curve in the context of the Si-Si dimer example, whereas the energy minimum of $E = -208.348$ eV occurs at an atom distance of $r = 2.15$ Å. Compared to the Si-Si dimers, the other dimers curves have the following energy minima:

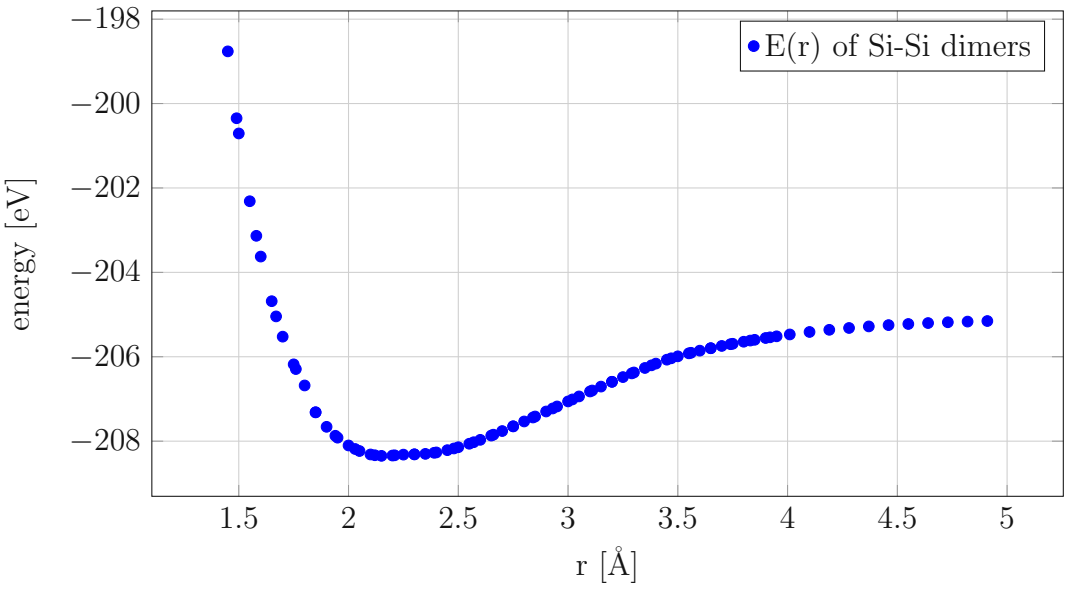


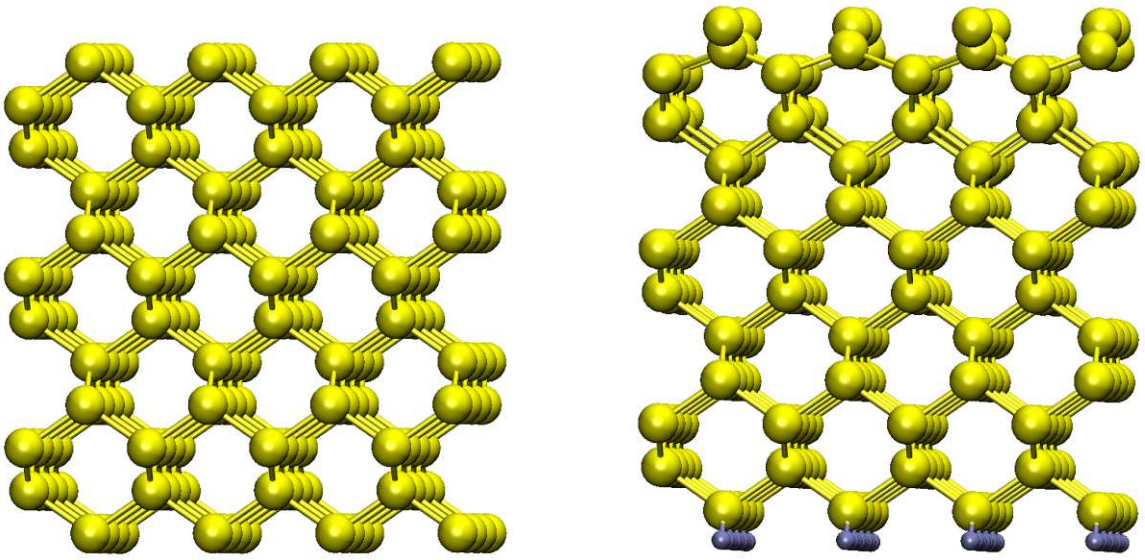
Figure 6.1: The total energy E of Si-Si dimers given in eV as a function of the atom distance r given in Å.

dimer	energy minimum	atom distance
Si-Si	-208.348 eV	2.15 Å
Si-O	-542.281 eV	1.50 Å
Si-H	-119.234 eV	1.55 Å
O-O	-868.486 eV	1.25 Å

6.2.3 Bulk Silicon

Bulk silicon has a diamond lattice structure in which each silicon atom is tetrahedrally surrounded by four other silicon atoms (for more detail refer to section 2.1). This lattice is highly regular and crystalline, contributing to the outstanding electronic properties of silicon.

A total of 201 slightly different bulk Si structures were calculated. Each of the bulk structures consists of 192 Si atoms with each cell measuring 15.523 Å in the x- and y-direction and 16.22 Å in the z-direction. The different bulk configurations were obtained from MD simulations as the initial Si bulk configuration is running at a defined temperature. After a certain time interval Δt, the positions of the atoms are written into an output file. In Fig. 6.2a one of these configurations is illustrated. The total energy, the forces on each atom, and the virial stress components for each atomic configuration were evaluated using periodic cells in DFT.



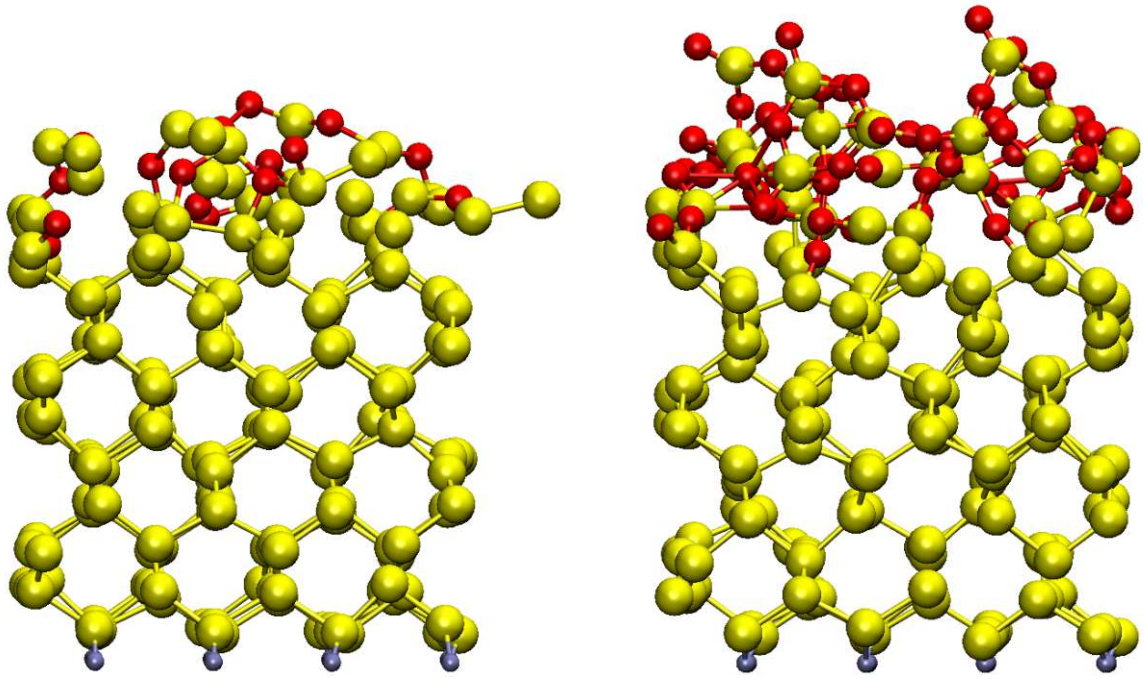
(a) Bulk silicon with 192 Si atoms

(b) Clean silicon surface with 224 atoms

Figure 6.2: Panel (a) shows one of the 201 bulk silicon structures with a diamond lattice consisting of 192 Si atoms and a cell size of 15.523×15.523×16.22 Å. Panel (b) shows one structure of the 93 clean silicon surfaces consisting of 192 Si atoms and 32 H atoms. The cell size of each surface is 15.523×15.523A×37.22 Å with around 21 Å of vacuum above each surface, which is not shown in the figure.

6.2.4 Clean silicon surfaces

Since the GAP is primarily trained for the oxidation of silicon surfaces, it is crucial that the training set also includes clean silicon surface structures. A total of 93 clean surface structures were calculated, which, like the silicon bulk structures, were generated by MD simulations. 6.2b shows one of the clean silicon surface structure. The cell size of each structure is 15.523 Å in the x- and y-direction and 37.22 Å in z-direction. Each structure consists of 192 Si atoms and 32 hydrogen atoms, a total of 224 atoms. Hydrogen is used to passivate the dangling bonds on one side of the Si slab, while the clean Si-surface reconfigurates upon relaxation.



(a) Oxidized silicon surface with 18 oxygen atoms

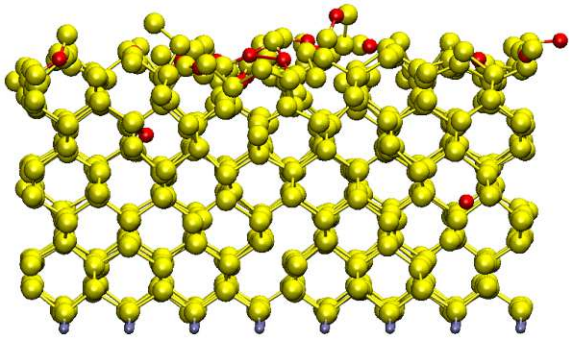
(b) Oxidized silicon surface with 60 oxygen atoms

Figure 6.3: Differently oxidized silicon surfaces. Panel (a) shows a silicon surface with an oxide consisting of 18 oxygen atoms. From this variant, 86 structures in total were computed. Panel (b) shows an oxidized silicon surface consisting of 60 oxygen atoms and 224 Si atoms, from which 76 structures in total were calculated. The cell sizes of both variants are 15.523×15.523×37.22 Å.

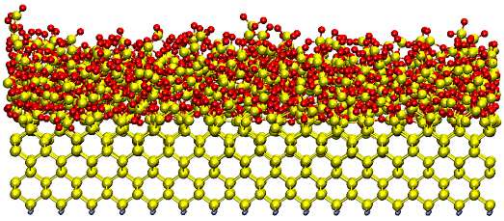
6.2.5 Oxidized silicon surfaces

To ensure that GAP accurately learns the oxidation process of silicon, various oxidized silicon surfaces have been added to particular GAP training sets. These structures may differ in cell size and number of atoms. An overview of these training structures is given in Tab. 6.1.
For structures with a cell size of 15.523×15.523×37.22 Å, four different oxidized variants, which differ in the number of atoms, were used as training structures. MD simulations were used to generate similar structures in all 4 variants, followed by subsequent single-point computations. A total of 422 training structures were created. Fig. 6.3a shows the variant consisting of 242 atoms and Fig. 6.3b shows the variant consisting of 284 atoms.

Of the oxidized structure with a cell size of 31.046×31.046×60.00 Å, single-point cal-



(a) Oxidized silicon surface with in total 920 atoms and a cell size of 31.046×31.046×60.00 Å



(b) Oxidized silicon surface with in total 5211 atoms and a cell size of 62.092×62.092×60.00 Å

Figure 6.4: Two oxidized silicon surfaces with different cell sizes. Panel (a) shows an oxidized surface with a cell size of 31.046×31.046×60.00 Å consisting of 920 atoms. In total, 51 structures of this variant were calculated. Panel (b) shows one of the 9 oxidized surfaces with a cell size of 62.092×62.092×60.00 Å. These structures, consisting of between 5211 and 5258 atoms, were created by means of MD simulations and subsequently geo-optimized using DFT calculations.

culations were carried out for a total of 51 structures, with the number of atoms varying between 901 and 1009 atoms. These structures were also generated using MD simulations. Fig. 6.4a shows one of these structures.

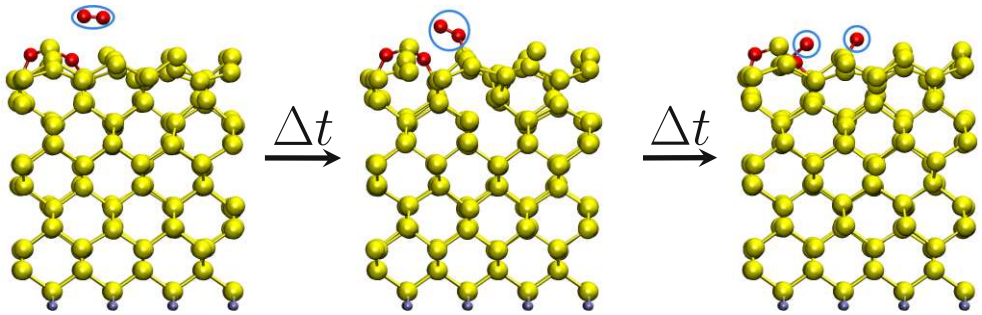
cell size (xyz) [Å]	number of atoms	number of structures
15.523×15.523×37.22	232	180
15.523×15.523×37.22	242	86
15.523×15.523×37.22	272	80
15.523×15.523×37.22	284	76
31.046×31.046×60.00	901 - 1009	51
62.092×62.092×60.00	5211 - 5258	9

Table 6.1: Overview of the oxidized silicon surfaces used in the training dataset

In contrast to the previously mentioned structures, the 62.092×62.092×60.00 Å training data were generated using geometry optimization calculation. Four differently oxidized structures were subjected to geometric optimization. The forces on the atoms and the total energy before and after the geometry optimization were included in the training dataset. Also, for one of the four structures, the forces and energy originating from during the geometry optimization process were added to the

training data, resulting in a total of 9 training structures. In Fig. 6.4b one of these geometry-optimized structures is shown.

6.2.6 Defect-free oxidized surfaces



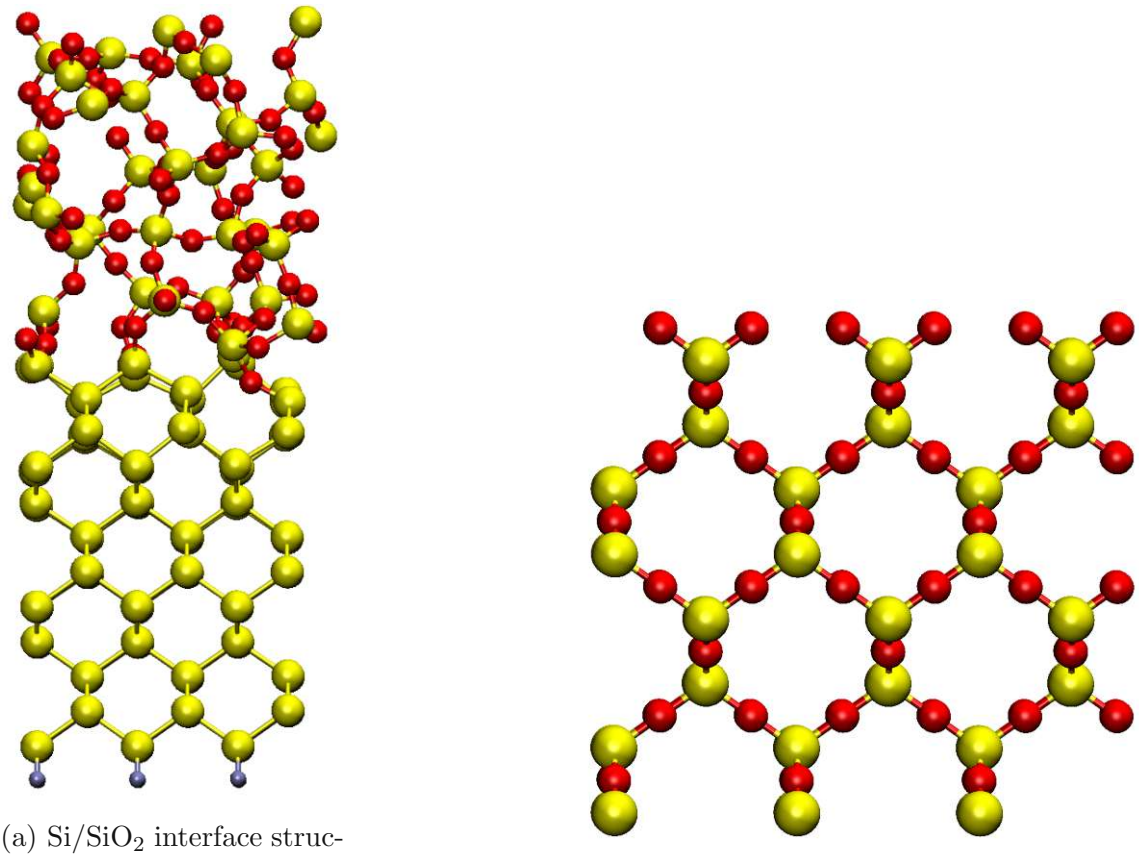
(a) Dissociation step 1 (b) Dissociation step 2 (c) Dissociation step 3

Figure 6.5: 3 steps of the dissociation process of an O_2 molecule on a silicon surface with a cell size of 15.523×15.523×37.22 Å. The three steps are the temporal sequence of the dissociation process, which is symbolized by an arrow, and the Δt between the subfigures. Panel (a) shows the first step of the dissociation process: The O_2 molecule (circled in blue) is not yet chemically bonded to the Si surface. Panel (b) shows step 2 where one oxygen atom of the O_2 molecule is chemically bonded to a Si atom. Panel (c) is showing step 3 of the O_2 molecule dissociates: The bond between the two O atoms breaks and the second, previously unbound oxygen atom also binds to a silicon atom. Both bonded oxygen atoms are circled in blue.

As mentioned in chapter 2.2 the properties of a SiO_2 layer are decisively determined by its defect-free condition. Therefore defect-free structures must also be present in the training dataset for GAP to learn how to produce structures free of defects.

For this thesis, the training data of such structures were generated using 2 methods:

1. Using AIMD with a total of 203 structures. Most of these data, totaling 184 structures, include frames that show an O_2 molecule dissociating on the surface of oxidized Si. The dissociations occur on ten distinct Si surfaces with varying degrees of oxidation. Fig. 6.5 depicts one of these dissociating processes in 3 steps. All surfaces possess a cell size of 15.523×15.523×37.22 Å. The remaining 19 structures are defect-free, oxidized Si surfaces with a cell size of 11.619×11.619×69.737 Å with oxidation layers of different thicknesses ranging from 0.8nm to 2.5nm.



(a) Si/SiO_2 interface structure

(b) Crystalline SiO_2

Figure 6.6: A Si/SiO_2 interface structure, which has only defect at the surface, and a crystalline silicon dioxide. Panel (a) shows one of the 25 oxidized Si surfaces from MD simulations with a total of 232 atoms and a cell size of 11.619×11.619×69.737 Å. The oxide layer is about 1.4 nm. (b) shows a bulk crystallin SiO_2 consisting of 216 atoms and a cell size of 15.201×15.201×14.332 Å. A total of 70 of these structures were created using MD simulations.

2. Using MD simulation with GAP as force field. Starting from two different, oxidized Si surfaces, a total of 47 training structures were generated using MD simulations, for which the total energy and the forces on each atom were then determined using single-point calculations. These structures possess a cell size of 11.619×11.619×69.737 Å with 232 and 233 atoms respectively. Fig. 6.6a shows the structure with 232 atoms, of which 25 training structures were created.

To prevent over-similarity between the individual structures, the period between two structures from the same AIMD or MD simulation was choosen to be sufficiently

long. A total of 250 defect-free training structures were generated.

6.2.7 Bulk SiO_2

Furthermore, pure SiO_2 structures were added to the training set to include a description of bulk a-SiO_2. Using MD simulations, a total of 70 bulk SiO_2 structures with $15.201 \times 15.201 \times 14.332$ Å cell sizes have been created. In combination with periodic boundary conditions, single-point calculations of infinity large bulk SiO_2 structures were carried. Each structure consists of 216 atoms in the form of periodically arranged SiO_4 particles, which are connected via the tetrahedron corners, whereby each tetrahedron is linked to four neighboring tetrahedrons [34]. Fig. 6.6b shows one of the bulk SiO_2 training structures.

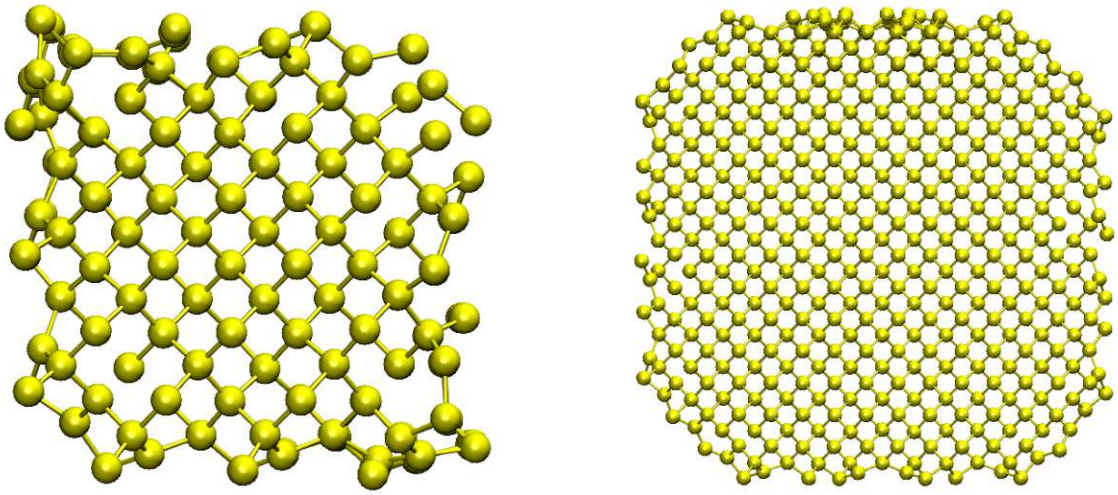
6.2.8 Clean silicon nanowires

In addition to the oxidation of silicon surfaces, GAP should also physically correctly reproduce the oxidation of silicon nanowires. Therefore it is necessary that the training set also includes clean silicon nanowire structures. For this thesis, two distinct nanowire structures were designed to train GAP, which were created by using the Wulff construction [32]. The first structure, which has a dimension in the z-direction of 31.86 Å, consists of 576 Si atoms. And the second structure, which has a dimension in the z-direction of 21.765 Å, consists of 1680 Si atoms. Before using MD simulations with GAP as force field to create different configurations, both structures were first cell-optimized to obtain their lowest total energy. These two cell-optimized structures can be seen in Fig. 6.7a and 6.7b. The cell dimension in the x- and the y-direction for the single-point calculation of the different configurations were set to 60 Å for both structures. A total of 99 training structures with 576 atoms and 101 training structures consisting of 1680 atoms were created.

6.2.9 Oxidized silicon nanowires

Two different oxidized silicon nanowires have been created to ensure that GAP accurately reproduces the oxidation process of the silicon nanowires in addition to the accurate oxidation of silicon surfaces.

The first structure consists of a clean Si nanowire with 1680 Si atoms and one O_2 molecule that dissociates at the surface. Ten dissociation processes were produced using MD simulations with GAP as force field, from which 90 training structures were taken from. Single-point calculations determined the energy, the forces on the atoms, and the virial stresses. Fig. 6.8a shows one exemplary Si nanonewire structure with one O_2 molecule adsorbing onto its surface.



(a) Clean silicon nanowire consisting of 576 Si atoms

(b) Clean silicon nanowire consisting of 1680 Si atoms

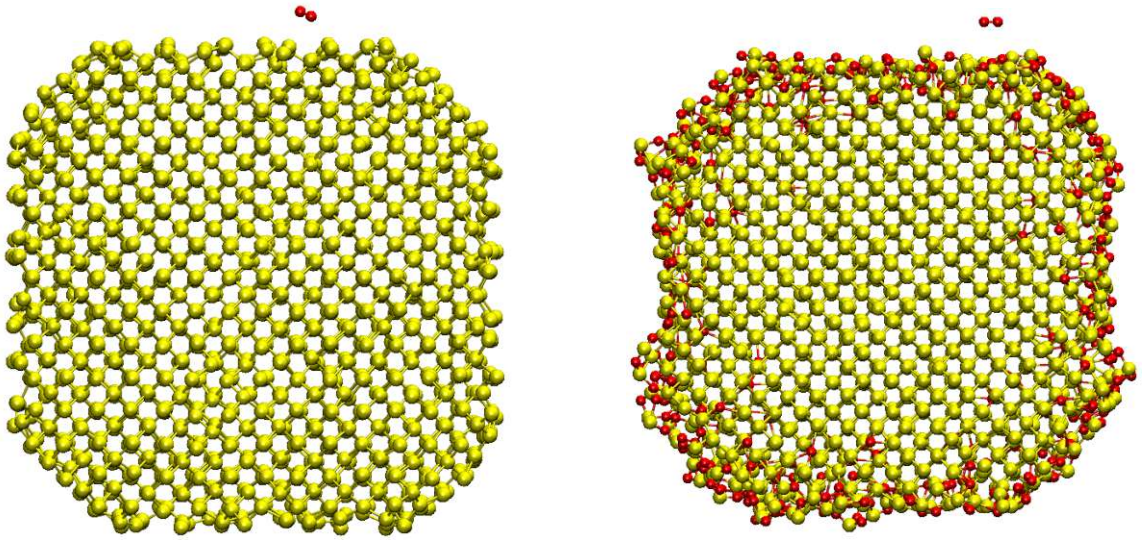
Figure 6.7: Two distinct clean silicon nanowires used for GAP training. Panel (a) shows one of the 99 cell-optimized nanowires consisting of 576 atoms. Panel (b) shows the cell-optimized nanowire consisting of 1680 atoms, from which 101 variants were calculated.

The second structure consists of an oxidized Si nanowire with 1680 Si atoms at which also one O_2 molecule dissociates. Using MD simulations, 10 dissociation processes emerged from this structure, resulting in a total of 180 structures. These structures were geometry-optimized using DFT, from which then the energy, forces, and virials were determined. Fig. 6.8b shows one of the geometry-optimized structures.

6.2.10 Active-learning training structures

As mentioned in section 5.2, active-learning techniques are a promising approach to increase the accuracy of a trained GAP. For this thesis, several training structures were created using an earlier version of the final GAP. These newly created active-learning structures served as additional training structures to improve the properties of GAP. The following structures were created using an active-learning technique:

- Oxidized silicon nanowire structures (consisting of 1680 silicon atoms, similar to Fig. 6.8b): A total of 490 structures were created in 3 training cycles. In the first training cycle, 85 structures were generated, in the second training cycle 201 structures and in the third training cycle 202 structures.
- Oxidized silicon surfaces with a cell size of 31.046×31.046×60.00 Å: A total of 500 structures were generated in 2 training cycles; 100 structures in the first



(a) Clean silicon nanowire with one O_2 molecule

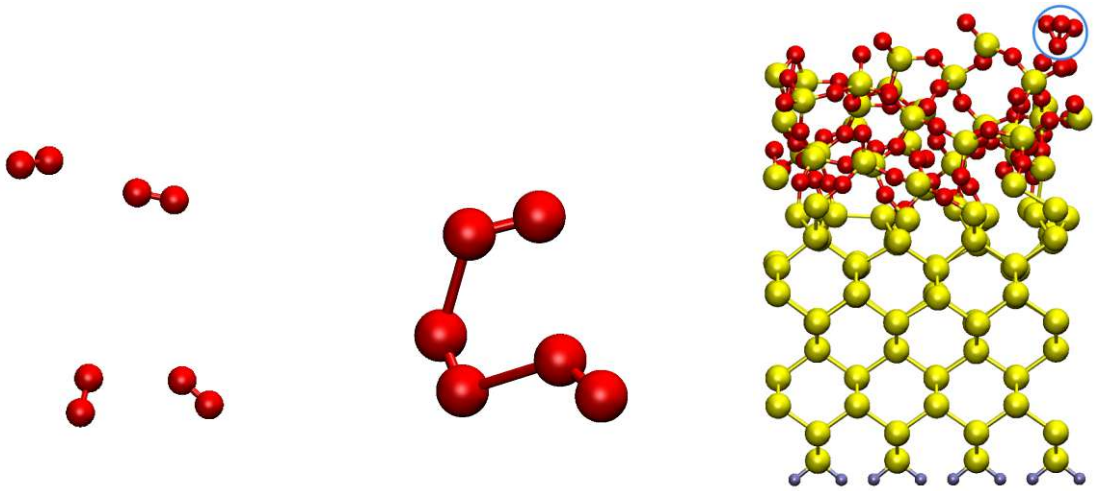
(b) Oxidized silicon nanowire with one O_2 molecule

Figure 6.8: Nanowires with O_2 dissociation. Panel (a) shows a clean nanowire, which consists of 1680 Si atoms, and one O_2 molecule near the surface. In total ten dissociation processes with this nanowire were used as training data, resulting in 90 training structures. A partly oxidized nanowire (oxide thickness $x = 0.4\,\text{nm}$) consisting of 1680 Si and 381 O atoms is shown in panel (b). Ten dissociation processes were again used as training data resulting in a total of 101 training structures.

cycle and 400 structures in the second training cycle.

- Oxidized silicon surfaces with a cell size of 15.523×15.523×60.00 Å: Using 4 differently trained GAPs, a total of 387 structures were generated.

Additionally, 15 defect-free and oxidized silicon surfaces were generated with a cell size of 15.523×15.523×60.00 Å. Here, defect-free silicon surfaces that had already been pre-oxidized with O_2 molecules were taken for the MD simulations using SiO_2 deposition to achieve a thicker oxide layer in a reasonable time. The reason is, as described in section 2.3.1, that after the first oxide formation process by chemisorption, the oxide layer grows much slower because further layer growth requires oxygen diffusion to the interface. Due to the small diffusion constant of oxygen, MD simulations using O_2 oxidation require extremely long computation times when dealing with thicker oxide layers. SiO_2 deposition accelates the oxide growth although it is certainly a more artificial approach to simulate the oxide growth and is similar to the TEOS oxide deposition [42], which is described in section 2.3.2. However, it is not feasible to reproduce the entire TEOS process in this form, as it is by far too com-



(a) 4 O_2 in a cell with size 10×10×10 Å

(b) Artifact serving for active-learning

(c) Oxidized Si surface with one artifact

Figure 6.9: Three of the five distinct oxygen structures to train GAP the behavior of oxygen. Panel (a) shows one of a total of 51 configurations generated by AIMD calculations, which consisting of four O_2 molecules placed in a 10×10×10 Å cell. Panel (b) shows an artifact consisting of 6 oxygen atoms which originates from an active-learning method. In total, 32 artifacts were created with this technique. Panel (c) shows an oxidized silicon surface with one oxygen artifact (circled in blue). The surface has a cell size of 15.523×15.523×60.00 Å.

plex. After the SiO_2 deposition process each defect-free and oxidized silicon surface was single-point calculated using DFT to get its total energy, forces, and virials.

6.2.11 Oxygen gas

Several oxygen training structures were developed for the accurate physical behavior of oxygen throughout the oxidation process in the MD simulations, which are:

- Structures consisting of two O_2 molecules: A total of 27 different structures in which two O_2 molecules are located at various distances and orientations from each other. These structures were generated by hand and the total energy and the forces on each atom were determined by single-point calculations using a in a box with side length 20 Å.
- Structures consisting of three O_2 molecules: 4 different configurations of three O_2 molecules with various distances and a cell size of 10×10×10 Åwere calculated using single-point calculations.

- Structures consisting of four O_2 molecules: These structures, in total 51, were generated by AIMD calculations and a cell size of 10 Å×10 Å×10 Å. One of these structures is shown in Fig. 6.9a.

- Active-learning structures: Six active-learning cycles were carried out, beginning with a GAP that was originally trained using only single O, O_2 dimers and configurations with two O_2 molecules. An MD simulation with 116 O_2 molecules was initiated following each GAP training. From this MD simulation, certain oxygen artifacts for new training structures were taken out, and single-point calculated using DFT calculations. A new GAP was trained using these artifacts in addition to the current training set. In this manner, 32 active-learning structures were produced. Fig. 6.9b shows one of these artifacts.

- Oxidized silicon surface with oxygen artifacts: Oxygen artifacts in the gas phase have been observed in certain MD simulations when a Si surface with a cell size of 15.523×15.523×60.00 Å was oxidized. Including oxygen artifacts, the energy of these gaseous O_2 clusters is determined by DFT. In total 29 training structures were created. Fig. 6.9c depicts one of these structures with the oxygen artifact circled in blue.

6.2.12 Overview of the training data

This section summarizes all the training structures in Tab. 6.2. This table consists of five columns: In the first column, the structure name is listed, and in the second column the elements, the number of atoms, or the configuration of the structure are listed. The third column lists the computation type of the DFT calculation and the last two columns show the number of each structure and the total number of created training structures. In total, 3311 training structures were created.

The training structures encompass a variety of configurations, including surfaces (surf.) and nanowires (NW), some of which are oxidized (oxid.) or defect-free (def.-free). These structures were selected and refined through active learning (AL) to ensure optimal performance in the GAP training process. Specifically, oxid. surf. 1 refers to the active-learning configuration of an oxidized silicon surface with a cell size of 31.046×31.046×60.00 Å, while oxid. surf. 2 denotes a similar configuration but with a smaller cell size of 15.523×15.523×60.00 Å. Additionally, oxid. surf. 3 describes defect-free and oxidized silicon surfaces with a cell size of 15.523×15.523×60.00 Å.

To obtain the necessary energies, forces, and virials at each atom for training the GAP, single-point calculations (SP), geometry optimizations (geoopt), and cell optimizations (cellopt) were performed. In some cases, *ab-initio* molecular dynamics

structure	atoms, conf.	computation	number of struct.	$\sum$
single atoms	Si, O, H	SP	3	3
dimers	Si-Si	SP	97	229
	Si-O	SP	23	
	Si-H	SP	52	
	O-O	SP	57	
bulk silicon	192	SP	201	201
clean silicon surf.	224	SP	93	93
oxidized silicon surf.	232	SP	180	482
	242	SP	86	
	272	SP	80	
	284	SP	76	
	901-1009	SP	51	
	5211-5258	geoopt	9	
def.-free oxid. surf.	178-323	AIMD	203	228
	232-233	SP	25	
bulk SiO_2	216	SP	70	70
clean silicon NW	576	cellopt	99	200
	1680	cellopt	101	
oxid. silicon NW	1682	SP	90	270
	2063	geoopt	180	
AL structures	oxid. NW	SP	490	1392
	oxid. surf. 1	SP	500	
	oxid. surf. 2	SP	387	
	oxid. surf. 3	SP	15	
oxygen gas	4	SP	27	143
	6	SP	4	
	8	AIMD	51	
	artifacts	SP	32	
	surf.+artifact	SP	29	
				3311

Table 6.2: Overview of all training data

(AIMD) simulations were also employed to capture the dynamic behavior of the atoms within the structures. These carefully curated configurations form the basis for the subsequent analysis and testing of the trained GAP models.

7 GAP Training

This chapter describes how the individual GAPs were trained. Generally speaking, GAP is trained through a meticulous procedure designed to capture the complex interactions among atoms in a material system. Therefore, the training dataset must include a wide variety of atomic configurations and chemical conditions that are representative of the material of interest. All the structures used to train the different GAPs for this thesis are listed in the preceding chapter. As will be demonstrated later, the selection and compilation of the training dataset is crucial for the GAP training. The process of training a GAP is nonlinear, making it challenging to predict how various training structures may affect the accuracy of the trained GAP.

Once the training dataset is assembled, the next step involves feature representation, where the descriptors are defined to encode information about the local atomic environments. These features serve as the basis for training the GAP model and capturing the complex atomic interactions. Furthermore, the choice of model architecture, including the selection of kernel functions and hyperparameters, is crucial to optimize the performance of the GAP model.

The contribution of each structure type can be weighted in order to accentuate or diminish the importance of the respective training data. In GAP, the weighting is controlled by a parameter σ to account for their varying importance in capturing the potential energy surface accurately. Lower sigma values indicate more critical structures, ensuring the model focuses on the most important regions of the potential energy surface.

In the following, the preparation of the training dataset and the parameters, that modify the GAP training, are described in more detail below. For the theoretical background of GAP training, refer to section 5.2.

7.1 Training dataset

The training dataset must be supplied in the extended XYZ format for the *gap_fit* program. This format contains the atomic number, the cartesian coordinates, and, if provided, additional forces and virial stresses for each atom in the simulation cell. The first line of each configuration contains the number of atoms followed by the

second line, which contains information about the cell size, the virials, the total energy, and the name of the subcategory (for assigning specific sigma values). The individual atoms are listed in the other lines, with the element written first, followed by its Cartesian coordinates and the forces in the x-, y-, and z-directions.

Fig. 7.1 shows an example of a section of a training dataset in the extended XYZ format, where three configurations can be identified: 2 dimers followed by a bulk silicon structure. In the example of the Si bulk structure, the most important details are highlighted in color: The first line shows the number of atoms for the configuration (here 192) which is highlighted in red. The second line shows, among other things, the cell size of the structure (in green), the nine virials (in purple), the subcategory (here *Si_bulk*, in yellow), and the total energy of the structure given in eV (in blue). The 192 Si atoms of the bulk structure are listed in the next 192 lines. In each line, the element is shown first (in grey), followed by the Cartesian coordinates and the three forces acting on the atom. Due to this listing of training structures, a training dataset can consist of a large number of lines.
The units of the coordinates are given in Å, the energy and the virials in eV and the forces in eV/Å. The link between the energies and forces of atoms and their configurations is what GAP is taught to reveal. Single atoms (Si, O, and H) and dimers (Si-Si, Si-O, Si-H, and O-H) were therefore always included in all training datasets so that the trained GAP can act as a suitable surrogate model for the PES of the system. Each training set also included bulk silicon structures, bulk SiO_2 structures, and clean silicon surfaces. Other training structures in the training dataset varied in type and number and the selection depended, among other things, on the performance of the previously trained GAP. An overview of the individual training datasets used for training the individual GAPs can be found in section 7.3.

It is important to remember that a GAP training time does not increase linearly with the number of training structures. The following factors also determine the duration of a GAP training:

- Type of training structure (including the number and type of atoms it contains)
- Provided structure information (total energy of the structure, forces on each atom, virials)
- Settings of the descriptors
- The trade-off between the smoothness of GAP and its accuracy in fitting the structures in the training dataset

The quality and quantity of the training data are crucial for the performance of the trained GAP. An insufficient or unrepresentative training dataset can lead to a

```
2
Lattice="20.0 0.000 0.000 0.000 20.0 0.000 0.000 0.000 20.0" Properties=species:S:1:pos:R:3:forces:R:3
dft_virial="0.0015699045669210153 3.3406804194205855e-09 2.2052777812547146e-08 3.3406804194205855e-09
0.0015698965304886813 2.076212711152301e-08 2.2052777812547146e-08 2.076212711152301e-08
3.3472670671290823" config_type=dimer energy=-866.6509568910611 pbc="T T T"
O       10.0    10.0    10.0            0.00000051      0.00000051      25.49921907
O       10.0    10.0    8.95            -0.00000051     -0.00000051     -25.45194777
2
Lattice="20.0 0.000 0.000 0.000 20.0 0.000 0.000 0.000 20.0" Properties=species:S:1:pos:R:3:forces:R:3
dft_virial="0.0015690055695780129 1.8145048376429287e-09 -4.333380302527006e-07 1.8145048376429287e-09
0.0015690545910330138 -1.7263726941733188e-06 -4.333380302527006e-07 -1.7263726941733188e-06
4.996129069717107" config_type=dimer energy=-865.0335654231795 pbc="T T T"
O       10.0    10.0    10.0            -0.00000668     -0.00002777     39.96144272
O       10.0    10.0    9.0             0.00000668      0.00002777      -39.94007428
192
Lattice="15.523 0.0 0.0 0.0 15.523 0.0 0.0 0.0 16.22" Properties=species:S:1:pos:R:3:forces:R:3
dft_virial="0.12503993075343584 7.795487669927857e-05 8.948953555979764e-05 7.795487669927857e-05
0.1255541849618535 -3.956401525716174e-05 8.948953555979764e-05 -3.956401525716174e-05
0.20966782715731422" config_type=Si_bulk energy=-20560.550633271705 pbc="T T T"
Si      2.9106642       2.91064648      0.67583897              -0.00014964     -0.00002982     -0.00014552
Si      10.67200635     8.73168678      10.13749846             0.00000668      0.00013627      0.00014707
Si      14.55267898     8.73167172      10.13750481             0.00171853      -0.00018461     -0.00068289
Si      2.91066049      12.61233988     10.13750846             0.00002571      0.00011210      0.00012238
Si      6.7913385       12.61235237     10.13749867             0.00011673      0.00003908      0.00005296
Si      10.67200596     12.61234758     10.13750759             -0.00018975     0.00001440      0.00003291
Si      14.55268059     12.6123475      10.13749655             0.00165888      0.00018975      -0.00084898
Si      2.91065186      2.91066751      11.48917782             -0.00010284     0.00007508      0.00021083
Si      6.79132798      2.91067714      11.48916294             -0.00009359     -0.00006891     0.00034710
Si      10.67199832     2.91066837      11.48917704             0.00013421      -0.00010799     0.00031213
Si      6.79133291      8.73167299      10.13750399             -0.00001440     -0.00000051     0.00019540
Si      14.55267274     2.91066982      11.48915965             0.00192010      0.00007302      0.00117705
```

Figure 7.1: Section of a training dataset in the extended XYZ format, where two dimers and one bulk silicon structure are listed. The most important data are highlighted in color in the example of the Si bulk structure: The number of atoms in red, the cell size in green, the virials in purple, the subcategory in yellow, the total energy in blue, and the element symbol in grey. For each atom, the Cartesian coordinates and the forces acting on the atom are displayed next to the element symbol.

poorly performing GAP model. A wide variety of atomic configurations and chemical environments is therefore essential. In addition, selecting and constructing suitable features or descriptors representing the local atomic environment is a challenge. Effective features should be able to capture important physical properties of the system while being dimensionally reduced to avoid overfitting. The problem of overfitting occurs when the model becomes too closely tailored to the specific training data, capturing not only the underlying trends but also the noise and inconsistencies [117]. This results in a model that performs exceptionally well on the training set but poorly on unseen data or new configurations.

7.2 The gap_fit program

The gap-fit program, a GAP learning algorithm that is included in the QUIP package, is essentially a single line of command that contains all the instructions to train a GAP on the provided training dataset [105]. Fig. 7.2 shows an example of how this line of code is structured. The following is a description of the most important keywords:

- *energy_parameter_name, force_parameter_name, virial_parameter_name*: These parameters tell QUIP where the energies, forces, and virials are written in the training dataset. These names have to match the ones listed in each atomic configuration's information line (the second line in the training dataset in the extended XYZ format).
- *at_file*: This parameter specifies the training dataset filename
- *gap=*: Start of the descriptor and kernel specifications. The individual descriptors are separated from each other by colons.
- *distance_Nb*: Sets an N-body descriptor, *order=2* defines an 2-body descriptor and *order=3* an 3-body descriptor. The *cutoff* is the cutoff distance in the kernel given in Åand defines the maximum distance of the interatomic interactions that will be evaluated. *n_sparse* defines the number of representative points and controls the resolution. The *covariance_type* sets the form of the kernel, which is in our example *ard_se*. *ard_se* means Automatic Relevance Determination Squared Exponential and models the correlation between the input variables as an exponentially decreasing function of the distance between the inputs. *delta*, given in eV, sets the scaling of the kernel and indicates how significant this description is to the total potential. *theta_uniform*, given in Å, sets the length scale of the Gaussian kernel and specifies the rate of decay of the Gaussian kernel. With *sparse_method* one can set the distribution of the representative points. In our example, it is set to *uniform*, which results in a uniform grid up to the cutoff. *compact_cluster* specifies how the cutoff is applied. *T* in our example sets it spherically around each atom.
- *soap*: Sets the SOAP descriptor. *atom_sigma*, given in Å, sets the Gaussian smearing width of atom density for SOAP. *l_max* and *n_max* define the number of angular and radial basis functions for SOAP. *cutoff* is again the cutoff distance in the kernel and *cutoff_transition_width* sets the distance across which kernel is smoothly taken to zero, both given in Å. *delta*, *covariance_type* and *n_sparse* are the same as for the N-body descriptors. *zeta* sets the power the kernel is raised to. In our example *zeta=4* means to the 4^{th} power.

```
gap_fit energy_parameter_name=energy force_parameter_name=forces
virial_parameter_name=dft_virial
at_file=training_data_ML_15.xyz
gap={distance_Nb order=2 cutoff=4.0 n_sparse=15 covariance_type=ard_se delta=4
theta_uniform=2.0 sparse_method=uniform compact_clusters=T :
distance_Nb order=3 cutoff=3.0 n_sparse=45 covariance_type=ard_se delta=1
theta_uniform=2.0 sparse_method=uniform compact_clusters=T :
soap atom_sigma=0.5 l_max=4 n_max=8 cutoff=5.0 cutoff_transition_width=1.0
delta=0.4 covariance_type=dot_product n_sparse=1000 zeta=4}
default_sigma={0.002 0.02 0.02 0.0}
config_type_sigma={dimer:0.01:0.1:0.1:0.0:single:0.0001:0.001:0.001:0.0:bulk:0.002:
0.02:0.02:0.0} sparse_jitter=1e-8
gp_file=GAP_name.xml > calc.out
```

Figure 7.2: Example of the code line for the *gap_fit* program. Some keywords are highlighted in color. The description of them can be found in the text.

- *default_sigma*: These 4 numbers regulate the trade-off between the smoothness of the GAP and its accuracy in fitting the structures in the training dataset. The first number corresponds to the energy, the second to the forces, the third to the virials, and the fourth to the Hessian. The GAP will match the training data more accurately the lower these numbers are but it increases the probability of overfitting.
- *config_type_sigma*: With this keyword, different sigma values can be attributed to the different training structures. The names must match the names of the sub-categories in the extended XYZ file.
- *sparse_jitter*: Sets the additional diagonal regulariser. To avoid numerical instabilities during GAP training, a small random error can be specified.
- *gp_file=*: Sets the name of the output files.

The calculation effort required for GAP training can be significant, especially for large datasets and complex training structures. To keep the calculation time within reasonable limits, all GAPs in this thesis were calculated on the Vienna Scientific Cluster (VSC).

7.3 Overview of the trained GAPs

For this thesis, a large number of GAPs were trained, which differ in the type and number of training structures, as well as in different settings in the *gap_fit* program.

The GAP training aimed to train a potential that can be applied to silicon oxidation in MD simulations to generate defect-free oxide layers. This proved to be quite challenging since the training of GAP involves optimizing a high-dimensional, non-linear

model to accurately reproduce quantum mechanical properties, such as energies and forces, across a wide range of atomic configurations. Successfully training a GAP model demands a deep understanding of both the underlying physics and the machine learning techniques involved. It requires expertise in selecting representative training structures that adequately capture the diverse atomic environments present in the system of interest. The large number of trained GAPs results from the initial uncertainty in predicting the impact of new training structures in the training data set on the behavior of the trained GAP.

There are two different sets of trained GAPs in this work. In the first set, an attempt was made to train a GAP also on datasets including non-defect-free training structures to produce defect-free oxides. For the second group, only defect-free training structures were used for the GAP training. It happened that none of the GAPs could be trained to consistently produce oxides free of defects on all different kinds of structures. The reason for that lies in the large variety of interatomic interactions for which the GAP needs to be trained. The trained GAP should be able to map the atomic interactions accurately as for crystalline Si, amorphous SiO_2, and gaseous oxygen [16].

The two lists in the Fig.s 7.3 and 7.4 provide an overview of the two different sets of trained GAPs. They indicate which training data was used and which settings were selected for the individual *gap_fit* parameters. The nomenclature is as follows: A capital *A* denotes those GAPs that were also trained on non-defect-free structures. These are numbered consecutively if additional structures were added to the training dataset for the GAP training (i.e. *A1*, *A2*, etc). Sub-variants are labeled with small letters starting with *a* (e.g. *A1a*, *A9b*, etc). This labeling indicates that the GAP was trained on the same training structures, but differs from the GAP with the same name in other settings (e.g. modified cutoff radii, different σ values, etc). All GAPs beginning with *A* are summarised in the table in Fig. 7.3. GAPs that were only trained on defect-free structures are labeled with a capital *B*. Otherwise, the nomenclature follows that of the *A*-GAPs. This GAP group is tabulated in Fig. 7.4. In both figures, the GAP names are listed in the left-hand column. The structures are listed in the other columns. The number of structures used for the training dataset is given in the rows of the GAPs. The following abbreviations are used: *NW* stands for nanowire, *cellopt* means cell optimization and *SP* single-point calculation, *geoopt* means geometry optimization, *surf.* stands for surface and *AL* stands for active-learning, *oxid.* means oxidized (which indicates oxidized structures), *AIMD* means *ab-initio* molecular calculation and *def.free* stands for defect-free.

In Fig. 7.3, *various small surfaces* is a summation of the following structures: bulk silicon, bulk silicon dioxide, clean silicon surfaces, and oxidized silicon surfaces with a cell size of 15.523×15.523×37.22 Å and 31.046×31.046×60.00 Å. *AL oxid. silicon*

NW are oxidized nanowires consisting of 1680 Si atoms, which were oxidized with GAP A4 in MD simulations using LAMMPS. *AL oxid. silicon surfaces* are silicon surfaces with a cell size of 31.046×31.046×60.00 Å, which were oxidized via MD simulations using GAP A5. The last 3 columns in Fig. 7.3 indicate whether a 3-body descriptor, virial stress values, or sigma values deviate from the default values used for the individual GAP training.

In Fig. 7.4, *AL oxid. Surface 1 - 4* denotes oxidized silicon surfaces with a cell size of 15.523×15.523×60.00 Å. *AL oxid. Surface 1*-structures were oxidized using GAP B1, *AL oxid. Surface 2*-structures were oxidized using GAP B2, *AL oxid. Surface 3*-structures were oxidized using GAP B4 and *AL oxid. Surface 1*-structures were also oxidized with GAP B4 but with Si_2 molecules in the gas phase during the MD simulations. The last column in Fig. 7.4 indicates if modified cutoff radii are used for the GAP training.

For most of the GAP trainings, the following settings were used. This setting has proven successful in previous GAP trainings [15]. GAPs that deviate from these setting are marked with an x in the last column(s):

- Single atoms: σ_{energy}=0.0001, σ_{forces}=0.001, $\sigma_{virials}$=0.001
- Dimers: σ_{energy}=0.01, σ_{forces}=0.1, $\sigma_{virials}$=0.1
- Other structures: σ_{energy}=0.002, σ_{forces}=0.02 , $\sigma_{virials}$=0.02
- Cutoff radius 2-body descriptor: $r_{cutoff\_Nb=2}$=4 Å
- Cutoff radius 3-body descriptor: $r_{cutoff\_Nb=3}$=3 Å
- Cutoff radius SOAP: r_{cutoff_SOAP}=5 Å

In the next chapter, we will discuss the individual performances of these trained GAPs. It will be shown, that the performance highly depends on the selected types of trainings structures and their quantity in the training data set.

GAP name	single + dimers	various small surfaces	clean Si NW (576 atoms, cellopt)	clean Si NW (1680 atoms, cellopt)	oxid. silicon NW (SP)	oxid. silicon surf. (gcopt)	AL oxid. silicon NW	oxid. silicon NW (gcopt)	AL oxid. silicon surface	oxygen gas (AIMD)	oxygen gas artifacts	surfaces with oxygen artifacts	3-body descriptor	virials	modified sigma values
A1	232	837	99	101	90										
A1a	232	837	99	101	90									x	
A2	232	837	99	101	90										
A2a	232	837	99	101	90									x	
A3	232	837	99	101	90										
A4	232	837	99	101	90	9									
A5	232	837	99	101	90	9	85								
A6	232	837	99	101	90	9	85	180		51					
A7	232	837	99	101	90	9	85		100						
A7a	232	837	99	101	90	9	85		100						x
A7b	232	837	99	101	90	9	85		100				x		x
A7c	232	837	99	101	90	9	85		100						x
A7d	232	837	99	101	90	9	85		100				x		x
A8	232	837	99	101	90	9	85	180	100						
A9	232	837	99	101	90	9	85	180	100		32				x
A9a	232	837	99	101	90	9	85	180	100		32				x
A9b	232	837	99	101	90	9	85	180	100		32		x		x
A10	232	837	99	101	90	9	85		100		32				x
A11	232	837	99	101	90	9	85		100			15			x
A11a	232	837	99	101	90	9	85		100			29			x
A12	232	837	99	101	90							15		x	
A12a	232	837	99	101	90							29		x	

Figure 7.3: Overview of trained GAPs, which also have non-defect-free structures in the training datasets, labeled with a capital A. The GAP names are shown in the left-hand column. The other columns show the number of respective structures as well as modifications in GAP training in the last 3 columns.

GAP name	single + dimers	bulk silicon	bulk silicon dioxide	clean silicon surface	def.-free oxid. surf. (AIMD) 1	surfaces with oxygen artifacts	clean Si NW (576 atoms, cellopt)	clean Si NW (1680 atoms, cellopt)	oxid. silicon NW (geoopt)	def.-free oxid. surf. (AIMD) 2	def.-free oxid. surf. (SP)	AL oxid. Surface 1	AL oxid. Surface 2	AL oxid. Surface 3	AL oxid. Surface 4	oxygen gas	virials	modified sigma values	modified cutoffs
B1	232	90	70	50	184														
B1a	232	90	70	50	184												x		
B2	232	90	70	50	184							49							
B3	232	90	70	50	184							49	87						
B3a	232	90	70	50	184							49	40						
B4	232	90	70	50	184						49	49	40						
B5	232	90	70	50	184						49	49	40	67					
B5a	232	90	70	50	184						49	49	40	137					
B5b	232	90	70	50	184						49	49	40	183					
B6	232	90	70	50	184						49	49	40		6				
B6a	232	90	70	50	184						49	49	40		15				
B7	232	90	70	50	184						49	49	40	183	6				
B7a	232	90	70	50	184						49	49	40	183	6		x		
B7b	232	90	70	50	184						49	49	40	183	15				
B8	232	90	70	50	184	29													x
B8a	232	90	70	50	184	29													x
B8b	232	90	70	50	184	29													x
B8c	232	90	70	50	184	29												x	
B9	232	90	70	50	184	29	50	50											x
B10	232	90	70	50	184	29	50	50	9										x
B11	232	90	70	50	184					22									
B12	232	90	70	50	184	29	50	50	9	22									
B13	232	90	70	50	184						49								
B14	232	90	70	50	184					22	49								
B15	232	90	70	50	184	29	50	50	9	22	49								
B16	232	90	70	50	184											143			
B17	232	90	70	50	184	29	50	50	9							143			

Figure 7.4: Overview of trained GAPs, which only have defect-free structures in their training datasets. These GAPs are labeled with a capital B. In the first row, the names of the GAPs are listed, followed by the number of the number of respective structures. The last three rows indicate modifications in the GAP training.

8 GAP Testing

The reliability of a trained GAP is directly related to its ability to predict the chemical and physical properties of the structures or combinations for which it has been trained. To evaluate the accuracy of a GAP, it is commonly compared with DFT data by testing it on a set of test structures. For this thesis, the test structures are taken from the training structures, which are left aside and not included in the training to serve as an independent benchmark.

The performance of a trained GAP is evaluated using statistical measures such as the mean deviation between the GAP predictions and the DFT data of the test structures. These metrics provide insights into the accuracy and reliability of the force field and help to identify and address potential weaknesses. Using this test data, the training datasets can be optimized for the new GAP training. However, the major challenge is to predict how a change in the training dataset will affect the accuracy of the GAP.

In this thesis, two methods are used to verify the accuracy and performance of the trained potentials. On the one hand, tests were carried out against selected test structures. In some cases, the test structures differ between those GAPs that were trained exclusively on defect-free structures and those GAPs that also contained defective structures in the training dataset. A description of the test structures follows in section 8.1. On the other hand, the GAPs that were trained on exclusively defect-free structures were also verified using MD simulations. Two MD simulations were started with the trained GAP in LAMMPS. In one MD simulation, a silicone surface was oxidized. In the second MD simulation, the behavior of O_2 in the gas phase was investigated. Both MD simulations are described in more detail in section 8.2.2.

8.1 Test structures

8.1.1 Bulk SiO_2

The bulk SiO_2 structures were chosen to verify if the trained GAP accurately reproduces silicon dioxide. Like the training structures, the test structures consist of regularly arranged SiO_2 structures with a total of 216 atoms (see Fig. 6.6b). Using MD simulations, a total of 25 bulk SiO_2 test structures with a cell size of $15.201 \times 15.201 \times 14.332$ Å have been created based on one geometry-optimized bulk

SiO_2 structure [39]. The total energy and the forces acting on the atoms were obtained by using single-point calculations.

8.1.2 Clean silicon nanowires

The oxidation of silicon nanowires is one of the trained GAP's main tasks. The GAP was tested against two different clean nanowires to make sure it accurately replicates the chemical and physical behavior of clean, non-oxidized nanowires:

- Testing against 101 structures consisting of 576 Si atoms (Fig. 6.7a).
- Testing against 101 structures consisting of 1680 Si atoms (Fig. 6.7b).

All structures and data were created using MD simulations in LAMMPS and subsequent DFT cell optimizations.

8.1.3 Oxidized silicon nanowires

To ensure the performance of GAP concerning oxidized nanowires, tests were also carried out against such structures. Therefore 20 oxidized nanowires, each consisting of a total of 2063 atoms (Fig. 6.8b), were selected for this purpose. The test data of the structures originating from MD simulation were generated by using DFT.

8.1.4 Active-learning testing structures

Tests were also conducted against two different active-learning structures that were oxidized utilizing a previously trained GAP. The test data was generated using MD simulations and single-point calculations. The following structures were selected for testing:

- 200 structures of oxidized silicon surfaces with a cell size of $31.046 \times 31.046 \times 60.00$ Å, which were oxidized in MD simulations using GAP *A5* (Fig. 6.4a).
- 201 structures of oxidized silicon nanowires consisting of 1680 silicon atoms (similar to Fig. 6.8b). These structures were oxidized using also GAP *A5* in the MD simulations.

8.1.5 Oxygen gas

Additionally, the trained GAP was evaluated against two pure oxygen structures to ensure it accurately represents oxygen behavior. These two structures, whose data were generated using MD simulations and single-point calculations, are as follows:

- 4 O_2 molecules in a $10 \times 10 \times 10$ Å box, of which a total of 21 test data were created (Fig. 6.9a).

- 16 O_2 molecules in a 20×20×20 Å box. Of these, 21 data were also created. This test structure looks similar to the other oxygen test structure, except that now 4 times as many O_2 molecules are distributed over an 8 times larger simulation cell.

8.1.6 Defect-free oxidized surfaces

The potentials that are trained exclusively with defect-free training structures were also tested against defect-free structures. Various defect-free structures were selected for testing, which are:

- Defect-free oxidized surfaces from AIMD calculations with a cell size of 15.523× 15.523×37.22 Å. Test data from a total of 395 different structures were created. These structures have varying oxide layer thicknesses and consist of a total of 228 to 323 atoms.
- Defect-free oxidized surfaces from MD calculations, which possess the same cell size of 15.523×15.523×37.22 Å. These structures consist of between 232 and 233 atoms (Fig. 6.6a). The test data, which originate from single-point calculations, were taken from a total of 189 structures.

8.1.7 Summary of the testing data

All test structures are summarized in Tab. 8.1. The first column contains the name of the structure and the second column contains the number of atoms that make up the structure or the configuration. The third and fourth columns contain the number of test structures.

The testing data includes nanowires (NW) and surfaces (surf). They can be defect-free (def.-free), partly oxidized (oxid) and trained by active-learning (AL). Both active-learning configurations, the oxidized silicon surfaces (oxid. surface) and the oxidized nanowires (oxid. NW), were oxidized using GAP *A5* as force field. Additionally, also defect-free and oxidized silicon surfaces from AIMD calculations (AIMD) and from MD and single-point calculations (MD) are contained in the testing data. Both possess a cell size of 15.523×15.523×60.00 Å.

8.2 Testing

8.2.1 Testing against test structures

The trained GAPs are applied to the test structures to calculate the energies and forces. These results are compared with the values from the DFT calculations by

structure	atoms, configuration	number of struct.	$\sum$
bulk SiO_2	216	25	25
clean silicon NW	576	101	202
	1680	101	
oxid. silicon NW	2063	20	20
AL oxid. surface	1223	200	401
AL oxid. NW	2413	201	
oxygen gas	8	21	42
	32	21	
def.-free oxid. surf.	228-323 for AIMD	395	584
	232-233 for MD	189	
			1274

Table 8.1: Overview of all testing structures

using the mean absolute error (MAE) to quantify the agreement between the GAP and DFT. The MAE values for the energy are given in units of meV/atom, and the MAE values of the forces in units of eV/Å. Based on these values, appropriate adjustments can now be made to the new training dataset to increase the accuracy of the new GAP. However, experience and sensitivity are required when creating the new training dataset. It is also important to have a critical point of view on the extent to which the test structures allow statements to be made about the accuracy of the trained GAP. The measured MAE values prevent an accurate statement about the behavior of the trained GAP with other structures if the test is conducted against an insufficient number of test structures.

The tables in Fig. 8.1 and Fig. 8.2 list the calculated MAE values for the trained GAPs, which were also trained on non-defect-free structures. These potentials were not tested against defect-free test structures. In Fig. 8.3 and Fig. 8.4 the MAE values of those GAPs trained on exclusively defect-free structures are tabulated. The MAE values for the energy and forces for the respective test structure are listed in separate columns in both figures.

8.2.2 Testing with MD simulations

The main aim of the trained GAP is to generate defect-free oxidized silicon structures. However, it is difficult to tell from the MAE values whether the oxidation of the silicon structures in the MD simulations produces defect-free oxides. For this reason, two additional MD simulations were carried out with those GAPs that were trained on defect-free structures for verification.

GAP name	bulk silicon dioxide		clean silicon NW (576 atoms)		clean silicon NW (1680 atoms)		oxidized silicon NW	
	energy	forces	energy	forces	energy	forces	energy	forces
A1	2.21	0.28	7.91	0.13	1.8	0.08	52.49	0.33
A1a	2.59	0.28	8.49	0.13	2.34	0.08	37.99	0.32
A2	54.7	1.12	7.82	0.14	1.62	0.1	34.81	0.38
A2a	29.19	0.93	8.91	0.14	2.49	0.1	33.44	0.34
A3	2.24	0.28	7.91	0.13	1.81	0.08	47.69	0.31
A4	4.28	0.32	7.95	0.13	1.82	0.08	29.75	0.35
A5	11.17	0.61	26.13	0.14	4.94	0.09	56.53	0.37
A6	187.9	0.69	62.1	0.17	7.72	0.12	26.67	0.34
A7	68.82	0.52	32.6	0.14	5.69	0.1	26.6	0.42
A7a	98.42	0.41	37.95	0.16	8.43	0.11	9.42	0.53
A7b	152.4	0.47	10.8	0.15	8.38	0.11	27.81	0.54
A7c	127.1	0.48	27.78	0.16	10.32	0.11	3.63	0.51
A7d	53.6	0.95	28.69	0.18	2.83	0.12	70.6	0.74
A8	119.6	0.72	62.05	0.16	9.8	0.12	19.76	0.34
A9	527.3	11.37	37.87	0.18	12.17	0.12	418.39	2.54
A9a	587.8	13.3	44.24	0.17	9.5	0.12	497.42	2.94
A9b	166.6	5.75	19.91	0.18	11.97	0.12	224.36	1.35
A10	121.6	0.45	45.13	0.16	9.27	0.11	66.5	0.52
A11	125.1	1.3	89.95	0.26	20.12	0.17	27.45	1.1
A11a	134.5	0.46	25.57	0.16	7.31	0.11	46.33	0.49
A12	2.74	0.28	8.55	0.13	2.33	0.08	39.54	0.31
A12a	2.06	0.28	8.38	0.13	2.43	0.08	50.32	0.32

Figure 8.1: Part I of the MAE values for the energy (in units of meV/atom) and the forces (in units of eV/Å) against the test structures for the trained GAPs which also have non-defect-free structures in the training datasets. The GAP names are listed in the left-hand column, and the test structures in the first line. The MAE values for the energy and the forces are written next to each other in separate columns for each test structure. GAP *A12a*, which performs best when averaged over all test structures, is marked in blue.

GAP name	Al oxidized surface		Al oxidized NW		Oxygen gas (8 atoms)		Oxygen gas (32 atoms)	
	energy	forces	energy	forces	energy	forces	energy	forces
A1	25.22	0.3	19.69	0.28	132.34	0.93	95.13	0.84
A1a	17.94	0.3	21.71	0.28	53.53	0.96	15.87	0.85
A2	186.92	0.55	175.16	0.57	2151	3.84	2242	3.66
A2a	173.28	0.48	148.37	0.51	99.25	2.51	146.21	2.34
A3	23.68	0.3	19.23	0.29	274.65	1.01	235.05	0.99
A4	29.58	0.26	34.98	0.27	421.6	1.41	405.41	1.46
A5	113.58	0.35	136.98	0.37	1008	1.99	1008	2.01
A6	72.89	0.38	105.53	0.39	654.73	1.19	614.77	1.08
A7	38.53	0.28	71.67	0.35	1654	1.78	1669	1.79
A7a	2.93	0.22	16.48	0.31	695.58	0.94	692.34	1.02
A7b	4.71	0.24	14.13	0.33	537.58	1.34	552.46	1.38
A7c	3.89	0.25	32.38	0.33	802.82	1.07	787.55	1.2
A7d	49.37	0.37	67.86	0.45	448.71	1.09	444.74	0.98
A8	60.95	0.36	105.3	0.42	1808	1.91	1853	1.83
A9	1967	4.89	2022	5.39	10676	10.8	11868	6.8
A9a	2312	5.72	2368	6.3	12815	12.41	14203	7.6
A9b	1010	2.49	1069	2.74	6143	5.66	6706	3.85
A10	3.33	0.23	20.1	0.31	33.86	1.04	28.82	1.14
A11	189.72	0.65	77.73	0.69	533.47	3.76	437	3.46
A11a	3.71	0.23	43.18	0.32	1436	0.86	1478	0.67
A12	18.06	0.3	22.67	0.28	606.75	1.12	619.55	1.27
A12a	17.4	0.27	20.72	0.27	183.75	1	125.76	0.86

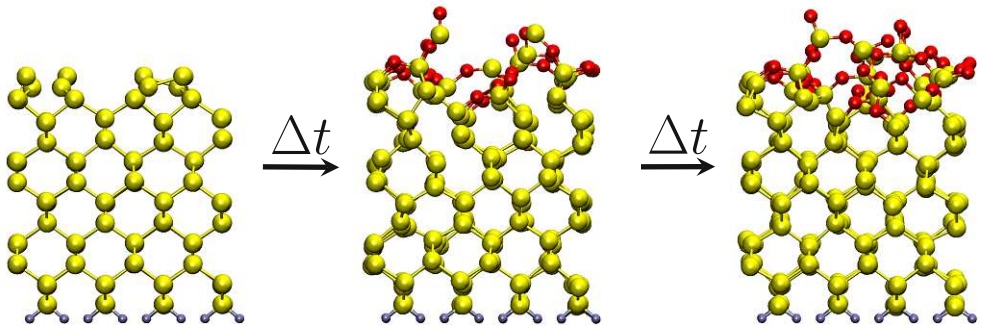
Figure 8.2: Part II of the MAE values for the energy (in units of meV/atom) and the forces (in units of eV/Å) for the trained GAPs which were also trained on non-defect-free structures. The left-hand column lists the GAP names and the test structures are written in the first line. MAE values for the energy and the forces are in separate columns for each test structure. *GAP12a* is also highlighted in color again.

GAP name	bulk silicon dioxide		clean silicon NW (576 atoms)		clean silicon NW (1680 atoms)		oxidized silicon NW		AL oxidized surface	
	energy	forces	energy	forces	energy	forces	energy	forces	energy	forces
B1	2.26	0.25	110.1	0.17	55.79	0.1	61.8	0.1	1.76	0.25
B1a	2.13	0.25	104.3	0.17	60.49	0.11	17.53	0.4	2.82	0.27
B2	2.37	0.26	15.28	0.17	8.18	0.11	76.96	0.38	2.16	0.24
B3	2.6	0.28	65.07	0.17	33.3	0.11	14.36	0.4	1.85	0.24
B3a	2.51	0.27	58.59	0.16	28.56	0.1	47.79	0.39	2.33	0.24
B4	3.06	0.27	26.5	0.17	12.85	0.1	45.03	0.37	26.62	0.23
B5	3.65	0.28	53.71	0.16	29.4	0.1	16.7	0.39	26.43	0.23
B5a	3.23	0.29	76.38	0.18	36.87	0.11	7.57	0.39	28.35	0.23
B5b	4.34	0.31	44.21	0.18	21.47	0.11	45.48	0.4	22.18	0.24
B6	2.74	0.27	60.83	0.17	32.07	0.11	24	0.39	25.82	0.23
B6a	3.46	0.28	64.04	0.18	26.82	0.11	52.72	0.38	20.74	0.23
B7	4.17	0.3	50	0.17	30.06	0.11	33.74	0.39	24.11	0.23
B7a	5.59	0.3	11.36	0.17	13.22	0.11	53.76	0.4	24.3	0.23
B7b	5.49	0.31	51.32	0.18	26.06	0.11	42.24	0.39	20.34	0.24
B8	2.33	0.26	151.8	0.17	77.34	0.1	53.49	0.43	16.81	0.25
B8a	3.84	0.28	216.3	0.22	118.5	0.12	107	0.42	26.31	0.27
B8b	4.37	0.29	156.8	0.25	95.08	0.13	80.64	0.48	20.97	0.28
B8c	2.08	0.25	126.3	0.17	63.46	0.1	28.83	0.43	7.74	0.26
B9	2.38	0.27	2.57	0.12	1.62	0.07	84.26	0.4	13.57	0.26
B10	5.13	0.33	2.72	0.12	1.74	0.08	90.95	0.41	5.78	0.27
B11	2.86	0.26	112.1	0.16	57.96	0.1	62.94	0.1	2.75	0.25
B12	6.07	0.33	2.65	0.12	1.69	0.08	46.94	0.42	7.87	0.25
B13	2.93	0.25	95.98	0.17	49.78	0.1	40.75	0.41	6.86	0.24
B14	2.92	0.26	102.2	0.17	50.63	0.11	71.67	0.41	2.63	0.23
B15	5.27	0.33	2.73	0.12	1.73	0.08	101.2	0.43	3.21	0.25
B16	2.37	0.26	116	0.17	58.4	0.1	16.32	0.4	11.3	0.25
B17	5.08	0.33	2.75	0.12	1.79	0.08	96.75	0.44	7.07	0.26

Figure 8.3: Part I of the table of the MAE values for the energy (in units of meV/atom) and the forces (in units of eV/Å) which only have defect-free structures in their training datasets. The GAP name is listed in the left column. The MAE values for each test structure are listed in the next columns, whereas the MAE values for energy and forces are presented in separate columns. The names of the test structures are listed in the first line.Here, GAP *B17*, which is highlighted in blue, performs best of all GAPs in terms of the averaged MAE values.

GAP name	AL oxidized NW		Oxygen gas (8 atoms)		Oxygen gas (32 atoms)		def.-free oxidized surface (AIMD)		def.-free oxidized surface (MD)	
	energy	forces	energy	forces	energy	forces	energy	forces	energy	forces
B1	34.68	0.32	62.78	1.2	135.1	0.93	8.65	0.16	54.9	0.29
B1a	32.18	0.33	232.5	1.02	233.4	0.9	7.71	0.15	51.46	0.31
B2	3.52	0.31	504.9	1.04	433	0.89	6.6	0.16	47.29	0.31
B3	13.18	0.3	225.6	0.24	272.7	0.11	8.11	0.16	53.14	0.27
B3a	14.82	0.31	226.9	0.28	249.7	0.13	8.14	0.16	51.93	0.28
B4	19.42	0.29	267.1	0.27	272.6	0.14	5.68	0.18	8.09	0.22
B5	32.79	0.28	210.5	0.17	266.4	0.1	5.44	0.18	9.05	0.23
B5a	53.03	0.29	194.3	0.16	201.8	0.08	6	0.19	9.23	0.23
B5b	37.83	0.28	186.3	0.12	165.1	0.05	6.05	0.19	12.27	0.23
B6	33.18	0.29	250.1	0.24	216.8	0.14	8.12	0.18	8.15	0.22
B6a	34.75	0.28	185.3	0.25	172.6	0.17	4.17	0.15	8.45	0.22
B7	41.29	0.28	195.3	0.12	184.3	0.05	4.65	0.19	11.75	0.23
B7a	25.22	0.28	221	0.12	206.1	0.05	3.17	0.19	14.83	0.23
B7b	39.8	0.27	214.5	0.11	203	0.07	4.77	0.16	11.43	0.23
B8	38.19	0.3	703.4	0.67	578.8	0.34	8.23	0.16	50.36	0.32
B8a	63.33	0.31	616.7	0.5	348.6	0.59	9.06	0.16	62.48	0.29
B8b	46.9	0.33	382.2	1.28	288.1	1.12	9.75	0.16	51.22	0.29
B8c	29.54	0.31	633.4	0.65	674.3	0.67	8.69	0.15	57.96	0.31
B9	11.11	0.3	414.5	0.47	485.2	0.23	9.43	0.16	59	0.3
B10	24.45	0.3	666.7	0.74	417.2	0.32	8.14	0.16	48.32	0.3
B11	31.77	0.31	283.3	0.98	278.8	0.75	4.8	0.15	13.97	0.25
B12	22.17	0.3	364.9	0.76	408	0.64	4.51	0.16	18.07	0.26
B13	10.08	0.3	224	1.19	76.02	0.96	3.74	0.15	4.56	0.22
B14	17.95	0.29	126.7	1	88.63	0.92	4.19	0.15	3.77	0.22
B15	18.96	0.29	733.3	0.49	601.9	0.43	4.7	0.16	4.45	0.25
B16	27.44	0.33	13.19	0.17	27.4	0.24	8.87	0.16	65.63	0.3
B17	28.17	0.31	28.45	0.25	51.8	0.16	9.05	0.17	52.53	0.31

Figure 8.4: Part II of the table of the MAE values for the energy (in units of meV/atom) and the forces (in units of eV/Å) which only have defect-free structures in their training datasets. The GAP name can be found in the left column. The MAE values for each test structure are listed in the next columns, whereas the MAE values for energy and forces are presented in separate columns. The names of the test structures are listed in the first line. As in Part I, GAP *B17* is highlighted in blue.



(a) Before oxidation (b) After oxidation cycle 2 (c) After oxidation cycle 13

Figure 8.5: Oxidation process of a silicon surface using GAP *B4* for the MD simulations. A total of 15 oxidation cycles were carried out, with the oxygen in the vacuum being renewed after each cycle. Panel (a) shows the clean silicon surface before the oxidation process. Panel (b) shows the surface after the second oxidation cycle. The oxidation progresses relatively quickly at this stage. Panel (c) shows the surface after the thirteenth oxidation cycle. Further oxidation becomes more challenging due to the thicker oxidation layer.

In the first MD simulation, a silicon surface with a cell size of 15.523×15.523×60.00 Å is oxidized in 15 MD cycles. Each MD cycle consists of 100.000 steps. After each oxidation cycle, the oxygen atmosphere is renewed so the oxide can continue growing. With each GAP, five silicon surfaces get oxidized in this manner. Fig. 8.5 illustrates this oxidation process on the example of GAP *B4*: The clean silicon surface before the oxidation is displayed in Fig. 8.5a, the silicon surface after the second cycle of oxidation is shown in Fig. 8.5b, and the silicon surface completing the 13^{th} cycle is shown in Fig. 8.5c. After the last oxidation cycle, the dangling bonds in the oxide got passivated with hydrogen atoms. This makes it easy to detect the defects.

In the second MD simulation, the behavior of pure oxygen was investigated with the trained GAP. For this purpose, two similar MD simulations with O_2 molecules were started: One at a pressure of 50 bar and the second with O_2 molecules at a pressure of 150 bar. The simulation cell size for both simulations is 40×40×40 Å, resulting in 116 O_2 molecules in the gas phase for the first simulation and 348 O_2 molecules for the second simulation. The information of the behavior of the O_2 gas with the investigated GAP was used to readjust the type and amount of the training structures for the new GAP training.

8.2.3 Testing results

A total of 22 GAPs, which also have non-defect-free structures in the training datasets (A-GAPs), and a total of 27 GAPs, which only had defect-free structures in their training dataset (B-GAPs), were trained. With a look on their MAE values against the different test structures, their is no linear increasing in the performance of the individual GAPs. The changes in the respective training sets also change the performance of the GAPs with respect to the individual test structures. The GAPs that performed best across all test structures averaged over all MAE values are highlighted in color in the respective tables, i.e. GAP *A12a* and GAP *B17*. A more detailed analysis of the performance of these two GAPs is provided in section 9.1, where the energies and the forces between GAP and the DFT values with regard to the different test structures are also shown graphically.

9 Results

In this chapter, the capabilities of selected GAPs are presented and compared with other modelling approaches. The main goal of this thesis is to develop a GAP that simulates the oxidation of silicon with *ab-initio* accuracy. The results will be evaluated using various analyses and the predictions of the trained GAP will be compared with *ab-initio* reference data to assess the agreement and accuracy of the GAP. In addition to the growth rate of the oxide, the structural properties of the formed oxide and the interface quality will be analyzed. Concerning MD simulations, the GAP is also compared with the reactive force field ReaxFF from Ref. [118]. Similar to the results presented within this thesis, a concise version of this work has been published on arXiv [16] and is currently under review in the *Journal of Chemical Physics*.

As shown in chapter 7, the GAP training is an optimization process attempting to obtain an even more accurate GAP by modifying the training data and the training parameters, which reflects the physical and chemical properties of the silicon oxidation process even better. In this context, numerous GAPs were trained and subsequently tested for accuracy. Two of these GAPs (*A12a* and *B17*) proved to be particularly powerful in terms of MAE values across all test structures. In the following, these two ML force fields are analyzed in more detail.

9.1 Comparison to DFT

The test structures, presented in section 8.1, are used to compare GAP with the DFT predicted energies and forces. Here, GAP *A12a*, which was trained on also non-defect-free training structures, and GAP *B17*, which was trained on exclusively defect-free training structures, will be compared with these test structures.

9.1.1 Comparison of GAP A12a with DFT

GAP *A12a* was trained on a total of 1388 training structures, whereby in addition to the energies and forces, also the virials were used to train GAP. The setting of the descriptor parameters is listed on the left-hand side in Tab. 9.1. Here, δ denotes the kernel scaling in eV, r_{cut} is the cutoff radius, and r_Δ is the distance across which kernel is smoothly taken to zero, both given in Å. n_{max} and l_{max} define the number of angular and radial basis functions, respectively and ζ denotes the power the kernel

parameter	SOAP	2-body
δ (eV)	0.4	4
r_{cut} (Å)	4	4
r_Δ (Å)	1	-
n_{max}	8	-
l_{max}	4	-
ζ	4	-

structure	σ_{energy}	σ_{forces}	$\sigma_{virials}$
single atoms	0.0001	0.001	0.001
dimer	0.01	0.1	0.1
default values	0.002	0.02	0.2

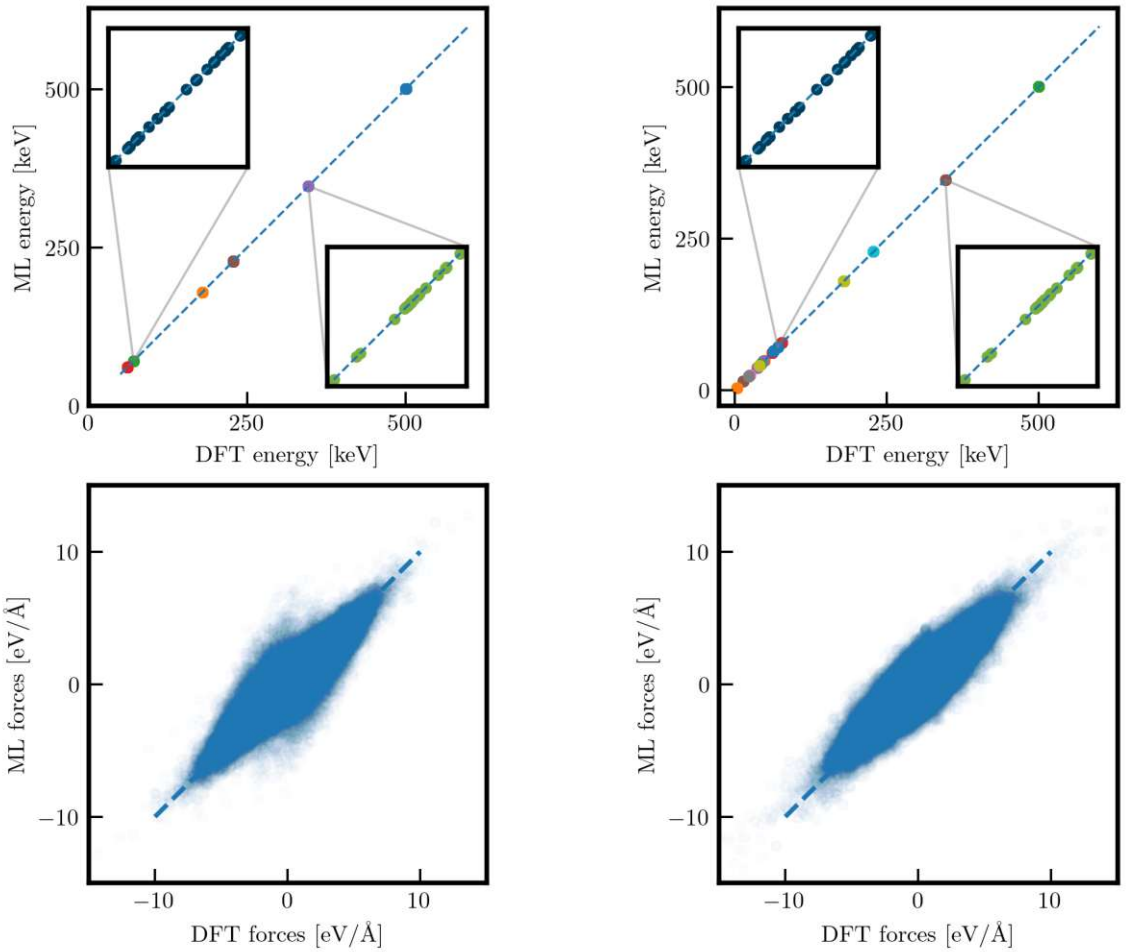
Table 9.1: Setting of the parameters for training GAP *A12a*. The descriptor parameters are listed in the table on the left, where δ denotes the kernel scaling, r_{cut} denotes the cutoff radius, r_Δ denotes the distance across which kernel is smoothly taken to zero, n_{max} and l_{max} define the number of angular and radial basis functions and ζ denotes the power the kernel is raised to. The table on the right lists the sigma values for each subcategory used in the training dataset. σ_{energy} denotes the σ parameter for the energy, σ_{forces} for the forces and $\sigma_{virials}$ for the virials, respectively.

is raised to. The setting of the sigma values for the training of GAP *A12a* is given on the right-hand side of Tab. 9.1. All training structures in the training dataset, except single atoms and dimers, were assigned the same σ values.

GAP *A12a* was tested against all test structures listed in Tab. 8.1, except the defect-free structures, which gives a total of 691 test structures. The result can be seen in Fig. 9.1a, which compares the DFT energies and the DFT forces with the ones calculated with GAP. The calculations for the energy and the forces yielded an averaged MAE of 37.74 meV/atom for the energy and 0.4 eV/Å for the forces. The MAE values especially against the oxygen test-structures have by far the highest value, as the GAP was primarily trained on structures containing silicon or silicon and oxygen together. Without including the oxygen test structures, the averaged MAE value for the energy yields to 16.89 meV/atom and for the forces to 0.23 eV/Å. Dashed blue lines with slope 1 in Fig. 9.1a indicate that the values of the energy (upper panel) and forces (lower panel) indicate a very good correlation throughout the whole range of structures. In the two small windows within the energy plot, the energy values for the bulk SiO_2 structures (blue dots) and for the oxidized NW (green dots) are shown zoomed in as an example.

9.1.2 Comparison of GAP B17 with DFT

GAP *B17* was trained on a total of 907 structures. In contrast to GAP *A12a*, no virials were included in the training but a 3-body descriptor. The settings of the descriptor parameters and for the σ values are listed in Tab. 9.2.



(a) Comparison of GAP *A12a* and DFT

(b) Comparison of GAP *B17* and DFT

Figure 9.1: Panel (a) compares GAP *A12a* and DFT on a total of 691 test structures. The values of the energies (in the upper panel) and the forces (lower panel) show very good agreement throughout the whole range of structures. The two small windows within the energy plot enlarge the energy values for the bulk SiO_2 structures (blue dots) and the oxidized NW (green dots). The blue dotted lines indicate perfect agreement, i.e. a line with slope 1. The average MAE for the energies is 37.74 meV/atom and the average MAE for the forces is 0.4 eV/Å. Panel (b) compares *B17* and DFT. In the upper panel, the MAE of the energies are plotted with a mean MAE value of 25.4 meV/atom. The lower panel shows the MAE of the forces with an average MAE of 0.25 eV/Å. The two graphs in the energy plot show again the energy values of the bulk SiO_2 structures (blue dots) and the oxidized NW (green dots). Compared to GAP *A12a*, GAP *B17* performs better in terms of both energy and forces.

parameter	SOAP	2-body	3-body
δ (eV)	0.4	4	1
r_{cut} (Å)	5	4	3
r_{Δ} (Å)	1	-	-
n_{max}	8	-	-
l_{max}	4	-	-
ζ	4	-	-

structure	σ_{energy}	σ_{forces}	$\sigma_{virials}$
single atoms	0.0001	0.001	0.001
dimer	0.01	0.1	0.1
default values	0.002	0.02	0.2

Table 9.2: Setting of the parameters for training GAP *B17*. Abbreviations are described in the text. The descriptor parameters are listed in the table to the left. In contrast to GAP *A12a*, a 3-body descriptor was also employed to train GAP *B17*. The settings of the sigma values, listed in the table to the right, are the same as for GAP *A12a*.

GAP *B17* was evaluated against all test structures listed in Tab. 8.1, including the defect-free ones. The result can be seen in Fig. 9.1b. When considering all structures, the energy and forces calculations result in an average MAE of 28.34 meV/atom for the energy and 0.24 eV/Å for the forces. Without the oxygen structures, the energy and forces calculations result in an average MAE for the energy of 25.4 meV/atom and the forces 0.25 eV/Å. If only those test structures are considered against which GAP *A12a* was tested, GAP *B17* performs better in terms of both energy and forces: The difference in the MAE values are 16.46 meV/atom for the energy and 0.16 eV/Å for the forces. The two graphs in the energy plot in Fig. 9.1b are in turn an enlarged energy plot of the bulk SiO_2 structures (blue dots) and of the oxidized NW (green dots). GAP *B17* demonstrates an excellent correlation between the DFT and GAP values.

Based on the comparison with DFT values, GAP *A12a* and *B17* demonstrated strong accuracy, indicating their reliability. Given their performance, these GAPs can now be effectively employed as force fields in MD simulations. This approach offers a significant advantage, as the GAP-based MD simulations run much faster than DFT calculations while maintaining a similar level of accuracy. This makes them highly suitable for large-scale simulations where both speed and precision are crucial.

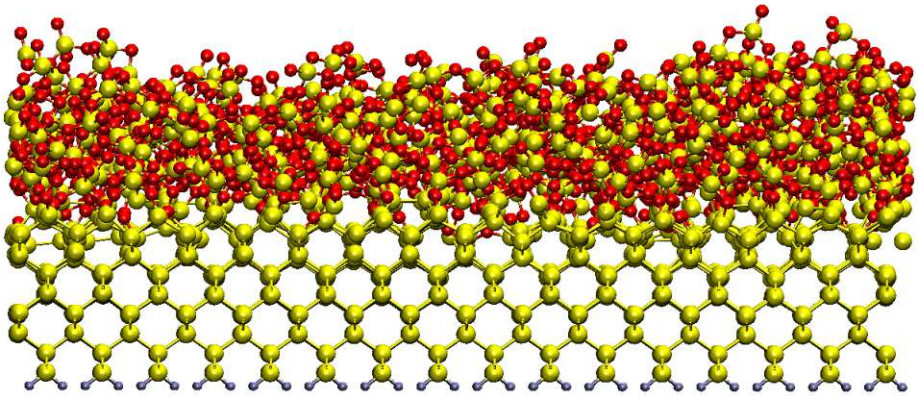


Figure 9.2: Oxidized silicon surface, which contains 5161 atoms and possesses a cell size of 62.092×62.092×60.00 Å, used for analyzing the structural properties of the oxide. The surface was oxidized using GAP *A5* in MD simulations.

9.2 Structural properties

Another means of validating the GAP is to study the structural properties of an oxidized structure that has been oxidized using the trained GAP. This analysis enables conclusions to be drawn regarding the correspondence between the generated oxide layer and the physical reality.

The oxidized silicon surface, which is shown in Fig. 9.2, was analyzed for structure. This surface possesses a cell size of 62.092×62.092×60.00 Å and consists of 5161 atoms. It exhibits an oxide thickness of around 1.2 nm and got oxidized using GAP *A5* in MD simulations. The structural properties of the surface are presented in Fig. 9.3. As can be seen in the upper left histogram, the mean Si–O bond length of the surface agrees with the average Si–O bond length from the literature [119, 120], which is 1.63 Å. Some bonds have a length of more than 1.8 nm. These bonds are limited to the interface between crystalline silicon and silicon oxide, indicating a considerable strain. The upper right histogram shows the angular distribution of the O–Si–O bond angles. The angular distribution reveals the formation of SiO_4 tetrahedrons in the oxide as ideal tetrahedrons exhibit a perfect angle of 109.47° between O–Si–O. Since SiO_4 tetrahedra begin to form at the very beginning of the oxidation process, even ultrathin oxide layers are comparable to bulk SiO_2 in terms of their characteristics [9]. Also, the formation of O–Si–O angles greater than the ideal 109.47° for tetrahedra is consistent with previous observations [121]. The histogram on the bottom left shows the coordination numbers of Si and O. It shows that most of the silicon atoms are coordinated by 4 oxygen atoms. The histogram to the right shows the spatial distribution of the 1-fold coordinated and the 4-fold

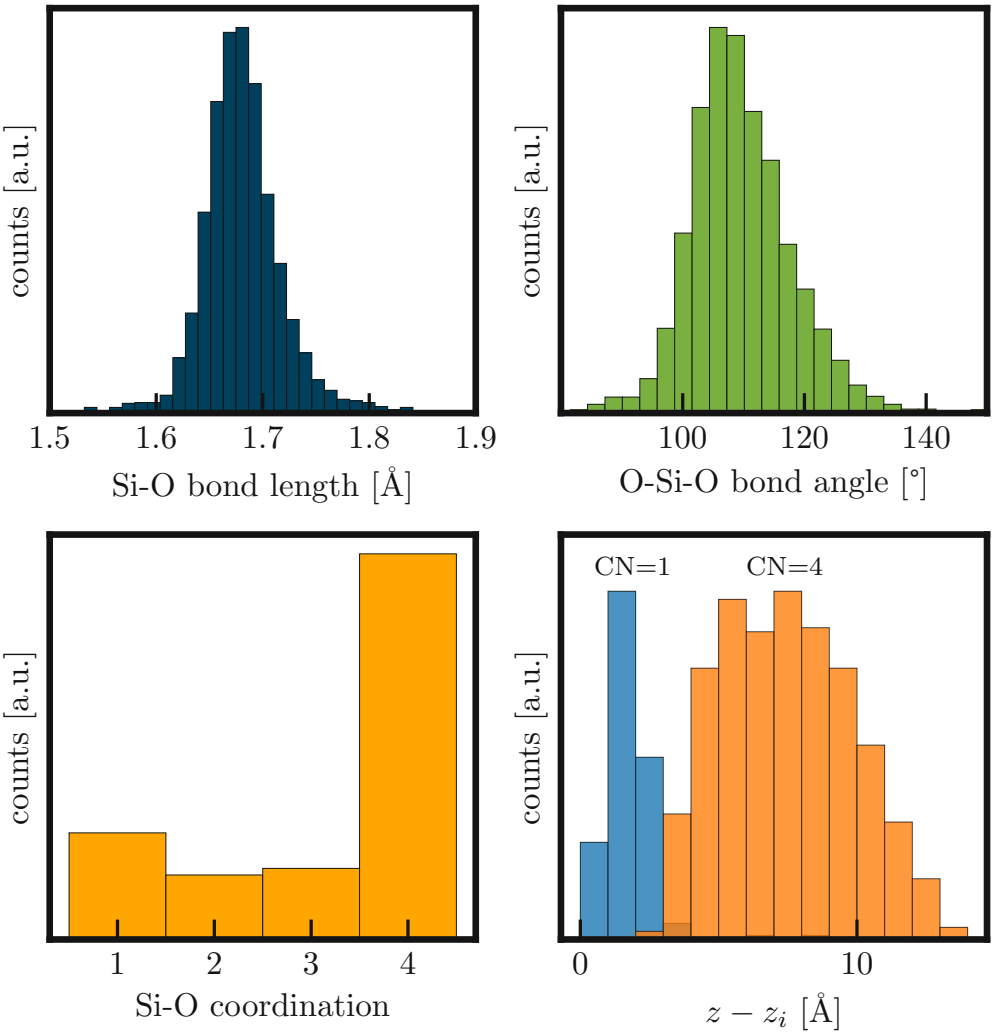


Figure 9.3: The structural properties of an ML oxidized surface with a cell size of 62.092×62.092×60.00 Å and which contains 5161 atoms. Panel on the top left shows the Si–O bond length, which is close to 1.63 Å from the literature [119, 120]. Panel top right: The O–Si–O bond angle distribution is plotted in the upper right histogram. The angles match a tetrahedron's ideal 109.47°. Panel bottom left displays the distribution of coordination numbers. The coordination of the Si atoms appears to range from 1 to 4 O atoms, with most of the Si atoms being coordinated with 4 O atoms. The histogram in the bottom right panel shows the spatial distribution of the Si atoms that are 1-fold coordinated with O (CN=1) and those that are coordinated with 4 O atoms (CN=4). With z_i as the z position of the interface, the lower coordinated Si atoms are located near the interface. Figure adaped from [16].

coordinated Si atoms, with z_i as the z-position of the interface. To evaluate z_i, the five lowest oxygen atoms' z-positions are averaged. Si atoms with one single oxygen coordination (CN = 1) are located primarily near the interface (z_i) between bulk Si and oxide, whereas the 4-fold coordinated silicon atoms (CN = 4) are located in the SiO_4 tetrahedra in the oxide. Furthermore, the oxide layer density is approximately 2.5 g / cm^3, which is in line with the experimental values for amorphous SiO_2 [119]. Furthermore, these results are also consistent with electron-energy-loss spectroscopy (EELS) [122, 123], photoemission studies [119, 124] and transmission electron microscope (TEM) images [119, 125].

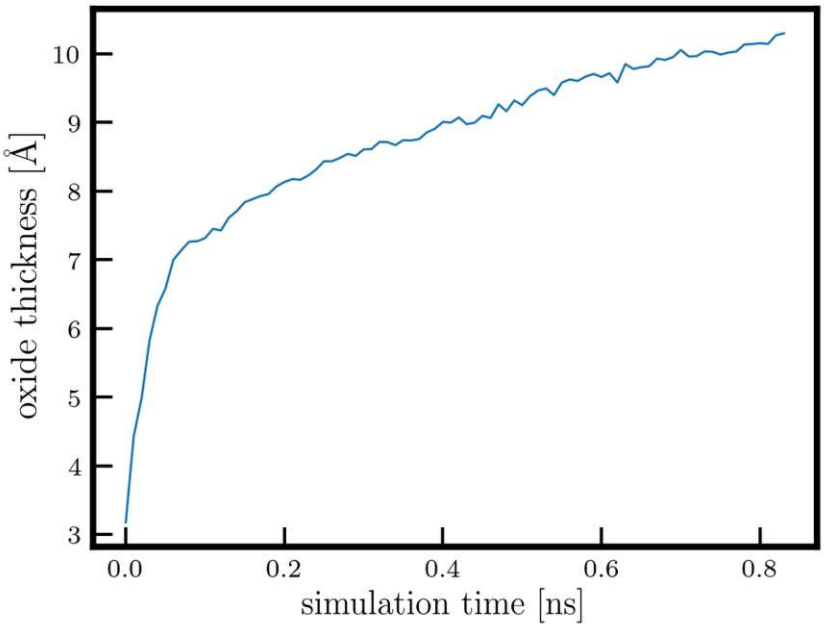


Figure 9.4: The averaged oxide growth kinetics of silicon surfaces with a cell size of 31.046×31.046×60.00 Å. Due to chemisorption, the initial oxidation process is rather fast with an oxidation growth rate around 75 Å/ns. As soon as the first SiO_2 layer has formed, the diffusion of oxygen through the oxide limits the further oxide growth and the growth rate decreases to around 4.2 Å/ns. Figure adaped from [16].

9.3 Oxide growth kinetics

The oxidation growth of silicon was analyzed using clean silicon surfaces with a cell size of 31.046×31.046×60.00 Å, which are exposed to O_2 gas. For greater significance, a total of 8 of these structures were oxidized. The layer growth was then averaged

over all structures. To specify a representative layer thickness, the average z-position of the five lowest oxygen atoms at the interface, z_i, and the average z-position of the five highest oxygen atoms of the oxide surface, z_s, were determined. The difference between these two mean values gives then the oxide thickness $d = z_s - z_i$. The result of this oxidation growth process is shown in Fig. 9.4, where the oxide layer growth is plotted over time.

The oxidation of the clean surfaces was carried out using GAP *A4* in MD simulations, whereby the surfaces were exposed to oxygen at a gas pressure of 50 bar. The oxygen gas was renewed every 10.000 time steps in the MD simulation so that new O_2 was always available for further oxidation. The pressure of the oxygen gas of 50 bar was chosen to keep the simulation time for complete oxidation within reasonable limits. Due to the dynamic oxidation process, the oxide does not grow at the same rate at each point on the Si surface, which leads to a conditional surface roughness. As already mentioned in section 2.3.1, the oxidation rate decreases with the thickness of the oxide. This behavior is consistent with the theoretical work on AIMD calculations [27] and was also confirmed experimentally [9]. The first oxidation stage can be characterized by Chemisorption, in which O_2 molecules dissociate and are adsorbed on the silicon surface. Consequently, the oxide layer grows rather rapidly at first, as can be observed in Fig. 9.4. The growth rate at this stage is around 75 Å/ns and is limited only by the number of O_2 molecules interacting with silicon. As soon as the first SiO_2 layer has formed, the growth rate decreases to around 4.2,Å/ns, as further layer growth requires oxygen to first diffuse through the oxide layer already formed. As can be seen in Fig. 9.4, the trained GAP is also able to reproduce this stage of the oxidation process correctly.

9.4 Surface and Interface roughness

The oxidation of silicon is accompanied by considerable surface roughness, which is confirmed by numerous experiments [28, 126]. Due to the random O_2 adsorption trajectories, layer growth on the Si surface does not occur simultaneously at all locations. During the initial oxidation phase, the roughness of the oxide layer increases with thickness, but stabilizes once the layer reaches 10 nm. At this stage, the oxidation rate decreases, becoming controlled by oxygen diffusion [28], as described in the Deal-Grove model [18], rather than by surface reactions of oxygen, which drive the faster oxidation observed in the early stages [9].

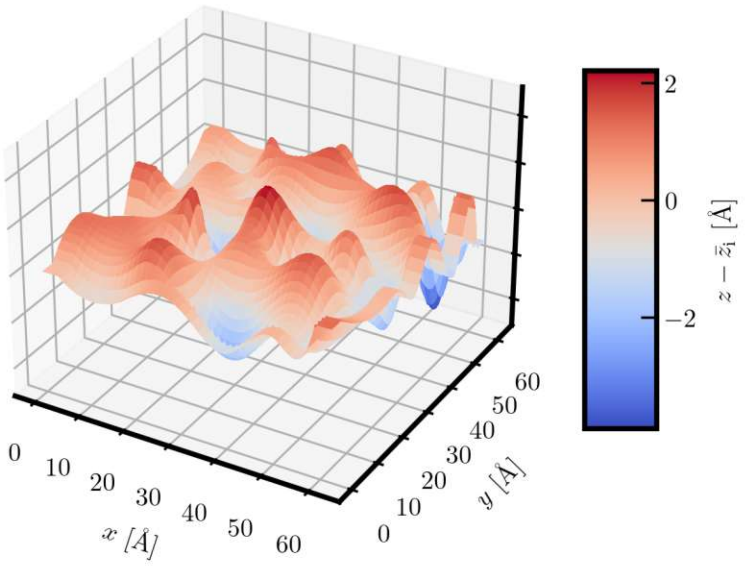


Figure 9.5: Interface roughness of an oxidized surface possessing a cell size of 62.092×62.092×60.00 Å. The roughness results from the random adsorption trajectories of the dynamic oxidation process. The interface has an RMS roughness of R_{RMS}=0.73 Å. The average height of the interface is $\bar{z}_i$=13.38 Å. Figure adaped from [16].

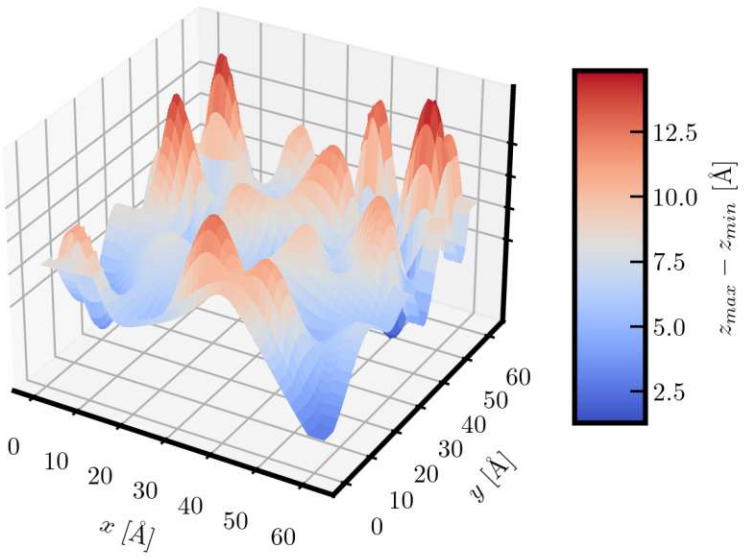
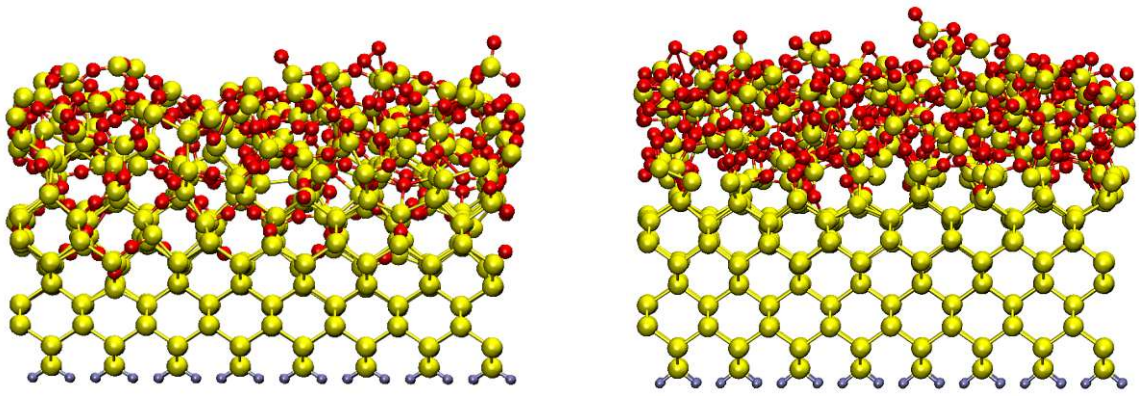


Figure 9.6: Surface roughness of an oxidized surface with a cell size of 62.092×62.092×60.00 Å. The oxide thickness ranges from 2.94 Å to 11.77 Å.

In the following, the interface roughness and the oxide roughness of an oxidized silicon surface are analyzed. This oxidized structure, which is similar to the structure presented in Fig. 9.2, consists of 4846 atoms and possesses a cell size of 62.092×62.092×60.00 Å. The interface roughness of the structure resolved in the in-plane directions is shown in Fig. 9.6. The deviation of the z-position from the lowest oxygen atom, z, to the average interface location of all lowest oxygen atoms, z_i, is used to express the roughness in color for each sub-segment. $R_{mean} = 0.56$ Å is the mean deviation, while $R_{RMS} = 0.73$ Å is the root mean square deviation. This result is in reasonable agreement with the observed values published in [28]. The highest absolute deviation is about 2.17 Å.



(a) Si surface oxidized using ReaxFF as force field

(b) Si surface oxidized using GAP as force field

Figure 9.7: Comparison between the growth kinetics of ReaxFF and GAP as force field in MD simulations. Both surfaces underwent 142 oxidation cycles. Panel (a) shows the oxidized surface with ReaxFF as force field (consisting of 1198 atoms), panel (b) shows the oxidized surface, which was oxidized using GAP as force field (consisting of 1263 atoms). It is evident that there is a significant difference in the growth kinetics between these two force fields. When using ReaxFF there are 15% less oxygen molecules dissolved on the surface. ReaxFF tends to overestimate oxygen diffusion, resulting in a lower density of oxygen atoms distributed throughout the silicon atoms in the crystal. Consequently, this leads to an expanded interface with a lower-than-anticipated Si–O coordination, as illustrated in Fig. 9.9.

The roughness of the oxide is shown in Fig. 9.6 in the in-plane direction. For each surface section, the oxide's thickness is shown in color and the difference in the z-positions between the lowest (z_{min}) and highest (z_{max}) oxygen atoms is used to calculate the thickness $d = z_{max} - z_{min}$ per segment. As the oxide does not grow at

the same rate everywhere, the oxide thickness varies between 2.94 Å and 11.77 Å.

9.5 Comparison between GAP and ReaxFF

ReaxFF is a widely used force field in MD simulations, described in section 4.1.2. The trained GAP and ReaxFF were compared using two identical MD simulations. Except for the force field, all parameters remained the same. In both simulations, a clean Si surface with a cell size of 31.046×31.046×60.00 Å underwent 142 oxidation cycles. After every 10.000 timesteps, which is one oxidation cycle, the oxygen in the gas phase was renewed, bringing the gas pressure back to 50 bar. Fig. 9.7 shows both oxidized surfaces after 142 cycles. On the left is the surface oxidized with ReaxFF and on the right the surface oxidized with GAP *A5*.

There are a total of 1198 atoms in the ReaxFF-oxidized structure and 1263 atoms in the GAP-oxidized structure. As the number of Si and H atoms is the same in both structures, the oxide in the GAP-oxidized structure consists of additionally 65 oxygen atoms. It is also noticeable in Fig. 9.7 that, in comparison to the GAP oxidized structure, the oxygen diffuses considerably deeper into the surface in the ReaxFF oxidized structure. In Fig. 9.8 the structural properties of the GAP oxidized surfaces are given. These are very similar to the structural properties of the oxidized surfaces presented in section 9.2: The mean bond length is close to the 1.63 Å from the literature [119, 120], although some longer bond lengths are also existing due to considerable strain in the interface area. As can be seen in the top right histogram, the mean O–Si–O bond angle matches the ideal 109.47° of the SiO_4 tetrahedron very well. The bottom left histogram illustrates that most Si atoms are 4-fold coordinated with O atoms. There are also 1-fold coordinated Si atoms existing, which are located exclusively in the interface area between the Si/SiO_2 (bottom right histogram).

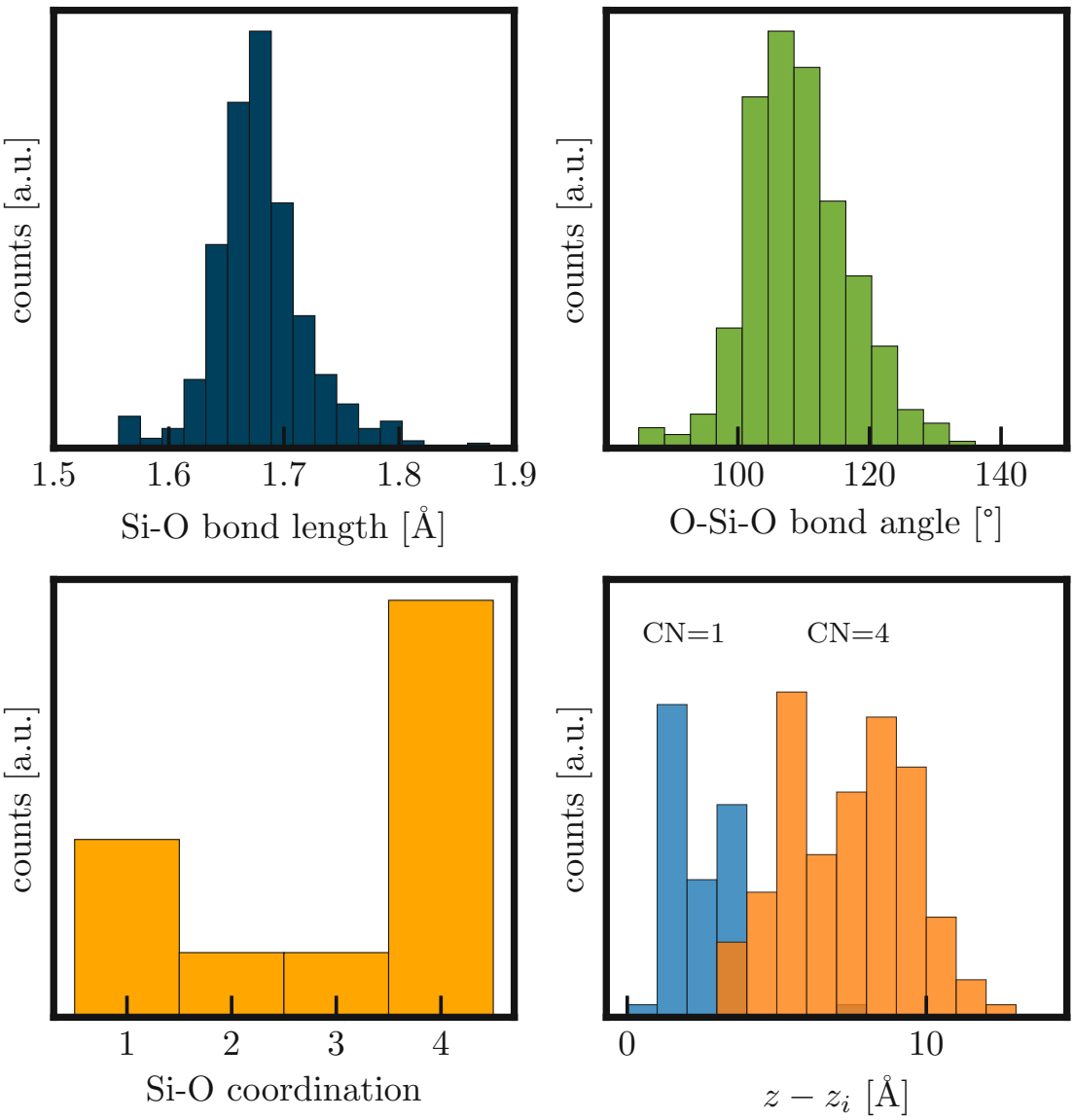


Figure 9.8: Structural properties of an oxidized surface with a cell size of 31.046×31.046×60.00 Å. This structure was oxidized using GAP *A5*. The properties are comparable to the structure given in Fig. 9.2: The mean Si–O bond length (histogram on the top left) is close to 1.63 Å, the mean O–Si–O bond angle (histogram on the top right) matches the tetrahedron's ideal 109.47°, most of the Si atoms in the oxide are coordinated with 4 oxygen atoms (histogram on the bottom left), whereas the 1-fold coordinated Si atoms are found close to the interface (histogram on the bottom right). Figure adaped from [16].

Fig. 9.9 presents the structural parameters of the ReaxFF oxidized surface for comparison. The typical Si–O bond length, as indicated by the top left histogram, is 1.55 Å on average, which is less than the 1.63 Å stated in the literature. The scatter of the average bond length is quite small and the densities that produce ReaxFF are roughly 10% higher than the experimental values [119], indicating that the bond lengths in the interfacial and oxide areas of the interfacial structure are shortened. The histogram at the upper right shows, that the average O–Si–O bond angle closely matches the 109.47° for the ideal SiO_4 tetrahedron. As can be recognized in the histogram on the bottom left, only very few Si atoms coordinate 4-fold due to the high diffusion of oxygen into the Si crystal. The 4-fold coordinated Si atoms are only found in the top third of the oxide, whereas the 1-fold coordinated Si atoms (CN=1) are mainly found in the deeper part of the oxide (histogram on the bottom right). This indicates, that ReaxFF overestimates the diffusion of oxygen and as a result, the distribution of O atoms among the Si atoms in the crystal has a low density. The outcome is a very large interface that strongly differs from the experimental results [119, 122–125] in that the Si–O coordination is lower than expected.

In summary, even though the MD simulations with GAP run around 25 times slower than MD simulations using ReaxFF, the resultant oxide layer formed by our GAP produces much more realistic interface structures. Because of the overestimated oxygen diffusion, ReaxFF produces oxides also with a much higher defect density.

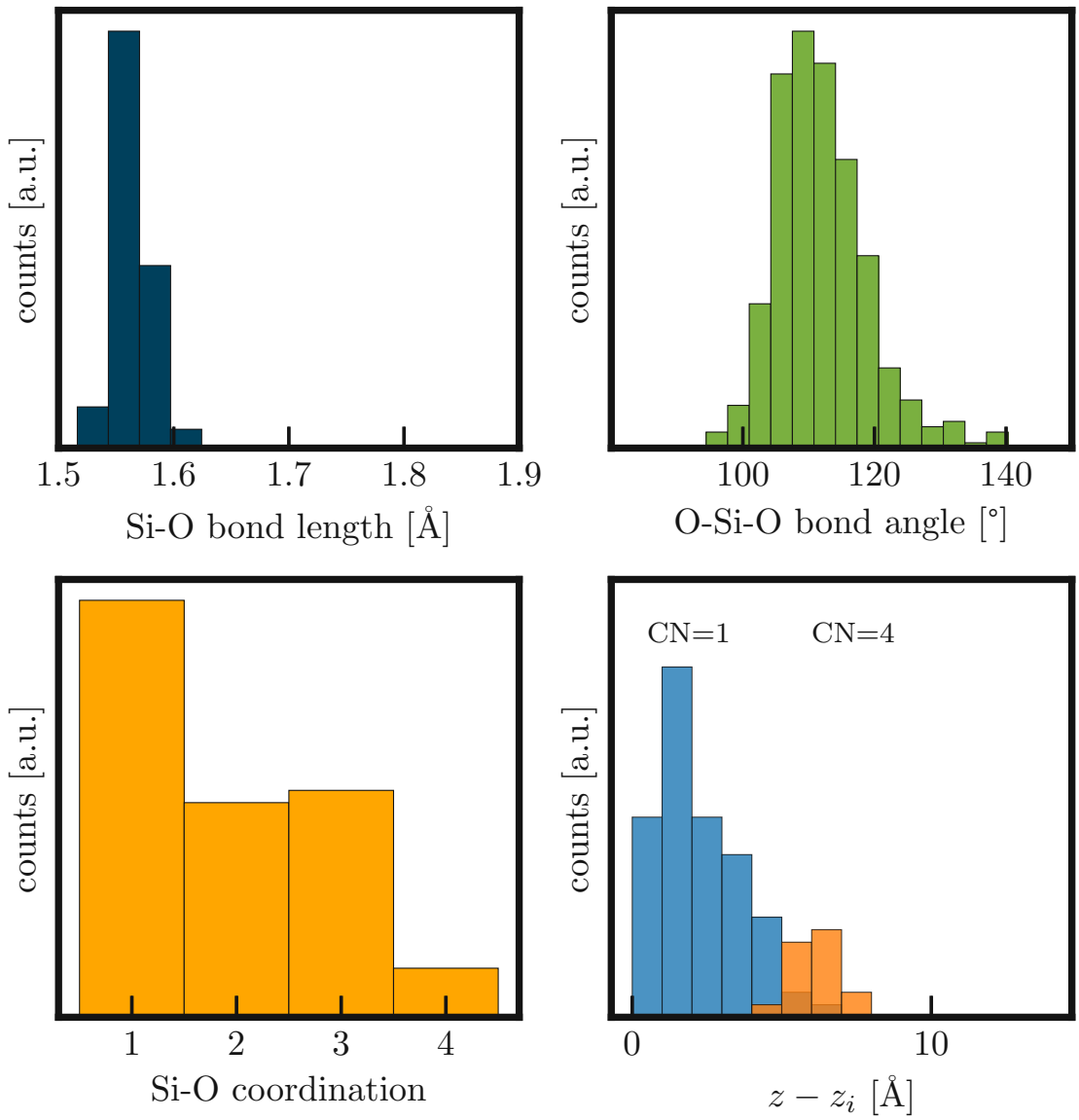


Figure 9.9: Structural properties of an oxidized surface with a cell size of 31.046×31.046×60.00 Å, which was oxidized using ReaxFF as a force field in MD simulations. The left top histogram indicates that the mean Si–O bond is 1.55 Å, which is significantly lower than the 1.63 Å from the literature. The mean O–Si–O bond angles (histogram on the top right) match the 109.47° from an ideal tetrahedron. The histogram at the bottom left shows that most of the Si atoms in the oxide are 1-fold coordinated (CN=1) and the fewest are 4-fold coordinated (CN=4). The 4-fold coordinated Si atoms are located exclusively in the top third of the oxide, which is illustrated in the histogram at the bottom right. Figure adaped from [16].

10 Summary and Outlook

The present master thesis focusses on training a interatomic ML potential based on GAP for the oxidation of silicon and investigates its applicability for MD simulations with the purpose of modeling the thermal oxidation of Si. This process was required by carefully creating and selecting training structures, adjusting the training parameters, and validating the trained potentials using selected test structures and MD simulations. The main results of this work show that the trained GAP potential can describe the atomistic structure, dynamics, and energetics of Si/SiO_2 systems with high accuracy and efficiency. By comparison with DFT data and experimental observations, it was shown that the GAP potential provides reproducible results that agree well with the experimental data while significantly reducing the computational effort.

Due to the extremely complex nature of the PES, which is supposed to reflect the chemical and physical behavior of silicon, oxygen, and hydrogen and their interactions, the training of such a GAP poses various challenges. The most challenging task was the prediction of the influence of how additional training structures will affect the accuracy of the current GAP. It was often found that adding additional training structures improved the MAE values for certain test structures, but simultaneously worsened the MAE values of the other test structures. Finding a GAP that had acceptable MAE values on all test structures proved to be a major challenge, resulting in a large number of trained GAPs. Also, an extensive training dataset will increase the GAP training time. For example, the training time for the GAP *A11a*, which utilized a training dataset of 1582 structures and virials, was 17 hours, compared to around 3 hours for the GAP *B1*, which employed a total of 626 training structures.

The ultimate goal of any ML potential is to predict the interactions of the atoms as accurately as possible, regardless of the size and shape of a given structure. However, this goal is extremely ambitious and hard to achieve. The ML potential presented in this thesis achieves excellent results in the oxidation of silicon, especially for surfaces. Compared to the very successful reactive force field (ReaxFF), the trained GAP produces much more realistic interface structures. It allows to run MD simulation with DFT accuracy for modeling interfaces and nanostructures comprised of the technologically highly relevant material system Si/SiO_2. Compared to AIMD calculations, MD simulations using GAP as a force field run substantially quicker by several orders of magnitude with significantly reduced computational costs.

Bibliography

[1] Kelin J. Kuhn. Considerations for ultimate cmos scaling. *IEEE Transactions on Electron Devices*, 59(7):1813–1828, 2012. doi: 10.1109/TED.2012.2193129.

[2] Floris A. Zwanenburg, Andrew S. Dzurak, Andrea Morello, Michelle Y. Simmons, Lloyd C. L. Hollenberg, Gerhard Klimeck, Sven Rogge, Susan N. Coppersmith, and Mark A. Eriksson. Silicon quantum electronics. *Rev. Mod. Phys.*, 85:961–1019, 2013. doi: 10.1103/RevModPhys.85.961.

[3] Ron Jansen. Silicon spintronics. *Nature Materials*, 11(5):400–408, May 2012. ISSN 1476-4660. doi: 10.1038/nmat3293.

[4] Igor Žutić, Jaroslav Fabian, and S. Das Sarma. Spintronics: Fundamentals and applications. *Rev. Mod. Phys.*, 76:323–410, Apr 2004. doi: 10.1103/RevModPhys.76.323.

[5] Lingjie Guo, Effendi Leobandung, and Stephen Y. Chou. A silicon single-electron transistor memory operating at room temperature. *Science*, 275(5300): 649–651, Jan 1997. doi: 10.1126/science.275.5300.649.

[6] M. Razeghi. chapter 2, pages 41–82. Springer US, 1 edition, 2010.

[7] Sokrates T. Pantelides, Sanwu Wang, A. Franceschetti, Ryszard Buczko, M. Di Ventra, Sergey N. Rashkeev, L. Tsetseris, M.H. Evans, I.G. Batyrev, Leonard C. Feldman, S. Dhar, K. McDonald, Robert A. Weller, R.D. Schrimpf, D.M. Fleetwood, X.J. Zhou, John R. Williams, Chin Che Tin, G.Y. Chung, Tamara Isaacs-Smith, S.R. Wang, S.J. Pennycook, G. Duscher, K. Van Benthem, and L.M. Porter. Si/SiO_2 and SiC/SiO_2 interfaces for mosfets – challenges and advances. In *Silicon Carbide and Related Materials 2005*, volume 527 of *Materials Science Forum*, pages 935–948. Trans Tech Publications Ltd, 10 2006. doi: 10.4028/www.scientific.net/MSF.527-529.935.

[8] Adri C. T. van Duin, Alejandro Strachan, Shannon Stewman, Qingsong Zhang, Xin Xu, and William A. Goddard. Reaxff SiO reactive force field for silicon and silicon oxide systems. *The Journal of Physical Chemistry A*, 107(19):3803–3811, 2003. doi: 10.1021/jp0276303.

[9] Lukas Cvitkovich, Dominic Waldhör, Al-Moatassem El-Sayed, Markus Jech, Christoph Wilhelmer, and Tibor Grasser. Dynamic modeling of Si(100) thermal oxidation: Oxidation mechanisms and realistic amorphous interface generation. *Applied Surface Science*, 610:155378, 2023. ISSN 0169-4332. doi: https://doi.org/10.1016/j.apsusc.2022.155378.

[10] Diego Milardovich, Christoph Wilhelmer, Dominic Waldhoer, Lukas Cvitkovich, Ganesh Sivaraman, and Tibor Grasser. Machine learning interatomic potential for silicon-nitride (Si_3N_4) by active learning. *The Journal of Chemical Physics*, 158(19):194802, 05 2023. ISSN 0021-9606. doi: 10.1063/5.0146753.

[11] D.S. Sholl and J.A. Steckel. *Density Functional Theory: A Practical Introduction*. Wiley, 2011. ISBN 9781118211045.

[12] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133–A1138, Nov 1965. doi: 10.1103/PhysRev.140.A1133.

[13] Albert P Bartók, Mike C Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.*, 104(13):136403, Apr 2010. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.104.136403.

[14] González, M.A. Force fields and molecular dynamics simulations. *JDN*, 12: 169–200, 2011. doi: 10.1051/sfn/201112009.

[15] Diego Milardovich, Dominic Waldhoer, Markus Jech, Al-Moatasem Bellah El-Sayed, and Tibor Grasser. Building robust machine learning force fields by composite gaussian approximation potentials. *Solid-State Electronics*, 200:108529, 2023. ISSN 0038-1101. doi: https://doi.org/10.1016/j.sse.2022.108529.

[16] Lukas Cvitkovich, Franz Fehringer, Christoph Wilhelmer, Diego Milardovich, Dominic Waldhör, and Tibor Grasser. Machine learning force field for thermal oxidation of silicon, 2024. URL https://arxiv.org/abs/2405.13635.

[17] Lukas Cvitkovich, Dominic Waldhör, Al-Moatasem El-Sayed, Markus Jech, Christoph Wilhelmer, and Tibor Grasser. Ab-initio modeling of the initial stages of si (100) thermal oxidation. In *PSI-K 2022: abstracts book*, page 209, 2022.

[18] B. E. Deal and A. S. Grove. General relationship for the thermal oxidation of silicon. *J. Appl. Phys.*, 36(12):3770–3778, 1965. doi: 10.1063/1.1713945.

[19] Angelo Bongiorno and Alfredo Pasquarello. Reaction of the oxygen molecule at the Si(100)$-SiO_2$ interface during silicon oxidation. *Phys. Rev. Lett.*, 93: 086102, Aug 2004. doi: 10.1103/PhysRevLett.93.086102.

[20] A. Bongiorno and A. Pasquarello. O_2 oxidation reaction at the Si(100)-SiO_2 interface: A first-principles investigation. *J. Mater. Sci.*, 40:3047–3050, 06 2005. doi: 10.1007/s10853-005-2663-7.

[21] Angelo Bongiorno and Alfredo Pasquarello. Multiscale modeling of oxygen diffusion through the oxide during silicon oxidation. *Phys. Rev. B*, 70:195312, Nov 2004. doi: 10.1103/PhysRevB.70.195312.

[22] A. Pasquarello, M. S. Hybertsen, and R. Car. Interface structure between silicon and its oxide by first-principles molecular dynamics. *Nature*, 396(6706): 58–60, 1998. doi: 10.1038/23908.

[23] F. J. Himpsel, F. R. McFeely, A. Taleb-Ibrahimi, J. A. Yarmoff, and G. Hollinger. Microscopic structure of the SiO_2/Si interface. *Phys. Rev. B*, 38:6084–6096, Sep 1988. doi: 10.1103/PhysRevB.38.6084.

[24] Toru Akiyama and Hiroyuki Kageshima. Reaction mechanisms of oxygen at SiO_2/Si(100) interface. *Surf. Sci.*, 576(1):L65–L70, 2005. ISSN 0039-6028. doi: https://doi.org/10.1016/j.susc.2005.01.001.

[25] E. P. Gusev, H. C. Lu, T. Gustafsson, and E. Garfunkel. Growth mechanism of thin silicon oxide films on Si(100) studied by medium-energy ion scattering. *Phys. Rev. B*, 52:1759–1775, Jul 1995. doi: 10.1103/PhysRevB.52.1759.

[26] E. Rosencher, A. Straboni, S. Rigo, and G. Amsel. An 18o study of the thermal oxidation of silicon in oxygen. *Appl. Phys. Lett.*, 34(4):254–256, 1979. doi: 10.1063/1.90771.

[27] M. A. Hopper, R. A. Clarke, and L. Young. Thermal oxidation of silicon: In situ measurement of the growth rate using ellipsometry. *Journal of The Electrochemical Society*, 122(9):1216, Sep 1975. doi: 10.1149/1.2134428.

[28] Keichiro Ohsawa, Yusuke Hayashi, Ryu Hasunuma, and Kikuo Yamabe. Roughness increase on surface and interface of SiO_2 grown on atomically flat Si(111) terrace. *Japanese Journal of Applied Physics*, 48(5S1):05DB02, May 2009. doi: 10.1143/JJAP.48.05DB02.

[29] W. Heywang and K. H. Zaininger. *Silicon: the Semiconductor Material*, pages 25–42. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-662-09897-4. doi: 10.1007/978-3-662-09897-4_2.

[30] M. Waldrop. The chips are down for moore's law. *Nature News*, 530:144, Feb 2016. doi: 10.1038/530144a.

[31] F. Shimura. *Semiconductor Silicon Crystal Technology*. Academic Press, 1989. ISBN 9780126400458.

[32] T.L. Einstein. 5 - equilibrium shape of crystals. In Tatau Nishinaga, editor, *Handbook of Crystal Growth (Second Edition)*, pages 215–264. Elsevier, Boston, second edition edition, 2015. ISBN 978-0-444-56369-9. doi: https://doi.org/10.1016/B978-0-444-56369-9.00005-8.

[33] Simon Min Sze. *Semiconductor devices: physics and technology*. John wiley & sons, 2008.

[34] Sergey Nekrashevich and V.A. Gritsenko. Electronic structure of silicon dioxide (a review). *Physics of the Solid State*, 56, Feb 2014. doi: 10.1134/S106378341402022X.

[35] C R Helms and E H Poindexter. The silicon-silicon dioxide system: Its microstructure and imperfections. *Reports on Progress in Physics*, 57(8):791, Aug 1994. doi: 10.1088/0034-4885/57/8/002.

[36] Christoph Wilhelmer, Dominic Waldhoer, Lukas Cvitkovich, Diego Milardovich, Michael Waltl, and Tibor Grasser. Over-and undercoordinated atoms as a source of electron and hole traps in amorphous silicon nitride (a-Si_3N_4). *Nanomaterials*, 13(16):2286, 2023.

[37] Yue-Yang Liu, Fan Zheng, Xiangwei Jiang, Jun-Wei Luo, Shu-Shen Li, and Lin-Wang Wang. Ab initio investigation of charge trapping across the crystalline-Si–amorphous-SiO_2 interface. *Physical Review Applied*, 11(4):044058, 2019.

[38] Markus Jech, Al-Moatasem El-Sayed, Stanislav Tyaginov, Alexander L. Shluger, and Tibor Grasser. Ab-initio treatment of silicon-hydrogen bond rupture at Si/SiO_2 interfaces. *Phys. Rev. B*, 100:195302, Nov 2019. doi: 10.1103/PhysRevB.100.195302.

[39] Christoph Wilhelmer, Dominic Waldhoer, Markus Jech, Al-Moatasem Bellah El-Sayed, Lukas Cvitkovich, Michael Waltl, and Tibor Grasser. Ab initio investigations in amorphous silicon dioxide: Proposing a multi-state defect model for electron and hole capture. *Microelectronics Reliability*, 139:114801, 2022. ISSN 0026-2714. doi: https://doi.org/10.1016/j.microrel.2022.114801.

[40] NV Rumak, VV Khatko, and VN Plotnikov. Structure and properties of silicon dioxide thermal films i. SiO_2 films of up to 50 nm thickness. *physica status solidi (a)*, 86(1):93–100, 1984.

[41] Eugene A Irene. Models for the oxidation of silicon. *Critical Reviews in Solid State and Material Sciences*, 14(2):175–223, 1988.

[42] Ulrich Hilleringmann. *Silicon Semiconductor Technology: Processing and Integration of Microelectronic Devices*. Springer Fachmedien Wiesbaden GmbH, Wiesbaden, 1 edition, 2023. ISBN 365841040X.

[43] TH Yeh. Thermal oxidation of silicon. *Journal of Applied Physics*, 33(9): 2849–2850, 1962.

[44] Hisham Z. Massoud, James D. Plummer, and Eugene A. Irene. Thermal oxidation of silicon in dry oxygen growth-rate enhancement in the thin regime: I . experimental results. *Journal of The Electrochemical Society*, 132(11):2685, Nov 1985. doi: 10.1149/1.2113648.

[45] Hiroyuki Kageshima and Kenji Shiraishi. First-principles study of oxide growth on Si(100) surfaces and at SiO_2/Si(100) interfaces. *Phys. Rev. Lett.*, 81:5936–5939, Dec 1998. doi: 10.1103/PhysRevLett.81.5936.

[46] EA Haupfear, EC Olson, and LD Schmidt. Kinetics of SiO_2 deposition from tetraethylorthosilicate. *Journal of the Electrochemical Society*, 141(7):1943, 1994.

[47] Frank Allen Shemansky Jr. *Low-pressure chemical vapor deposition of silicon dioxide from tetraethylorthosilicate*. Arizona State University, 1991.

[48] M. K. Bruska and J. Piechota. Density functional study of sulphur hexafluoride (SF_6) and its hydrogen derivatives. *Molecular Simulation*, 34:1041–1050, 2008. doi: 10.1080/08927020802258708.

[49] Thomas D. Kühne, Marcella Iannuzzi, Mauro Del Ben, Vladimir V. Rybkin, Patrick Seewald, Frederick Stein, Teodoro Laino, Rustam Z. Khaliullin, Ole Schütt, Florian Schiffmann, Dorothea Golze, Jan Wilhelm, Sergey Chulkov, Mohammad Hossein Bani-Hashemian, Valéry Weber, Urban Borštnik, Mathieu Taillefumier, Alice Shoshana Jakobovits, Alfio Lazzaro, Hans Pabst, Tiziano Müller, Robert Schade, Manuel Guidon, Samuel Andermatt, Nico Holmberg, Gregory K. Schenter, Anna Hehn, Augustin Bussy, Fabian Belleflamme, Gloria Tabacchi, Andreas Glöß, Michael Lass, Iain Bethune, Christopher J. Mundy, Christian Plessl, Matt Watkins, Joost VandeVondele, Matthias Krack, and Jürg Hutter. CP2K: An electronic structure and molecular dynamics software package - Quickstep: Efficient and accurate electronic structure calculations. *The Journal of Chemical Physics*, 152(19):194103, May 2020. ISSN 0021-9606. doi: 10.1063/5.0007045.

[50] E. Schrödinger. Quantisierung als Eigenwertproblem. *Annalen Phys.*, 386(18): 109–139, 1926. doi: 10.1002/andp.19263861802.

[51] E. Merzbacher. *Quantum Mechanics*. Wiley, 1998. ISBN 9780471887027.

[52] F. Giustino. *Materials Modelling Using Density Functional Theory: Properties and Predictions*. Oxford University Press, 2014. ISBN 9780199662449.

[53] John David Jackson. *Classical electrodynamics*. John Wiley & Sons, 2012.

[54] Peter Fulde and Hermann Stoll. Dealing with the exponential wall in electronic structure calculations. *The Journal of Chemical Physics*, 146(19):194107, May 2017. ISSN 0021-9606. doi: 10.1063/1.4983207.

[55] M. Born and R. Oppenheimer. Zur quantentheorie der molekeln. *Annalen der Physik*, 389(20):457–484, 1927. doi: https://doi.org/10.1002/andp.19273892002.

[56] C. Zhu, a. A. W. Jasper, and D. G. Truhlar. Non-born-oppenheimer liouville-von neumann dynamics. evolution of a subsystem controlled by linear and population-driven decay of mixing with decoherent and coherent switching. *Journal of Chemical Theory and Computation*, 1:527–540, 2005. doi: 10.1021/ct050021p.

[57] Chris Lorenz and Nikos L. Doltsinis. *Molecular Dynamics Simulation: From "Ab Initio" to "Coarse Grained"*, pages 195–238. Springer Netherlands, Dordrecht, 2012. ISBN 978-94-007-0711-5. doi: 10.1007/978-94-007-0711-5_7.

[58] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136: B864–B871, Nov 1964. doi: 10.1103/PhysRev.136.B864.

[59] 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, Jul 2001. ISSN 0094-243X. doi: 10.1063/1.1390175.

[60] M. Ropo, K. Kokko, and L. Vitos. Assessing the perdew-burke-ernzerhof exchange-correlation density functional revised for metallic bulk and surface systems. *Phys. Rev. B*, 77:195445, May 2008. doi: 10.1103/PhysRevB.77.195445.

[61] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77:3865–3868, Oct 1996. doi: 10.1103/PhysRevLett.77.3865.

[62] John P. Perdew and Mel Levy. Physical content of the exact kohn-sham orbital energies: Band gaps and derivative discontinuities. *Phys. Rev. Lett.*, 51:1884–1887, Nov 1983. doi: 10.1103/PhysRevLett.51.1884.

[63] Axel D. Becke. A new mixing of Hartree-Fock and local density-functional theories. , 98(2):1372–1377, Jan 1993. doi: 10.1063/1.464304.

[64] P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch. Ab initio calculation of vibrational absorption and circular dichroism spectra using density functional force fields. *The Journal of Physical Chemistry*, 98(45):11623–11627, 1994. doi: 10.1021/j100096a001.

[65] John P. Perdew, Matthias Ernzerhof, and Kieron Burke. Rationale for mixing exact exchange with density functional approximations. *The Journal of Chemical Physics*, 105(22):9982–9985, Dec 1996. ISSN 0021-9606. doi: 10.1063/1.472933.

[66] Alejandro Garza and Gustavo Scuseria. Predicting band gaps with hybrid density functionals. *The Journal of Physical Chemistry Letters*, 7, Aug 2016. doi: 10.1021/acs.jpclett.6b01807.

[67] David Bohm and David Pines. A collective description of electron interactions. i. magnetic interactions. *Phys. Rev.*, 82:625–634, Jun 1951. doi: 10.1103/PhysRev.82.625.

[68] Vladimir I Anisimov, F Aryasetiawan, and A I Lichtenstein. First-principles calculations of the electronic structure and spectra of strongly correlated systems: the lda+ u method. *Journal of Physics: Condensed Matter*, 9(4):767, Jan 1997. doi: 10.1088/0953-8984/9/4/002.

[69] J. C. Slater. Atomic shielding constants. *Phys. Rev.*, 36:57–64, Jul 1930. doi: 10.1103/PhysRev.36.57.

[70] Samuel Francis Boys. Electronic wave functions - i. a general method of calculation for the stationary states of any molecular system. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 200: 542 – 554, 1950.

[71] Richard C. Raffenetti. General contraction of Gaussian atomic orbitals: Core, valence, polarization, and diffuse basis sets; Molecular integral evaluation. *The Journal of Chemical Physics*, 58(10):4452–4458, Aug 2003. ISSN 0021-9606. doi: 10.1063/1.1679007.

[72] Joost VandeVondele, Matthias Krack, Fawzi Mohamed, Michele Parrinello, Thomas Chassaing, and Jürg Hutter. Quickstep: Fast and accurate density

functional calculations using a mixed gaussian and plane waves approach. *Computer Physics Communications*, 167(2):103–128, 2005. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2004.12.014.

[73] Gerald Lippert, Jurg Hutter, and Michele Parrinello. A hybrid Gaussian and plane wave density functional scheme. *Molecular Physics*, 92(3):477–488, Oct 1997. doi: 10.1080/002689797170220.

[74] S. Goedecker, M. Teter, and J. Hutter. Separable dual-space gaussian pseudopotentials. *Phys. Rev. B*, 54:1703–1710, Jul 1996. doi: 10.1103/PhysRevB.54.1703.

[75] C. Hartwigsen, S. Goedecker, and J. Hutter. Relativistic separable dual-space gaussian pseudopotentials from h to rn. *Phys. Rev. B*, 58:3641–3662, Aug 1998. doi: 10.1103/PhysRevB.58.3641.

[76] S. Goedecker, M. Teter, and J. Hutter. Separable dual-space gaussian pseudopotentials. *Phys. Rev. B*, 54:1703–1710, Jul 1996. doi: 10.1103/PhysRevB.54.1703.

[77] A. D. Becke. Density-functional exchange-energy approximation with correct asymptotic behavior. *Phys. Rev. A*, 38:3098–3100, Sep 1988. doi: 10.1103/PhysRevA.38.3098.

[78] John P. Perdew. Density-functional approximation for the correlation energy of the inhomogeneous electron gas. *Phys. Rev. B*, 33:8822–8824, Jun 1986. doi: 10.1103/PhysRevB.33.8822.

[79] Chengteh Lee, Weitao Yang, and Robert G. Parr. Development of the colle-salvetti correlation-energy formula into a functional of the electron density. *Phys. Rev. B*, 37:785–789, Jan 1988. doi: 10.1103/PhysRevB.37.785.

[80] Burkhard Miehlich, Andreas Savin, Hermann Stoll, and Heinzwerner Preuss. Results obtained with the correlation energy density functionals of becke and lee, yang and parr. *Chemical Physics Letters*, 157(3):200–206, 1989. ISSN 0009-2614. doi: https://doi.org/10.1016/0009-2614(89)87234-3.

[81] Fred A. Hamprecht, Aron J. Cohen, David J. Tozer, and Nicholas C. Handy. Development and assessment of new exchange-correlation functionals. *The Journal of Chemical Physics*, 109(15):6264–6271, Oct 1998. ISSN 0021-9606. doi: 10.1063/1.477267.

[82] Victor Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, Ekin Cubuk, Samuel Schoenholz, A. Obika, A. Nelson, T. Back, D. Hassabis, and P. Kohli. Unveiling the predictive power of static structure in glassy systems. *Nature Physics*, 16: 448–454, Apr 2020. doi: 10.1038/s41567-020-0842-8.

[83] Yu Wang, Xinxing Peng, Alex Abelson, Penghao Xiao, Caroline Qian, Lei Yu, Colin Ophus, Peter Ercius, Lin-Wang Wang, Matt Law, and Haimei Zheng. Dynamic deformability of individual pbse nanocrystals during superlattice phase transitions. *Science Advances*, 5(6):eaaw5623, 2019. doi: 10.1126/sciadv.aaw5623.

[84] Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications*, volume 1 of *Computational Science Series*. Academic Press, San Diego, second edition, 2002.

[85] Efrem Braun, Justin Gilmer, Heather B. Mayes, David L. Mobley, Jacob I. Monroe, Samarjeet Prasad, and Daniel M. Zuckerman. Best practices for foundations in molecular simulations [article v1.0]. *Living Journal of Computational Molecular Science*, 1(1):5957, Nov 2018. doi: 10.33011/livecoms.1.1.5957.

[86] Ralf Schneider, Amit Raj Sharma, and Abha Rai. *Introduction to Molecular Dynamics*, pages 3–40. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-74686-7. doi: 10.1007/978-3-540-74686-7_1.

[87] J. E. Jones. On the determination of molecular fields. ii. from the equation of state of a gas. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 106(738):pp. 463–477, 1924. ISSN 09501207.

[88] Adri C. T. van Duin, Siddharth Dasgupta, Francois Lorant, and William A. Goddard. Reaxff: a reactive force field for hydrocarbons. *The Journal of Physical Chemistry A*, 105(41):9396–9409, 2001. doi: 10.1021/jp004368u.

[89] Michael F. Russo and Adri C.T. van Duin. Atomistic-scale simulations of chemical reactions: Bridging from quantum chemistry to engineering. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 269(14):1549–1554, 2011. ISSN 0168-583X. doi: https://doi.org/10.1016/j.nimb.2010.12.053. Computer Simulations of Radiation Effects in Solids.

[90] A W Lees and S F Edwards. The computer study of transport processes under extreme conditions. *Journal of Physics C: Solid State Physics*, 5(15):1921, Aug 1972. doi: 10.1088/0022-3719/5/15/006.

[91] D.C. Rapaport. *The Art of Molecular Dynamics Simulation*. Cambridge University Press, 2004. ISBN 9780521825689.

[92] Richard Rennie and Jonathan Law. *A Dictionary of Physics*. Oxford University Press, 2019. ISBN 9780191860805. doi: 10.1093/acref/9780198821472.001.0001.

[93] Josiah Willard Gibbs. *Elementary Principles in Statistical Mechanics: Developed with Especial Reference to the Rational Foundation of Thermodynamics*. Cambridge Library Collection - Mathematics. Cambridge University Press, 2010.

[94] Herman JC Berendsen, JPM van Postma, Wilfred F Van Gunsteren, ARHJ DiNola, and Jan R Haak. Molecular dynamics with coupling to an external bath. *The Journal of chemical physics*, 81(8):3684–3690, 1984.

[95] Tetsuya Morishita. Fluctuation formulas in molecular-dynamics simulations with the weak coupling heat bath. , 113(8):2976–2982, Aug 2000. doi: 10.1063/1.1287333.

[96] Oded Farago. Langevin thermostat for robust configurational and kinetic sampling. *Physica A: Statistical Mechanics and its Applications*, 534:122210, 2019. ISSN 0378-4371. doi: https://doi.org/10.1016/j.physa.2019.122210.

[97] Paul Langevin. On the theory of brownian motion. *CR Acad Sci (Paris)*, 146: 530, 1908.

[98] Giorgio Parisi and Ramamurti Shankar. Statistical Field Theory. *Physics Today*, 41(12):110–110, Dec 1988. ISSN 0031-9228. doi: 10.1063/1.2811677.

[99] James M Haile. *Molecular dynamics simulation: elementary methods*. John Wiley & Sons, Inc., 1992.

[100] Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications*. Academic Press, Inc., USA, 1st edition, 1996. ISBN 0122673700.

[101] M. García. The nosé-hoover thermostat in molecular dynamics for nuclear matter. *Journal of Mathematical Chemistry*, 40:63–69, Jan 2006. doi: 10.1007/s10910-006-9120-y.

[102] Puneet Kumar Patra and Baidurya Bhattacharya. Nonergodicity of the nose-hoover chain thermostat in computationally achievable time. *Phys. Rev. E*, 90: 043304, Oct 2014. doi: 10.1103/PhysRevE.90.043304.

[103] Loup Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Phys. Rev.*, 159:98–103, Jul 1967. doi: 10.1103/PhysRev.159.98.

[104] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton. LAMMPS

- a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comp. Phys. Comm.*, 271:108171, 2022. doi: 10.1016/j.cpc.2021.108171.

[105] A. Bartók-Pártay. *The Gaussian Approximation Potential: An Interatomic Potential Derived from First Principles Quantum Mechanics*. Springer Theses. Springer Berlin Heidelberg, 2010. ISBN 9783642140679.

[106] Christoph Wilhelmer, Dominic Waldhör, Lukas Cvitkovich, Diego Milardovich, Michael Waltl, and Tibor Grasser. Polaron formation in the hydrogenated amorphous silicon nitride Si_3N_4. *Phys. Rev. B*, 110:045201, Jul 2024. doi: 10.1103/PhysRevB.110.045201.

[107] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O Anatole Von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical review letters*, 108(5):058301, 2012.

[108] Albert P. Bartók, Mike C. Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.*, 104:136403, Apr 2010. doi: 10.1103/PhysRevLett.104.136403.

[109] Albert P. Bartók and Gábor Csányi. Gaussian approximation potentials: A brief tutorial introduction. *International Journal of Quantum Chemistry*, 115 (16):1051–1057, 2015. doi: https://doi.org/10.1002/qua.24927.

[110] Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Phys. Rev. B*, 87:184115, May 2013. doi: 10.1103/PhysRevB.87.184115.

[111] Volker L. Deringer, Albert P. Bartók, Noam Bernstein, David M. Wilkins, Michele Ceriotti, and Gábor Csányi. Gaussian process regression for materials and molecules. *Chemical Reviews*, 121(16):10073–10141, 2021. doi: 10.1021/acs.chemrev.1c00022. PMID: 34398616.

[112] Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(65):1939–1959, 2005.

[113] Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005.

[114] A. N. Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Soviet Math. Dokl.*, 4:1035–1038, 1963.

[115] Gábor Csányi, Steven Winfield, J R Kermode, A De Vita, Alessio Comisso, Noam Bernstein, and Michael C Payne. Expressive programming for computational physics in fortran 95+. *IoP Comput. Phys. Newsletter*, page Spring 2007, 2007.

[116] Gábor Csányi, Noam Bernstein, and James Richard Kermode. libAtoms/QUIP, Nov 2021.

[117] Pedro Domingos. A few useful things to know about machine learning. *Commun. ACM*, 55(10):78–87, oct 2012. ISSN 0001-0782. doi: 10.1145/2347736.2347755.

[118] Nadire Nayir, Adri C. T. van Duin, and Sakir Erkoc. Development of the reaxff reactive force field for inherent point defects in the Si/Silica system. *The Journal of Physical Chemistry A*, 123(19):4303–4313, May 2019. ISSN 1089-5639. doi: 10.1021/acs.jpca.9b01481.

[119] Alain C. Diebold, David Venables, Yves Chabal, David Muller, Marcus Weldon, and Eric Garfunkel. Characterization and production metrology of thin transistor gate oxide films. *Materials Science in Semiconductor Processing*, 2(2):103–147, 1999. ISSN 1369-8001. doi: https://doi.org/10.1016/S1369-8001(99)00009-8.

[120] R. L. Mozzi and B. E. Warren. The structure of vitreous silica. *Journal of Applied Crystallography*, 2(4):164–172, 1969. doi: https://doi.org/10.1107/S0021889869006868.

[121] K. Hirose, H. Nohira, T. Koike, K. Sakano, and T. Hattori. Structural transition layer at SiO_2/Si interfaces. *Phys. Rev. B*, 59:5617–5621, Feb 1999. doi: 10.1103/PhysRevB.59.5617.

[122] D. A. Muller, T. Sorsch, S. Moccio, F. H. Baumann, K. Evans-Lutterodt, and G. Timp. The electronic structure at the atomic scale of ultrathin gate oxides. *Nature*, 399(6738):758–761, Jun 1999. ISSN 1476-4687. doi: 10.1038/21602.

[123] David A. Muller and Glen D. Wilk. Atomic scale measurements of the interfacial electronic structure and chemistry of zirconium silicate gate dielectrics. *Applied Physics Letters*, 79(25):4195–4197, 12 2001. ISSN 0003-6951. doi: 10.1063/1.1426268.

[124] J. H. Oh, H. W. Yeom, Y. Hagimoto, K. Ono, M. Oshima, N. Hirashita, M. Nywa, A. Toriumi, and A. Kakizaki. Chemical structure of the ultrathin SiO_2/Si interface: An angle-resolved Si $2p$ photoemission study. *Phys. Rev. B*, 63:205310, Apr 2001. doi: 10.1103/PhysRevB.63.205310.

[125] Noriyuki Miyata, Heiji Watanabe, and Masakazu Ichikawa. Atomic-scale structure of SiO_2/Si interface formed by furnace oxidation. *Phys. Rev. B*, 58:13670–13676, Nov 1998. doi: 10.1103/PhysRevB.58.13670.

[126] A. H. Carim and R. Sinclair. The evolution of Si/SiO_2 interface roughness. *Journal of The Electrochemical Society*, 134(3):741, Mar 1987. doi: 10.1149/1.2100544.

List of Figures