Efficiently coupling QM and MD for the study of electrode-electrolyte interfaces

by

S.A.H. Hermans

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday June 23, 2022 at 10:30 AM.

Student number:	4582608
Project duration:	September, 2021 – June, 2022
Supervisor:	dr. Remco M. Hartkamp
Independent examiners:	prof. dr. Peter G. Steeneken
	dr. Timon Idema

TUDelft

Abstract

In this thesis, a proof of concept was established for the use of a novel coupled QM-MD approach to modelling metallic (copper) electrode-electrolyte interfaces. SCC-DFTB calculations of the instantaneous electronic structure of a copper electrode were coupled to a classical MD simulation of an electrode-electrolyte interface. The applied QM-MD method was described rigorously, and used to investigate the compound distribution and dynamics at the interface, relative to a fully classical MD simulation. Polarisation effects were observed to bring about a significant increase in the attraction between cations and the cathode. Moreover, local polarisation of the cathode was found to immobilise adsorbed cations, and induce an increased orientational preference of the nearby water dipoles. The secondary goal of this thesis was to explore to what extent neural networks are able to replicate SCC-DFTB calculations of the electronic charge density on a metallic electrode. Using a computer vision approach, qualitative evidence was obtained indicating that neural networks can be used to replicate SCC-DFTB predictions on periodic metallic surfaces.

List of abbreviations

ADF	-	Amsterdam density functional
AMS	-	Amsterdam modeling suite
BO	-	Born-Oppenheimer
CNN	-	Convolutional neural network
DFT	-	Density functional theory
DFTB	-	Density functional tight-binding
ECC	-	Electronic charge/continuum correction
EDL	-	Electric double layer
EWMA	-	Exponentially weighted moving average
FCC	-	Face-centered cubic
FCN	-	Fully connected network
GGA	-	Generalized gradient approximation
HK	-	Hohenberg-Kohn
KS	-	Kohn-Sham
LAMMPS	-	Large-scale atomic/molecular massively parallel simulator
LCAO	-	Linear combination of atomic orbitals
LDA	-	Local-density approximation
LJ	-	Lennard-Jones
MAE	-	Mean absolute error
MD	-	Molecular dynamics
MSD	-	Mean squared displacement
NN	-	Neural network
QM	-	Quantum mechanics
RMSE	-	Root-mean-square error
SCC	-	Self-consistent charge
SCF	-	Self-consistent field
SGD	-	Stochastic gradient descent
SK	-	Slater-Koster
SPC/E	-	Extended simple point charge model, force field[1]
VdW	-	Van der Waals
VMD	-	Visual molecular dynamics[2]
XC	-	Exchange-correlation

Contents

1

Introduction

1.1. Motivation - Carbon dioxide reduction

As a result of the increasing global population and improving living standards, the demand for energy is expected to continue to grow in the near future. Simultaneously, evidence shows that carbon dioxide levels in the atmosphere are rising, primarily because of the use of fossil fuels for energy generation.[3] This rise has numerous detrimental effects on the natural world, and with it, on life of many forms. Solutions for curbing the effects of this development, are urgently needed. Electrochemical carbon dioxide reduction is a process in which carbon dioxide is converted to more reduced chemical species using electrical energy.[4] This way, useful and valuable products like ethanol could be synthesized, that can help meet the need for long-term energy carriers.[5] Additionally, by converting carbon dioxide, this process could play a role in controlling the concentration of the greenhouse gas in the atmosphere.

Currently, the applicability of the carbon dioxide reduction process is limited due to a lack of efficient electrocatalysts, which are required to bind and convert the carbon dioxide. Copper is considered the most suitable electrocatalyst material for the reduction of carbon dioxide to ethylene and ethanol.[6] This is because its binding energy for carbon monoxide, an intermediate product of carbon dioxide reduction, is between that of other metals.[7][8] Consequently, on copper, on the one hand, carbon monoxide binds long enough for it to be converted into more useful products, while on the other hand, it releases quickly enough for new carbon monoxide to be bound. To find and design suitable electrocatalyst setups, mechanistic understanding of the carbon dioxide reduction process on copper is required.

This understanding is currently lacking. For example, it is unclear what the (primary) mechanism is behind the observed dependence of the product selectivity of carbon dioxide reduction on the kind of cations present in the electrolyte.[9]. In this context, the description of the electrode-electrolyte interface, and more specifically that of the electrical double layer (EDL), schematically depicted in figure 1.1, is of great interest.[8][9][10] The EDL depends on many physical effects and characteristics of the system, like the ions present, the strength of the present electric field, and the local shape and material of the solid. Consequently, modelling the EDL accurately has proven to be difficult, and has been of considerable interest for many years.[11][12]

A physical effect that can significantly alter the interaction at the electrode-electrolyte interface, is electrostatic induction. By means of electrostatic induction, electric charges in an electrolyte solution, like those carried by ions in the EDL, can cause a redistribution of charges in the nearby solid electrode. This effect, commonly referred to as polarisation, is a function of the constantly evolving EDL configuration. In turn, the locally induced charges can significantly influence the interaction of the electrode with the electrolyte, and thus the EDL. In this thesis, the interest primarily goes out to efficiently modelling copper electrode-electrolyte interfaces. Specifically, the focus is on efficiently modelling, and attempting to improve the understanding of, the effects of electrode surface polarisation at that interface.
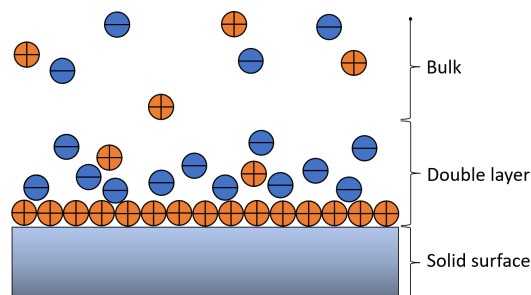
Figure 1.1: Schematic depiction of an electrical double layer (EDL), that represents the charge distribution in the vicinity of a surface exposed to a fluid. It is most basically envisioned as depicted here, as being composed of two parallel arrays of opposite charges. A negatively charged solid is shown, with an aqueous solution at the interface. Cations are coloured in red, anions in blue. To neutralise the negative charge of the solid, the first layer consists mainly of cations adsorbed on the surface. In turn, the cations at the surface attract anions from the bulk, forming the double layer. This description of the EDL is not necessarily physically accurate, as its structuring depends on many characteristics of the system.

1.2. Problem - Modelling polarisation effects

Metallic solids, like the copper electrode of interest, can only be described on a fundamental level using an *ab initio* approach, like density functional theory (DFT). In DFT, the electronic structure of many-electron systems is directly computed. However, DFT simulations are computationally costly, and are therefore usually limited to the simulation of only a few hundred atoms on relatively short time scales, i.e., on the order of picoseconds.

For many applications, classical molecular dynamics (C-MD) simulations can be very useful for describing and understanding the interactions between atoms and molecules. In C-MD, Newton's second law of motion is used to predict the physical trajectories of atoms, for which the interatomic forces are modelled using fitted interatomic potential functions. This method allows for simulating much larger systems on much longer timescales than *ab initio* approaches, generally being able to simulate systems of thousands of atoms on a timescale up to microseconds. When simulating electrode-electrolyte interfaces, however, polarisation effects are usually not rigorously accounted for, and surface polarisation effects are assumed to have a negligible effect on the electrolyte structuring at the interface.

It has been found that this assumption does not always hold in the case of metallic and semi-metallic surfaces. For example, DFT simulations of graphene sheets have shown a sizable redistribution of surface charge, induced by the proximity of ions in the nearby electrolyte.[13][14][15] Moreover, it has been demonstrated that not taking into account surface polarisation in the context of carbonaceous nanostructures, leads to incorrectly predicted adsorption tendencies of various cations (like Na^+ and K^+) at the surface of carbon nanotubes[15][16], and an overestimation of the Van der Waals interaction between liquid water and graphene, and thus an incorrectly predicted water contact angle.[17] The importance of surface polarisation has also been highlighted in several studies on the molecular charging mechanism of nanoporous electrodes.[18][19][20]. It is clear that polarisation effects can potentially be important for accurately describing electrode-electrolyte interfaces.

In an attempt to appropriately model the electrostatics, without having to continuously fall back on *ab initio* methods, various different modelling techniques have been developed, for both classical and coupled quantum and classical simulations. Unfortunately, the modulation of the Coulombic interaction due to the dynamic polarisation of the electrode is hard to predict, and all approaches, as described below, have shortcomings of some kind.

In most C-MD simulations, if any, only polarisability of the electrolyte is included. Usually, in that case, polarisation effects are accounted for in an averaged way through pre-set interaction potentials, with fixed charges assigned to the atoms. This way, the dynamical aspect of polarisation is not considered. Many polarisable force fields have been developed to explicitly model polarisability. In most standard polarisable force fields, using the so-called 'core-shell' approach, a positively charged core particle, and a negatively charged particle representing the electron cloud, are connected to each other by a spring-like harmonic oscillator potential.[21]

While these kind of models can work very well for describing, for example, certain electrolyte solutions, such atomic-centered approaches cannot account for charge transfer within a conductor, like the metallic copper electrode used in carbon dioxide reduction systems. Moreover, to provide an accurate description of polarisability, polarisable force fields, more than classical force fields, often need to be adapted to the specific context for which they are intended, limiting their applicability.

In recent decades, the importance of taking into account the polarisation of metal electrodes by ions in their immediate vicinity has been recognised, leading to a variety of approaches.[22][23][24][25][26] Most of these approaches are derived from the proposed model by Siepmann and Sprik[27], and/or its later adaptation by Reed et al.[28], in which electrode charges are allowed to fluctuate in order to maintain a constant electric potential throughout the electrode. This constrained value for the electric potential forms the main assumption of these methods, and is considered to hold for ideal conductors. Recently, in the work of Elliott et al.[14], it was shown that the ideal conductor assumption, at the basis of these constant potential methods, does not necessarily hold for conductors of any type, as it was seen it does not for the semi-metallic graphene.

1.3. Approach - Coupling of QM and MD

Alternatively, Elliott et al.[14] demonstrated a novel coupled quantum mechanical-classical molecular dynamics (QM-MD) approach to simulating electrode/electrolyte interfaces. In the demonstrated workflow, the electronic structure of the electrode was calculated using self-consistent charge density functional tight binding (SCC-DFTB), a semi-empirical approximation of DFT. The results of this calculation were subsequently communicated to a classical MD simulation, that computed the classical dynamics of the system. By iterating this scheme, the simulated system was allowed to evolve over time in response to the electronic polarisation of the electrode.

In this thesis, it is attempted to extend this approach to the study of a copper electrode-electrolyte interface. The main interest goes out studying the importance of taking into account polarisation effects in a copper electrode-electrolyte interface, as compared to the C-MD approach. Simultaneously, a proof of concept for using the described QM-MD approach will be established in the context of metallic electrode-electrolyte interfaces in general, to provide a different approach to the constant potential methods that, instead of relying on direct computations of the electronic structure, relies on the ideal conductor assumption.

1.4. The timescale issue

The timescale that is thought to be within reach using the described QM-MD loop, is on the order of a couple of hundreds of nanoseconds. Many problems where polarisation effects are important for an accurate description of the dynamics occur on much longer time scales. For example, atomistic protein folding, in which polarisation effects can play an essential role[29], is not unlikely to happen on a timescale of microseconds.[30] Moreover, studies on the development of electric double layers have indicated that timescales of microseconds to seconds are not exceptional.

For example, EDL formation on graphene FETs was demonstrated to occur on a timescale of microseconds to milliseconds[31], and EDL formation on gold electrodes was shown to be characterised by relaxation times spanning the range of less than microseconds to longer than seconds[32]. Similarly, in a study on EDL formation on charged silica, it was estimated that relaxation of the system could occur on a timescale on the order of seconds when involving lithium ions, and on the order of milliseconds when involving cesium.[33]

Studying these, or similar, problems using the described QM-MD is not feasible. Recently, Di Pasquale et al.[34] have shown that fully connected neural networks (NNs) are able to replace SCC-DFTB calculations to some extent in the context of a non-periodic graphene electrode. This is interesting because, once the network is trained, the time required to perform the relevant computations is negligible com-

pared to the SCC-DFTB computation time. Therefore, this thesis will explore the possibility of extending the previous work by investigating the ability of NNs to replace QM modelling in our system, which contains a periodic copper electrode.

1.5. Scope

The primary aim of this thesis is to establish a proof of concept for using the described QM-MD approach in the context of metallic electrode-electrolyte interfaces in general, and for copper specifically, and to improve the understanding of the local properties of the copper electrode-electrolyte interface, and specifically to obtain more insight in what role polarisability of the electrode plays. To do so, our approach will be rigorously described, and systematically tested. The main research question is as follows:

"To what extent are the compound distribution and dynamics at the copper electrode-electrolyte interface altered, when quantum mechanically modelled charge induction effects on the electrode surface are taken into account, as compared to a classical modelling approach?"

The secondary goal of this project is to integrate a trained NN into the MD simulation, as to replace the need for coupling the much more expensive QM models with the MD simulation. For this purpose, an additional (sub) research question is formulated:

"To what extent are neural networks able to replace periodic SCC-DFTB calculations on copper in a coupled SCC-DFTB/C-MD approach?"

1.6. Structure of the thesis

Multiple steps in the modelling of the copper electrode-electrolyte interface need to be taken. First, separately working SCC-DFTB and C-MD models need to be created. Subsequently, these need to be successfully coupled, and evaluated. Finally, appropriate neural networks that can be used to accelerate the coupling between the QM and MD models must be created, trained and evaluated..

In the following chapter, classical molecular dynamics, density functional theory, self-consistent charge density functional tight binding, and neural networks are described, that together constitute all methods used in this work. In chapter 3, the methods are described in detail. In chapter 4 all results are presented and discussed. Lastly, chapter 5 contains the conclusions.

2

Methodological background

In this chapter background information is provided on all simulation methods used in this thesis. Specifically, section 2.1 provides background on molecular dynamics, section 2.2 on density functional theory and section 2.3 on (self-consistent charge) density functional tight-binding. Finally, section 2.4 provides information on neural networks.

2.1. Molecular dynamics

Molecular dynamics (MD) is a computer simulation method for predicting the physical trajectories of atoms. To this end classical MD as applied in this thesis, solves Newton's second law of motion by integrating over discrete time steps. To predict the trajectory, the initial positions and velocities of the atoms in the system are required. Forces between atoms are modelled using interatomic potential functions for which suitable expressions need to be found. In the following subsections, both discrete time integration and interatomic potentials are shortly expanded upon. Lastly, an explanation is given of the periodic boundary conditions used in this work.

2.1.1. Interatomic potential

The interatomic force a particle experiences is generally given as the sum of three forces in the context of MD: the Coulombic interaction (charge attraction/repulsion), attractive Van der Waals (VdW) forces and repulsive electronic forces as a result of the Pauli exclusion principle for two identical fermionic particles. Generally, only pair-wise interaction is considered, because multi-body (more than two) potentials are too computationally expensive. A potential model provides a mathematical form for the interaction between neutral particles as a function of the distance between them. In the context of classical molecular dynamics, different kinds of functional forms have been used to approximate these forces. The electric potential from which the Coulombic attraction/repulsion arises is generally modelled using the standard Coulombic interaction potential

$$U_C\left(r\right) = \frac{1}{4\pi\epsilon_0}\frac{q_i q_j}{r}, \qquad (2.1)$$

given here for charged particles i and j. ϵ_0 represents the vacuum permittivity, q the charge of the respective particles and r the distance between them.

The Lennard-Jones (LJ) interaction potential

$$U_{LJ}\left(r\right) = 4\epsilon\left(\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6\right), \qquad (2.2)$$

is most commonly used to model the VdW and Pauli repulsion forces. The first term represents the exchange interaction between two particles, which yields a repulsive force as a result of the Pauli exclusion principle. The second term describes the attractive force as a result of VdW forces. The value

of ϵ corresponds to the minimum value of the potential well, whereas the value of σ corresponds to the distance at which the potential is equal to zero.

ϵ and σ are generally referred to as pair coefficients (pairwise force field coefficients). These are simulation parameters that have to be fitted and set for each pair of different atom types. Often, pair coefficients are only provided for interaction with itself (σ_{ii} & ϵ_{ii}). In that case, combining rules are used to determine the exchange interaction parameters between different types of non-bonded atoms. For the LJ-potential, the most commonly used combining rules are the Lorentz-Berthelot rules[35]

$$\epsilon_{ij} = \sqrt{\epsilon_{ii}\epsilon_{jj}}, \quad \sigma_{ij} = \frac{\sigma_{ii} + \sigma_{jj}}{2}, \tag{2.3}$$

and the geometric mixing rules

$$\epsilon_{ij} = \sqrt{\epsilon_{ii}\epsilon_{jj}}, \quad \sigma_{ij} = \sqrt{\sigma_{ii}\sigma_{jj}}. \tag{2.4}$$

Once suitable functional forms and fitting parameters for the interaction potential have been found, the respective forces on each atom can be calculated by taking the derivative of this potential, taking into account the direction of the force. This way, the force on a particle with label i can be written as

$$\mathbf{F}(\mathbf{r}) = -\frac{\mathbf{r}}{|r|}\frac{\partial U}{\partial r}. \tag{2.5}$$

Subsequently, by integration of the coupled equations of motion, the movement of the atoms in time can be predicted.

Intermolecular interactions between atoms use the same potential functions as previously described. For intramolecular interaction, however, the previously mentioned potential functions are generally set to zero for atoms at most two bonds away, and either scaled down or also set to zero for atoms more than two bonds away. Atoms that form a molecule (e.g. oxygen and hydrogen in water molecules), are generally held together using a different set of potential functions that prevent atoms from moving relative to each other in an unphysical manner. The bonds, angles and more complex atom relations such as impropers and dihedrals within molecules, are modelled using a set of potential functions that depend on their individual parametrisations.

2.1.2. Velocity-Verlet integration

For the systems of interest, Newton's second law as given by equation (2.5) cannot be solved exactly. Therefore, to determine the time evolution of the particle trajectories, a numerical integration scheme is required. In order to preserve (a discrete version of) the total energy as much as possible, a symplectic integrator is chosen, namely the velocity-Verlet algorithm.[36] The solutions of a symplectic integrator reside on a symplectic manifold. For solving a Hamiltonian system this is important, as the solutions to Hamiltonian equations naturally do so, and on which the total energy must be constant for the true solution. Due to its property of symplecticity, the velocity-Verlet algorithm is relatively stable and its numerical drift relatively low. Therefore, it is better suited for performing long MD simulations than other well-known integrators like the Euler scheme or the Runge–Kutta scheme.[37]

The velocity-Verlet algorithm can be derived, by performing and combining Taylor expansions of $\mathbf{x}(t + \Delta t)$, $\mathbf{x}(t - \Delta t)$, $\mathbf{v}(t + \Delta t)$ and $\dot{\mathbf{v}}(t + \Delta t)$, as

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \Delta t \mathbf{v}(t) + \frac{\Delta t^2}{2}\mathbf{F}(\mathbf{r}(t)) \tag{2.6}$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{h}{2}(\mathbf{F}(\mathbf{r}(t + \Delta t))) + \mathbf{F}(\mathbf{r}(t)) \tag{2.7}$$

where force vector $\mathbf{F}$ is given by eq. 2.5. Eq. (2.6) gives the particle's position at one time step Δt later, and eq. (2.7) gives a particle's velocity at one time step h later.

2.1.3. Periodic boundary conditions

Most preferably the simulated MD system would contain a macroscopic number of atoms. Computationally, however, this can not be achieved. Instead, in order to emulate a larger system, the MD system is contained by periodic boundary conditions. Essentially, the simulation box is thus surrounded by copies of itself. This has consequences for the particles moving within the simulation box. When a particle moves through one end of the simulation box, it enters the box from the opposite side. To avoid self interacting particles, and particles experiencing a force from the same particle from multiple directions, a cutoff distance is enforced on the evaluation of the interatomic potential functions.

Instead of calculating the force every particle experiences due to every other particle, a particle in the periodic system only experiences forces from particles within a certain cutoff distance from itself. This way, a larger system can be mimicked, while only calculating the trajectories of a limited number of atoms. The choice for the cutoff distance is generally justified by looking at how quickly the LJ-potential falls off as a function of the distance between particles. To prevent a particle from interacting with the same particle more than once, the choice for the cut-off distance should always be smaller than half the length of the smallest simulation box dimension.

The hard cut-off distance does not apply to the Coulombic force evaluation, as this falls off much slower than the other relevant forces. However, after the cut-off distance, the Coulombic evaluation is usually replaced using a long-range solver. This is an algorithm designed to efficiently approximate the force.
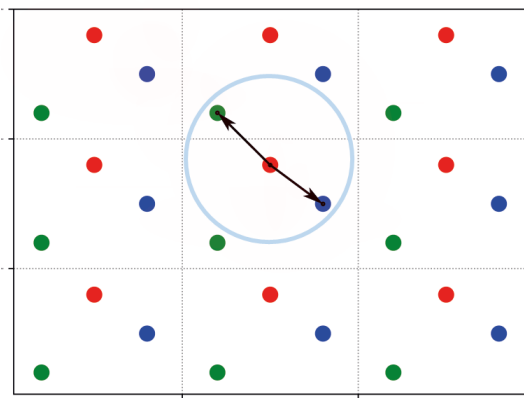

Figure 2.1: Depiction of periodic boundaries and the cutoff distance. Only particles within the cutoff distance from the red particle in the middle box, represented by the blue circle, can interact with it. So, within the middle box, the red particle experiences a force only from the blue particle in the same box and the green particle in the box above. The green particle in the middle box does not contribute to the force on the red particle.

2.1.4. Force fields

As outlined in the previous sub-sections, an MD simulation depends on many parameters that need to be set for all used atom types. Firstly, the masses and partial charges of the used atom types need to be set. Secondly, some functional forms for the interaction potential need to be chosen, and the pair coefficients between atoms, together with the cut-off distance, need to set. Lastly, for molecules, the required bond, angle, improper and dihedral functional forms and corresponding fitting parameters need to be specified.

All these parameters are provided by 'force fields', parameter sets made for specific atoms/molecules, which are generally supposed to be used under similar conditions as under which they were created. In unpolarisable force fields, which are most commonly used, the partial charges are taken to be constants that do not change during the simulation.

2.2. Density functional theory

Density functional theory (DFT) is used to investigate the electronic structure of many-body systems. In this section, a step-by-step introduction into the basics of DFT is provided. Throughout this section, Hartree atomic units ($e = \hbar = m_e = 4\pi\epsilon_0 = 1$) are used. Basic knowledge of functionals and functional derivatives is recommended. As a good introduction into functionals in the context of DFT, the short exposition given in Appendix A of the book by (Parr and Yang, 1994)[38] is recommended.

2.2.1. The many-body problem

Without external perturbations, for a system of n electrons and N nuclei, the time-independent Schrödinger equation for the many-body wave function $\Psi\left(\{\mathbf{r}_i, \mathbf{R}_I\}\right)$, can be written as

$$\hat{H}\Psi\left(\{\mathbf{r}_i, \mathbf{R}_I\}\right) = E\Psi\left(\{\mathbf{r}_i, \mathbf{R}_I\}\right), \tag{2.8}$$

where Hamiltonian operator $\hat{H}$ is given as

$$\hat{H} = \sum_{i=1}^{n}\sum_{j>i}^{n}\frac{1}{||\mathbf{r}_i - \mathbf{r}_j||_2} + \sum_{i=1}^{n}\sum_{I=1}^{N}\frac{-Z_I}{||\mathbf{r}_i - \mathbf{R}_I||_2} + \sum_{I=1}^{N}\sum_{J>I}^{N}\frac{Z_I Z_J}{||\mathbf{R}_I - \mathbf{R}_J||_2}$$
$$+ \sum_{i=1}^{n}-\frac{1}{2}\nabla_{\mathbf{r}_i}^2 + \sum_{I=1}^{N}-\frac{1}{2M_I}\nabla_{\mathbf{R}_I}^2. \tag{2.9}$$

Here, $\mathbf{r}_i$ and $\mathbf{R}_I$ denote the positions of the electrons and the nuclei, respectively. The nuclei masses are denoted by M_I (in atomic units), and Z denotes the atomic numbers. The first three terms in the Hamiltonian expression represent the electron-electron, electron-nuclear and nuclear-nuclear interactions, respectively. The final two terms describe the kinetic energies of the electrons and the nuclei. Solutions to this equation are sought-after, as they determine the static properties of the system.

2.2.2. Born-Oppenheimer approximation

In order to simplify eq. (2.9), the Born-Oppenheimer (BO) approximation is employed, that allows for the separation of the motion of the nuclei and that of the electrons.[39] In the BO approximation, a static arrangement of the nuclei is assumed. The physical basis for this approximation is the fact that the mass of a nucleus is multiple orders of magnitude larger than the mass of an electron. As a consequence hereof, electrons move and respond to forces much more quickly than the nuclei. By neglecting nuclei motion, the BO approximation essentially assumes that the electrons in the system are able to adapt to changes in the nuclear positions instantaneously.[40]

As a consequence of the BO approximation, the time-independent Schrödinger equation can be given as

$$\left[\sum_{i=1}^{n}\sum_{j>i}^{n}\frac{1}{||\mathbf{r}_i - \mathbf{r}_j||_2} + \sum_{i=1}^{n}\sum_{I=1}^{N}\frac{-Z_I}{||\mathbf{r}_i - \mathbf{R}_I||_2} + \sum_{i=1}^{n}-\frac{1}{2}\nabla_{\mathbf{r}_i}^2\right]\Psi\left(\{\mathbf{r}_i, \mathbf{R}_I\}\right) = E\Psi\left(\{\mathbf{r}_i, \mathbf{R}_I\}\right). \tag{2.10}$$

Although the motion of nuclei is now neglected, the wave function still depends on the positions of the nuclei. The potential energy due to nuclear-nuclear interaction is merely a constant, and plays no relevant role in the DFT framework. Therefore, it is left out of the Hamiltonian, which reduces to the so-called electronic Hamiltonian. A generalised notation of this expression can be given as

$$\left[\hat{U} + \hat{V} + \hat{T}\right]\Psi = E\Psi, \tag{2.11}$$

where the three terms in the l.h.s. of equation (2.10) are represented by $\hat{U}$ (electron-electron interaction energy), $\hat{V}$ (potential energy from external field, like the electron-nuclear interaction previously mentioned) and $\hat{T}$ (kinetic energy) operators, respectively.

2.2.3. Hohenberg-Kohn theorems

The expression given in eq. (2.11) is very commonly used in quantum chemistry for a multi-electron system. Because of the electron-electron interaction ($\hat{U}$), it cannot be separated into single-particle equations. In order to reduce the required computational effort, DFT attempts to solve a similar problem, by mapping the many-body problem with the problematic $\hat{U}$ term, onto a single-body problem. In DFT, the total energy of the system is treated as a functional of the electron density $\rho(\mathbf{r})$.[40] By not explicitly solving the system for a 3N-dimensional wave function (4N if spin was to be taken into account), but rather using a single 3 dimensional density variable, this approach reduces the computational cost of finding solutions significantly.[41][42]

In what are conventionally known as the Hohenberg-Kohn (HK) theorems, it was demonstrated that considering only the density suffices to characterise any physical property of a quantum mechanical system.[43] These theorems, as given below, provide a framework to find the ground state of any electronic system under the influence of an external potential through the electron density.

- HK-1: The external potential is a unique functional, except for a constant, of the electron density of a non-degenerate ground state.

- HK-2: The electron density obeys a variational principle.

To understand the consequences of these theorems, first the total energy functional is written in terms of some external potential v_{ext} and some functional F_{HK} of the electron density, as

$$E[\rho] = \int \rho(\mathbf{r})v_{ext}(\mathbf{r})d\mathbf{r} + F_{HK}[\rho]. \tag{2.12}$$

The so-called Hohenberg-Kohn functional F_{HK}, with corresponding operator $\hat{F_{HK}}$, is a generalised representation of the energies corresponding to the $\hat{U}$ and $\hat{T}$ operators shown in eq. (2.11). This functional is said to be universal, i.e., it does not depend on the system under study; it is the same for all n-electron systems. Since the F functional is universal, the Hamiltonian of the system is completely defined by the number of electrons in the system and the external potential. HK-1 (by means of a relatively straight-forward *reductio ad absurdum*) shows that the external potential itself is uniquely determined by the electron density. From the first theorem it follows that the ground state energy E_0 of the system can be formulated, not only as a functional of some wave function, but also as a universal functional of the electron density

$$E_0 = \langle \Psi | \hat{H} | \Psi \rangle = E[\rho]. \tag{2.13}$$

After obtaining this functional of the electron density, HK-2 provides a way to actually obtain the ground state energy by establishing a variational principle. Consequently, the ground state energy can be obtained by means of an energy minimisation with respect to the density, as the density that minimises the total energy must be the ground state density. This can be expressed as

$$E_0 = E[\rho_0] \leq E[\rho]. \tag{2.14}$$

Naturally, this minimisation should happen under the constraint that $\int \rho(\mathbf{r})d\mathbf{r} = n$.

2.2.4. The Kohn-Sham equations

There are different approaches in DFT for actually solving the ground state. Commonly used is the formulation of DFT provided by Kohn and Sham.[44] In Kohn-Sham DFT the system of interacting electrons in an external potential is reduced to a fictitious system of non-interacting electrons moving in an effective potential. Importantly, this reduces the Hamiltonian to a sum of single-particle operators.

Correspondingly, the electrons in this fictitious reference system are described by single-electron Kohn-Sham orbitals ϕ_i, which together are taken to yield the same ground state electron density as the interacting system. In principle, Kohn-Sham orbitals are mathematical objects used for solving the system, lacking strict physical meaning (even though they can prove useful quantities for interpretation[45][46]). Consequently, the wave function ψ^{KS} of the fictitious n-fermionic system can be described by a Slater determinant constructed from the set of Kohn-Sham orbitals. The density for the n-electron system can be expressed as a sum over the orbitals squared

$$\rho(\mathbf{r}) = \sum_{i}^{n} |\phi_i(\mathbf{r})|^2. \tag{2.15}$$

For the fictitious system, the Hohenberg-Kohn functional from eq. (2.12) is expressed as

$$F_{HK}[\rho] = T_s[\rho] + E_H[\rho] + E_{XC}[\rho]. \tag{2.16}$$

Here, the first two terms represent the non-interacting Kohn-Sham kinetic energy T_s and the Hartree electrostatic self-repulsion of the electron density E_H. Both terms do not represent the exact kinetic and electron-electron interaction energies. The exchange-correlation functional E_{XC} is defined to account exactly for all missing interactions in the non-interacting representation of the system due to exchange and correlation effects, as

$$E_{XC} = (T - T_s) + (E_{ee} - E_H), \tag{2.17}$$

where T and E_{ee} give the exact kinetic energy and electron-electron interaction potential, respectively. In other words, the E_{XC} functional contains the sum of corrections to all (quantum) effects not captured by either E_T or E_H. Now, the total energy can be written as

$$E[\rho] = T_s[\rho] + \int \rho(\mathbf{r}) v_{ext}(\mathbf{r}) d\mathbf{r} + E_H[\rho] + E_{XC}[\rho]. \tag{2.18}$$

In order to find a set of equations that can be used to solve the system, a minimisation of the total energy is performed. For this purpose, the following Lagrangian is introduced[47]

$$\mathcal{L}(\{\phi_i\}) = E(\{\phi_i\}) - \sum_{i=1}^{N} \epsilon_i \left(\int \phi_i^*(\mathbf{r})\phi_i(\mathbf{r}) d\mathbf{r} - 1 \right). \tag{2.19}$$

Here $E(\{\phi_i\})$ represents the total energy, as given by eqs. (2.18), only now written in terms of the set of orbitals $\{\phi_i\}$, via eq. (2.15). ϵ_i is the Lagrange multiplier associated with the constraint on the orbitals of being orthonormal, given as

$$\int \phi_i^*(\mathbf{r})\phi_i(\mathbf{r}) d\mathbf{r} = 1. \tag{2.20}$$

To find the ground state, the given Lagrangian should be stationary with respect to variations of the orbitals ϕ_i, i.e.,

$$\frac{\delta \mathcal{L}}{\delta \phi_i^*} = 0, \tag{2.21}$$

where the l.h.s. denotes the functional derivative of the Lagrangian. By computing this functional derivative, it is found that the orbitals satisfy a (coupled) set of single-particle Schrödinger-like equations, known as the Kohn-Sham equations

$$\left(-\frac{1}{2}\nabla_{\mathbf{r}}^2 + v_H(\mathbf{r}) + v_{XC}(\mathbf{r}) + v_{ext}(\mathbf{r}) \right) \phi_i(\mathbf{r}) = \epsilon_i \phi_i(\mathbf{r}). \tag{2.22}$$

In this expression, v_H corresponds to the Hartree potential, given as

$$v_H([\rho]; \mathbf{r}) = \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}, \qquad (2.23)$$

and v_{XC} is simply defined as the functional derivative of the exchange-correlation energy with respect to the density as

$$v_{XC}([\rho]; \mathbf{r}) = \frac{\delta E_{XC}[\rho]}{\delta \rho}(\mathbf{r}). \qquad (2.24)$$

The problem is reduced to an eigenvalue problem, in which the goal is to find the orbitals ϕ_i with (eigen-)energies ϵ_i.

2.2.5. Application of the theory

In practical calculations, usually orbitals ϕ_i are constructed as a linear combination of a set of other functions $\{\chi_\mu\}$, called basis functions, as

$$\phi_i = \sum_\mu c_\mu^i \chi_\mu, \qquad (2.25)$$

where $\{\chi_\mu\}$, for example, are taken to be Gaussian-type basis functions centered on the nuclei. Thus, finding the orbitals ϕ_i amounts to calculating their respective orbital coefficients c_μ^i. Since the v_H and v_{XC} terms in eq. (2.22) depend on the density, which itself depends on the orbitals ϕ_i, which in turn depend on the potential, the Kohn-Sham equations have to be solved self-consistently. This usually results in an iterative solving scheme, referred to as the self-consistent field (SCF) method. First the Kohn-Sham equations (eq. (2.22)) are solved for a given input density ρ^{in}. Subsequently, a new output density ρ^{out} is calculated using eq. (2.15). Finally, the iteration scheme is halted if ρ^{in} and ρ^{out} are in sufficiently good agreement according to some pre-set criterion. If not, a new iteration is started using a new input density, usually formed by mixing ρ^{in} and ρ^{out}.

If the exchange-correlation functional would be exact, the exact electron density of the system could be obtained. However, exact analytic expressions of the exchange-correlation functional E_{XC} are unknown, so approximate functionals must be introduced. The two most common approximations are the local-density approximation (LDA) and the generalised gradient approximation (GGA).

In the LDA, the exchange-correlation functional depends only on the density at the position where it is evaluated. Usually, exchange-correlation functional is split into an exchange part and a correlation part, and each parametrisation is individually derived from the homogeneous electron gas model. In the GGA, the exchange-correlation energy depends on the density at the point of evaluation (local) and on the gradient of the electron density (non-local), such that non-local changes in the density around the position of evaluation can be taken into account.[48][49]

The given formulation of DFT does not take into account relativistic effects, as spin does not appear naturally in the (non-relativistic) Schrödinger equation. In the case that spin polarisation plays an important role in the system, spin would have to be introduced as an additional degree of freedom to compensate for this, by letting the single-particle wave functions describe the spin state. In that case, usually, the total density is split up in a up and down-state density. Next, the exchange-correlation functionals must be extended accordingly, to take into account the spin projections. For brevity, all corresponding equations and considerations are not going to be provided here. Jambrina et al.[50] provide an introduction to the subject.

2.3. Self-consistent charge density functional tight binding

Density functional tight-binding (DFTB) is an approximation of DFT, in which the Kohn-Sham equations are reduced to a form of tight-binding. By means of a second-order expansion of the Kohn-Sham total energy, a charge self-consistent treatment is enabled. The resulting method is self-consistent charge DFTB (SCC-DFTB), which is the most widely applied version of the DFTB method. Below, the SCC-DFTB method is introduced.

The notation used in this section is consistent with the previous discussion of DFT in section 2.2. i and j represent electron indices, whereas I and J are atom indices. μ and ν are indices of basis states. r and R denote electron and atom positions, respectively. Again, Hartree atomic units are used throughout this section.

In DFTB, the Hamiltonian elements are taken to be the principal parameters of the method (following a tight-binding formalism). To obtain a set of equations that can be parametrised in such a way, the total energy of the non-interacting Kohn-Sham system is approximated. Firstly, these approximations will be introduced. Subsequently, the tight-binding formalism is expanded upon.

2.3.1. Expansion of the total energy

In order to derive the relevant equations for SCC-DFTB, first the expression for the total energy of the non-interacting Kohn-Sham system is given, as before, as

$$E[\rho] = T_S[\rho] + \int v_{ext}(\mathbf{r})\rho(\mathbf{r})d\mathbf{r} + E_H[\rho] + E_{XC}[\rho] + E_{core}, \tag{2.26}$$

wherein now only the E_{core} term contains the contributions to the energy of the nuclear-nuclear interactions. Below it is shown that this terms plays an important role in the parametrisation of DFTB. This expression can be rewritten as

$$E[\rho] = \sum_i \langle \phi_i | \left(-\frac{1}{2}\nabla_{\mathbf{r}}^2 + v_{ext} \right) | \phi_i \rangle + \frac{1}{2}\int\int d\mathbf{r}d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} + E_{XC}[\rho] + E_{core}. \tag{2.27}$$

DFTB is an approximation of DFT, and the SCC-DFTB method is derived from DFT by a second-order expansion of the given DFT total energy with respect to charge density fluctuations $\delta\rho$ around some reference density ρ_r.[51] The result allows the energy functional to be written as a functional of the charge fluctuation, that can be split into three parts

$$E[\delta\rho] = E_{BS}[\delta\rho] + E_{rep} + E_{coul}[\delta\rho]. \tag{2.28}$$

Firstly, E_{BS} denotes the so-called band-structure energy

$$E_{BS}[\delta\rho] = \sum_i \langle \phi_i | \left(-\frac{1}{2}\nabla_{\mathbf{r}}^2 + v_{ext} + \int d\mathbf{r}' \frac{\rho_r(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} + v_{XC}[\rho_r] \right) | \phi_i \rangle = \sum_i \langle \phi_i | H^{KS}[\rho_r] | \phi_i \rangle, \tag{2.29}$$

in which the Kohn-Sham Hamiltonian $H^{KS}[\rho_r]$ itself is independent of the charge fluctuation.

2.3.2. Repulsive energy term

Secondly, the repulsive energy E_{rep}, expressed as

$$E_{rep} = -\frac{1}{2}\int\int d\mathbf{r}d\mathbf{r}' \frac{\rho_r(\mathbf{r})\rho_r(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} + E_{XC}[\rho_r] - \int \mathbf{r}v_{XC}[\rho_r]\rho_r + E_{core}, \tag{2.30}$$

gets its name from the present nuclear-nuclear interaction term E_{core}. The first three terms correspond to the correction for the double counting terms from the Coulombic and exchange-correlation contributions contained in the $H_r[\rho_r]$ Hamiltonian (Coulombic interaction between two electrons, for example,

is currently counted for both electrons experiencing it). E_{rep} is usually approximated in its entirety as a sum of terms over atom pairs, where each term depends solely on the type of elements and the distance between them. Generally, this can be written as

$$E_{rep} = \sum_{I<J} v_{rep}^{IJ}(R_{IJ}). \tag{2.31}$$

To get, from the Hartree term, to this expression from, the total volume over which is integrated is divided up into atomic volumes, through which the integral can be written as a sum over atom pairs.[42] Similar reasoning is applied to the remaining terms. The pair-wise functions, v_{rep}^{IJ}, are obtained by means of a fitting procedure in a higher-level theoretical computation program, like DFT.

2.3.3. Charge fluctuations term

The third term, E_{coul}, represents the Coulomb and exchange-correlation contributions to the total energy from the charge density fluctuations and is expressed as

$$E_{coul}[\delta\rho] = \frac{1}{2} \int \int' \left(\frac{\delta^2 E_{XC}[\rho_r]}{\delta\rho\delta\rho'} + \frac{1}{|\mathbf{r}-\mathbf{r}'|} \right) \delta\rho\delta\rho'. \tag{2.32}$$

The two terms on the r.h.s. only appear in the second order expansion of the total energy. Their incorporation forms the difference between the SCC-DFTB method and the traditional DFTB approach, in which they are neglected. As in the SCF procedure in DFT, in SCC-DFTB the effective Hamiltonian and the charge distribution are kept mutually consistent by means of an iterative scheme similar to the one described for DFT. To approximate E_{coul}, the continuous electronic density is first discretised into a set of atom centered contributions, by means of dividing the real space $\mathcal{V}$ into arbitrary atomic volumes $\mathcal{V}_I$.[52] This way, the charge fluctuation $\delta\rho((r)$ and the additional electron population on an atom I, Δq_I, are associated through

$$\Delta q_I = \int_{\mathcal{V}_I} \delta\rho(\mathbf{r})d\mathbf{r}, \tag{2.33}$$

and

$$\delta\rho(\mathbf{r}) = \sum_I \Delta q_I \delta\rho_I(\mathbf{r}). \tag{2.34}$$

In the latter expression, the $\delta\rho_I$ functions are normalised over the atomic volume, such that $\int_{\mathcal{V}_I} \delta\rho_I(\mathbf{r})d\mathbf{r} = 1$. Furthermore, these functions are commonly taken to be spherically symmetric functions (i.e., any angular deformation is neglected) with a Gaussian profile[42]

$$\delta\rho_I(\mathbf{r}) = \frac{1}{(2\pi\sigma_I^2)^{1.5}} \exp\left\{ -\frac{\mathbf{r}^2}{2\sigma_I^2} \right\}, \tag{2.35}$$

With these assumptions, the Coulomb energy is a function of two spherically symmetric Gaussian charge distributions. By subsequently evaluating the self-interaction and cross-interaction individually, and neglecting non-dominating terms, ultimately an expression is found for E_{coul}, of the form

$$E_{coul} = \frac{1}{2} \sum_{IJ} \gamma_{IJ}(R_{IJ})\Delta q_I \Delta q_J, \tag{2.36}$$

where

$$\gamma_{IJ}(R_{IJ}) = \begin{cases} U_I, & I = J. \\ \frac{erf(C_{IJ},R_{IJ})}{R_{IJ}}, & I \neq J, \end{cases} \tag{2.37}$$

in which

$$U_I = \frac{1}{\sqrt{\pi}\sigma_I}, \tag{2.38}$$

and

$$C_{IJ} = \sqrt{\frac{1}{2(\sigma_I^2 + \sigma_J^2)}}. \tag{2.39}$$

Eq. (2.38) couples U_I, that represents the atomic Hubbard potential, and the so-far unspecified parameter σ_I. The Hubbard potential, also known as (twice) the chemical hardness, describes how much the energy of a system changes upon adding or removing electrons. It can be estimated as the difference between the ionisation potential and the electron affinity of a type of atom. Consequently, via the coupling, the Hubbard potential determines all charge transfer energetics. Usually for DFTB, it is either found from standard tables, or calculated using DFT.[51]

2.3.4. Tight binding formalism and application

So far tight binding aspect of DFTB has not been referenced. Density functional tight binding assumes tightly bound electrons, hence the name. In essence, this assumes that electrons on an atom only have limited interaction with states and potentials on surrounding atoms, such that the wave function of the bound electrons are similar to the atomic orbitals of the quasi-free atom to which they belong.[53] To this extent, a minimal linear combination of atomic orbital (LCAO)-type basis set is used to expand the single-particle wave functions ϕ_i, i.e., each valence orbital μ is defined by a single basis function η_μ.[54] Consequently, single-particle wave functions ϕ_i are expanded as

$$\phi_i(\mathbf{r}) = \sum_\mu c_\mu^i \eta_\mu(\mathbf{r}). \tag{2.40}$$

Furthermore, throughout the DFTB method, only valence orbitals are considered in order to reduce computational effort, and since core electrons are generally chemically inactive. Using only $1s$ for hydrogen, and using only $2s, 2p_x, 2p_y, 2p_z$ for first row elements of the periodic table, are examples of basis sets resulting from these assumptions. These basis functions, representing localised atomic orbitals, are determined using DFT, where an additional harmonic potential is taken into account to localise the orbitals which punishes an electron for moving away from its confinement radius.[55]

Using the expansion given by eq. (2.40), the final energy expression becomes

$$E = \sum_i \sum_{\mu\nu} c_\mu^{i*} c_\nu^i H_{\mu\nu}^{KS} + \frac{1}{2} \sum_{IJ} \gamma_{IJ}(R_{IJ}) \Delta q_I \Delta q_J + \sum_{I<J} v_{rep}^{IJ}(R_{IJ}), \tag{2.41}$$

where $H_{\mu\nu}^{KS} = \langle \phi_\mu | H^{KS}[\rho_r] | \phi_\nu \rangle$ and the overlap matrix element is $S_{\mu\nu} = \langle \phi_\mu | \phi_\nu \rangle$. The diagonal $H_{\mu\mu}$ terms represent the effective single-electron atomic energy levels that are associated with the valence shell atomic orbitals. The so-called hopping integrals, the off-diagonal elements, describe the effects due to electron delocalisation.

By means of an energy minimisation by variation of $\delta(E - \sum_i \epsilon_i \langle \phi_i | \phi_i \rangle)$ (where terms ϵ_i function again as the Lagrange multipliers constraining the wave function norms), the following set of secular equations is found for all i and μ

$$\sum_\nu c_\nu^i (H_{\mu\nu} - \epsilon_i S_{\mu\nu}) = 0, \tag{2.42}$$

where

$$H_{\mu\nu} = H_{\mu\nu}^{KS} + \frac{1}{2} S_{\mu\nu} \sum_K (\gamma_{IK} + \gamma_{JK}) \Delta q_K, \text{ for } \mu \in I, \nu \in J. \tag{2.43}$$

Eq. (2.42) is the SCC-DFTB equivalent of the Kohn-Sham equations in DFT. As shown, the SCC-DFTB electronic structure is not influenced by the repulsive energy, other than through its effect on what the stable geometry would be in a DFTB-MD simulation. Indeed, the repulsive energy is logically dropped out of the equation as it does not depend on the orbitals; its importance lies in the force calculations between atoms in DFTB.

The matrix elements $H_{\mu\nu}$ (and overlap elements $S_{\mu\nu}$) are taken to be the principal parameters of the method, following the tight-binding formalism. Because the minimum basis set is used, the Hamiltonian and overlap matrices can be pre-calculated using DFT and stored. Along with the repulsive energy, they are stored in so-called Slater-Koster files, wherein the data is formatted in an atom type pair-wise fashion (C-C, C-H etc.). Using these parameters, DFTB does not have to perform any integral evaluation during run time.

In SCC-DFTB, eq. (2.42) has to be solved self-consistently, like in DFT. However, instead of guessing a density, an initial guess is made for the charge fluctuations Δq. From this a Hamiltonian $H_{\mu\nu}$ is obtained. Using this Hamiltonian, subsequently new orbital coefficients c_{μ}^i can be calculated. These determine a new Δq, and the scheme can be iterated over.

2.4. Neural networks

Artificial neural networks (referred to as neural networks or NNs), or multilayer perceptrons as they are alternatively known, are computing systems inspired by biological neural networks. They generally consist of an ordered set of layers. The first layer of a neural network is called the 'input layer', and the final layer is called the 'output layer'. All layers in between are called 'hidden layers'. Every layer consists of a set of nodes called 'neurons'. Like in a biological brain, these neurons are connected to each other in some way and can transmit signals to other connected neurons The signals in the context of neural networks are real numbers, that are processed by a neuron into an output. The output of a neuron is generally some non-linear function (e.g., ϕ) of the sum of its inputs.

By moving consecutively from the input layer through the hidden layers and finally the output layer, an output is computed from the input. Using a set of observations to 'train' on, in the form of input and corresponding output data, the goal of a neural network is to make an accurate prediction given an arbitrary input. Many different types of neural networks have been developed over the past decades. In this thesis, fully connected networks and the convolutional neural networks are used. These, and other general neural network terminology and techniques, are discussed in this section.

2.4.1. Fully connected layers

In a fully connected network (FCN) all layers are fully connected, i.e., all neurons from one layer are connected to all neurons in the consecutive layer. A schematic representation of such a network can be found in figure 2.2.

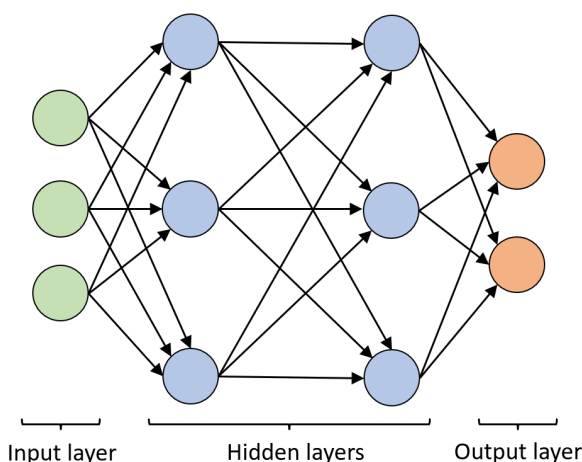Input layer   Hidden layers   Output layer

Figure 2.2: Schematic representation of a fully connected network. The neurons (green) on the left side represent the input layer, whereas the neurons on the right side (red) represent the output layer. All neurons in between (blue) form the hidden layers. All neurons in one layer, are connected to all neurons in the successive layer.

To provide more insight into what exactly happens in such a network, a mathematical description of a fully connected network with one single hidden layer will be given next. In the description, there are D input variables, M neurons in the hidden layer, and N outputs. Given two consecutive layers, $l^{(k-1)} \in \mathbb{R}^{v^{(k-1)} \times 1}$ and $l^{(k)} \in \mathbb{R}^{v^{(k)} \times 1}$ (where v represents the number of neurons in the layer), the weight matrix connecting the two is given as $w^{(k)} \in \mathbb{R}^{v^{(k-1)} \times v^{(k)}}$. Weight matrix $w^{(1)}$ connects the input layer with the hidden layer, whereas weight matrix $w^{(2)}$ connects the hidden layers with the output layer. Firstly, M linear combinations of input variables x_i are constructed, which are then transformed using some activation function $h(\cdot)$ into the hidden layer. The consequential output of the hidden layer is given as

$$o_j^{(1)} = h\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + b_j^{(1)}\right) \qquad \text{for } j = 1,...,\text{M} \tag{2.44}$$

where $o_j^{(1)}$ is the output of the hidden layer, $h(\cdot)$ is some differentiable non-linear activation function and $w_{(ji)}^{(1)}$ and $b_{(j)}^1$ give the weight and bias parameters from the hidden layer. Subsequently, the output of the hidden layer flows through the output layer, is

$$o_n^{(2)} = \sum_{j=1}^{M} w_{nj}^{(2)} o_j^{(1)} + b_n^{(2)} \qquad \text{for } n = 1,...,\text{N} \tag{2.45}$$

where $o_n^{(2)}$ represents the final output of the network. To summarise, the final output of the FCN with one hidden layer can be written as one equation, given as

$$o_n^{(2)} = \sum_{j=1}^{M} w_{nj}^{(2)} h\left(\sum_{i=1}^{D} w_{ji}^1 x_i + b_j^{(1)}\right) + b_n^{(2)} \qquad \text{for } n = 1,...,\text{N} \tag{2.46}$$

In this way, the fully connected neural network can be seen as an extension of a classic linear model for regression and classification problems, which is based on a linear combination of fixed non-linear basis functions ϕ.[56] A generic form of such a model, can be given as

$$y(\mathbf{x}, \mathbf{w}) = h\left(\sum_{j=1}^{N} w_j \phi_j(\mathbf{x})\right) \tag{2.47}$$

The difference with a FCN is that the basis functions are not fixed, but depend on adaptive parameters.

2.4.2. Activation functions
As described in section 2.4.1, activation functions are functions that are applied at the output of a neuron. Since the multiplication with weights and the subsequent addition of a bias in itself is a linear operation, a layering of neurons would never be able to produce non-linear behaviour. Therefore, generally non-linear functions are chosen as activation function, to allow for non-linear regression problems to be solved, or to allow classification problems to be solved that are not linearly separable.

Many activation functions have been developed for neural networks. The ReLU activation function is probably the most widely used activation function.[57]

$$\text{ReLU}(x) = x^+ = \max(0, x) \tag{2.48}$$
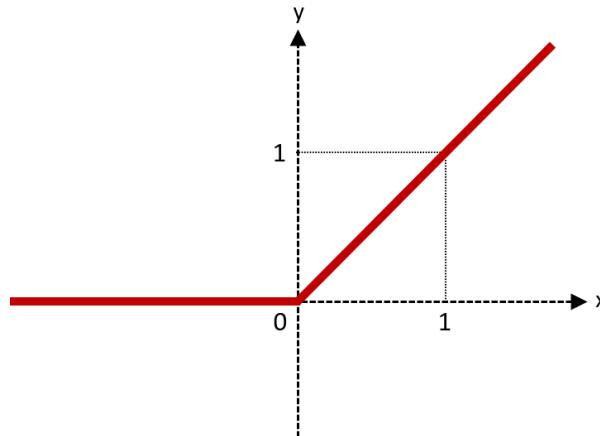
A plot can be found in figure 2.3.

Figure 2.3: The ReLU activation function, a commonly used activation function in neural networks.

Other commonly used activation functions are the sigmoid activation function and the tanh activation function. Both functions, however, saturate across most of their domain, i.e., they saturate to their maximum value when the input is very positive, and saturate to their minimum value when the input is very negative. In other words, their derivatives decrease quickly as the input value moves towards the extremes. Therefore, these activation functions are only strongly sensitive when their input is near zero.[58] As will be shown in section 2.4.4 about loss minimisation, this can lead to difficulties in the loss minimisation procedure (specifically in the back propagation step). To circumvent this problem, these activation functions are usually paired with the so-called cross entropy loss function in classification problems.

2.4.3. Loss functions

In the context of neural networks, a loss function is a mathematical function that represents a measure of the dissimilarity between the actually produced output of a network, and the output values that the network is seeking to reproduce, called the targets. The task of the neural network is to minimise the value of the loss function to improve the quality of the neural network prediction. The most commonly used loss function in the context of regression is the root-mean-square error (RMSE),

$$\mathcal{L}_{\text{RMSE}} = \sqrt{\frac{\sum_{i=1}^{N}\left(y_i^{predicted} - \hat{y}_i^{target}\right)^2}{N}}, \tag{2.49}$$

where N is the size of the set of predictions and targets.

2.4.4. Loss minimisation

Optimisation of the adaptable neural network parameters (like weights and bias terms) happens during the training stage. During the training stage the network is shown training data, input data and corresponding output (target) data, and a loss is calculated for each prediction on the input data. Firstly, for the network to subsequently be able to minimise this loss, the loss must be communicated back to all adaptable parameters in the network. Most commonly this is done using the backpropagation algorithm. Secondly, the parameters need to be changed accordingly. This is commonly done using stochastic gradient descent (SGD).

SGD is an iterative method for optimising a differentiable loss function. It is commonly expressed as

$$\mathbf{w} := \mathbf{w} - \eta \nabla_{\mathbf{w}} \mathcal{L}. \tag{2.50}$$

Given some initial trainable parameter w, SGD moves the value of $\mathbf{w}$ in the direction opposite to the derivative of the loss with respect to the parameter $\mathbf{w}$. The step size, corresponds to some parameter

η called the learning rate, multiplied by the absolute value of the derivative.

Each iteration, all adaptable parameters in a network are updated this way. In 'stochastic' gradient descent, the relevant derivatives are not calculated using the entire training data set, but rather using a subset thereof. This is done to lighten the computational burden, and thereby to speed up the iterations. In the context of neural networks, these subsets of the data set are called 'batches'. The number of 'epochs' refer to the number of times the weights in the neural network have been updated using all training data, i.e., in one epoch, all batches, and thus all training data, is used once.

Various extensions of SGD have been developed to improve the performance of the optimisation. When the effective gradients are calculated from a tracked exponentially weighted moving average (EWMA) of the gradients during the optimisation, this is called 'momentum' in the context of SGD. RMSprop is an extension of SGD that smooths the zero-centered variance of noisy gradients using an EWMA of squared gradients. The Adam algorithm combines the momentum and RMSprop techniques.[59]

As previously stated, all gradients in the network are computed and communicated using the backpropagation algorithm. Instead of computing every gradient directly for all adaptable parameters, which would require redundant recalculations of individual derivatives multiple times, the backpropagation algorithm computes all gradients in the network only once. This is made possible by use of the chain rule. By computing the gradients one layer at a time and iterating backward from the output layer, the chaining of gradients allows for the efficient re-use of previously computed gradients such that none of the derivatives have to be calculated again.[58]

Some networks, especially deeper networks (networks with many hidden layers), can suffer from what are called 'vanishing gradients'. Since backpropagation computes gradients by the chain rule, most gradients in a deep network are products of many gradients. Consequentially, if those gradients are small, the gradient decreases exponentially with the number of layers, ergo the term 'vanishing gradients'. As a result, far away from the output layer, the adaptable parameters of the network can be effectively prevented from changing their values. This would slow down or even halt training of the network.[60]

As stated in section 2.4.2, some activation functions are more susceptible to this effect than others. Ideally a function that has a significant gradient everywhere is used, and for this reason the ReLU is so popular. Another solution to this problem and to allow for the use of deeper neural networks, comes in the form of 'skip connections' or 'residual connections'.[61] Skip connections are additional connections added to the network between the output of some layer and the input of another, where in between other layers are skipped. In this manner, the gradients from the deeper layers can be passed unchanged to the shallower layers. A schematic representation of the skip connection can be found in figure 2.4.
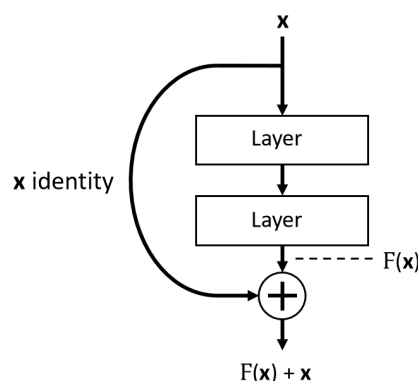

Figure 2.4: Schematic representation of a skip connection. Input vector **x** enters a block of two layers and is transformed into $F(\mathbf{x})$. Upon leaving the second layer, the initial input vector **x** is added to $F(\mathbf{x})$ unchanged. Input vector **x** has 'skipped' the two layers.

2.4.5. Convolutional layers

In the fully connected layer, every neuron from a layer is connected to every neuron in the consecutive layer. In doing so, the fully connected architecture disregards potential structure inherent to the training data. If the input was an image, represented by a 2D matrix, a fully connected network would treat pixels that are close-by and pixels far apart on equal footing. As a way to embed spatial structure into the layer itself, each output in a convolutional layer is only connected to a limited region in the input, called the 'receptive field'. A convolutional operation, applied in a convolutional layer, uses a so called 'kernel', a matrix of weights, that is characterised by width W and height H in the case of a 2D convolution. The kernel slides, as in a convolution, over the height and width dimensions of an input image. The contents of the sliding window are called 'patches', which correspond to the local receptive field. By multiplying the weight in the kernel by the input values in the patches, the output is computed. A schematic of the convolutional operation, is provided in figure 2.5.
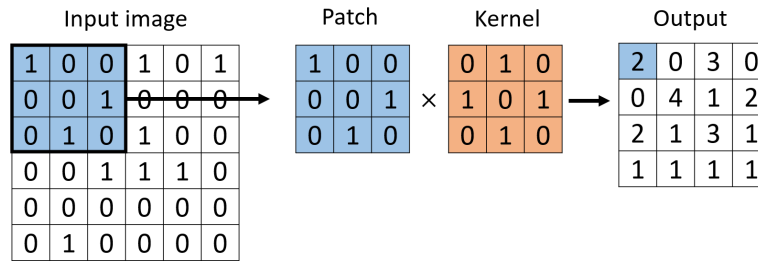

Figure 2.5: Schematic of a convolutional layer. The kernel slides along the over the height and width dimension of the input image. The contents of the sliding window, the image patch, is multiplied by the kernel weights to obtain the output. Because no patching was applied in this example, the output dimensions are not the same as those of the input image.

As it is described above, the 2D convolutional operation can be written as

$$y[i,j] = \sum_{m=-\lfloor \frac{k}{2} \rfloor}^{\lfloor \frac{k}{2} \rfloor} \sum_{n=-\lfloor \frac{k}{2} \rfloor}^{\lfloor \frac{k}{2} \rfloor} w[m,n] \cdot x[i+m, j+n] \qquad (2.51)$$

where k represents both the width and the height of the kernel and $\lfloor \cdot \rfloor$ is the floor operator. The convolutional operation, as given in eq. (2.51), can be extended in multiple ways. For example, the step size that the kernel makes between patches in either dimension can be increased. This step size is called the 'stride'. By choosing a larger stride, the dimensionality of the input can be reduced and the receptive field increased.

Moreover, to prevent border pixels from being lost (as was seen in figure 2.5), so as to keep the dimensions of the input, pixels can be added to the borders of an image. This is called 'padding'. Typically, a 'same padding' is applied, where the padded pixels are zeroes, and the applied number of padded pixels is two less than the corresponding kernel dimension. For example, the kernel size in figure 2.5 is three. In order to keep dimensions of the input, a padding could have been applied of one single pixel, one on either side of the image in both dimensions. By taking both padding and the value of the stride into account, output dimension D_{out} of a convolutional layer, can be written as

$$D_{out} = \lfloor \frac{D_{in} + 2 \times \text{padding} - \text{kernel size}}{\text{stride}} + 1 \rfloor, \qquad (2.52)$$

where the padding, the kernel size and the stride can be set for each individual dimension.

Just like for fully connected layers, usually a bias and an activation are applied at the output of the convolutional layer. Because not every input pixel gets assigned its own individual weight, like in a fully connected layer, but the same kernel is applied to the entire image, the weights in the architecture are shared by the pixels in the input image. One consequence of this is that the response of a convolutional

layer is translational-equivariant. For images, this means that, for example, if the input is moves a single pixel to the right, its representation in the output also moves a single pixel to the right. This property can be very beneficial in image classification problems, as for classification of a particular feature, it only matters that the network is able to recognise that the feature is somewhere on the image, and it should not matter where exactly on the input image it is located.

Real life images are usually not represented by a single 2D matrix of values. For example, an RGB image is represented by three separate such matrices. Each individual representation of the input is called a 'channel'. Convolutional layers can be extended from 1D and 2D to any multi-dimensional input. For example, in 3D convolutional layers, the kernel itself is also three dimensional, and moves not only through the height and width dimensions, but also in the depth dimension. Then, the number of channels refers to the number of 3D representations of the system that are used in the convolution.

While networks with one or two hidden layers might be able to provide the same performance as deeper (i.e., having more layers) networks, they generally require significantly more parameters (i.e., 'wider' layers with more neurons) to do so, increasing their complexity, and thus the amount of training data required. Therefore, a general observation is that deeper networks outperform shallower but wider networks, with the same parameter budget.[61][62] Because of the weight-sharing within convolutional layers, networks containing convolutional layers can be made much deeper than their fully connected counterparts, for the same parameter budget.

Convolutional layers form the core building blocks in convolutional neural networks (CNNs). Due to the advantages mentioned in this section, CNNs have long represented the state of the art in many computer vision problems.[62][63]

2.4.6. Pooling layers

Pooling layers can be used in convolutional networks to reduce the dimensionality of a layer and increase the receptive field of individual neurons. A 'max' pooling layer is a convolutional layer that outputs the maximum value within each patch, whereas an 'average' pooling layer outputs the average of the values within each patch. Finally there are global or 'adaptive' pooling layers that apply their effects on the entirety of each channel, and output only a single value. Global pooling layers are mainly placed before the final fully connected layer.

2.4.7. Batch normalisation

The distribution of inputs for any layer in a network changes during training time, as the adaptable parameters from the preceding layers are optimised.[64] A batch normalisation layer aims to normalise the output distributions of neurons, and make the (training of) the network more stable. By computing, for every batch on a per channel basis, the mean and variance across the batch dimension, a batch normalisation layer tries to learn how to normalise the learned features. The applied normalisation usually is based on an EWMA of the computed mean and variance. Trainable parameters are usually added as well, which allow for the batch norm layer to turn off the normalisation entirely, if deemed beneficial.

2.4.8. Regularisation

Overfitting is the phenomenon where a neural network is adapted in such a way that the error on the training set is decreased, but not for the benefit of the error on the test set. The network adapts to the particular training data too much, such that its generalisation power on test data usually is decreased. Consequently, a measure of the level of overfitting is the difference between the apparent training error and the true/test error. The root cause of overfitting is that the network is too complex, e.g., the network contains more parameters than can be justified based on the problem/data.

To illustrate overfitting, one could think about trying to fit a linear function using a quadratic model, when the training data of the linear function only consists of a few data points. With such a limited amount

of data, the quadratic model would probably be too 'complex' and would predict a parabolic shape far from being linear. To solve the overfitting problem, either the model complexity must be decreased, the amount of training data should be increased (by either obtaining more data, or by augmenting the already available data), or other techniques aimed at controlling overfitting should be applied in the modelling procedure. Many of such 'regularisation' techniques have been developed, all of which seek to discourage overly complex models.

Very commonly applied is L2 regularisation. In L2 regularisation the L2-norm of all weights is added as a penalty term to the loss function, expressed as

$$\mathcal{L}_R = \mathcal{L} - \lambda ||\mathbf{w}||_2^2. \tag{2.53}$$

Here $\mathcal{L}_R$ and $\mathcal{L}$ give the regularised and unregularised loss, respectively.

This form of regularisation is also called weight decay, as it encourages weights to decay to zero, unless supported by the data. Thus, it discourages complex models. L1 regularisation is a variant of this procedure, where the magnitude of all weights is added as penalty term.

By augmenting data, networks can also be made more robust to new inputs. By expanding the training with augmented data, invariance of a network to a set of transformations can be encouraged. For example, slightly rotating or adding some white noise to an image of handwritten letters that a network is trying to classify/'read', can help combat overfitting.[58] The shared-weight architecture of a convolutional layer can also be seen as a form of built-in regularisation, as this effectively decreases the complexity of the model.[56]

Finally, early stopping is another well-known regularisation technique. For many optimisation algorithms applied in training a neural network, the measured error on the training set is a non-increasing function (apart from some oscillations). However, it is often seen that this does not hold for the error on the test data. After some point, the test error often starts to increase as a result of an onset of overfitting in the network. Therefore, better generalisation behaviour can be expected if the training procedure is halted when the minimum test error seems to have been reached, and no further decrease is expected anymore. This is called early stopping.[56]

3

Methods

All simulation methods used in this thesis are described this chapter. Molecular dynamics simulations formed the basis of this thesis. A variety of MD systems was created and tested. The observed behaviour of these systems determined whether they were going to be used in the QM-MD coupled simulations. To investigate the effect of charge induction effects, the final results of these MD simulations also provided a reference to compare the QM-MD coupled results with. The molecular dynamics simulation is discussed in chapter 3.1.

DFTB simulations were used to simulate the induced charges on a single electrode present in the MD simulations. By alternating between MD and DFTB simulations and exchanging the results, coupled DFTB/MD (QM-MD) simulations were performed. In this way, the dynamics of the system could be modeled with the speed of the classical MD framework, while at the same time including a quantum mechanical description of the surface charges. Before the QM-MD simulations were performed, DFT simulations were used to verify the usability of the DFTB simulations. The DFT simulation is discussed in chapter 3.2 and the DFTB simulation is discussed in chapter 3.3.

The total simulation time in the coupled QM-MD simulations was dominated by the DFTB simulations, which therefore represented a bottleneck for the simulation time. To enable long term and/or large scale QM-MD simulations of this kind, it was attempted to train a variety of neural networks to reproduce the DFTB results. The specific implementation of neural networks in this thesis is described in chapter 3.4.

3.1. MD simulation

In this thesis all molecular dynamics simulations were performed using the open source classical molecular dynamics code LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator).[65] In the following subsections, it is explained how the used systems were generated, what force field parameter sets were used and under what conditions the simulations were performed.

3.1.1. Topology

The topologies used in this thesis, as well as the information about what atoms form what bonds, angles, and potential impropers and dihedrals with what other atoms, were mainly generated using predefined functions in LAMMPS. Two systems were created that need to be differentiated from each other. These were both similarly structured. They were made up by an electrolyte, consisting of water and a number of solutes, and two copper electrodes enclosing it on either side in the z-direction. Only the solutes differ per system. Both systems were periodic in x,y-direction and non-periodic in the z-direction (the dimension bound by the electrodes). In every simulation, a net charge of +4.0 e was assigned to the anode, and a net charge of -4.0 e was assigned to the cathode.

Table 3.1: Table specifying the total number of atoms of each type present, in each of the two MD systems used in this thesis. In the case of molecules, specifies the number of molecules.

	System 1	System 2
H_2O	2009	2009
K^+	37	37
Cl^-	0	37
HCO_3^-	37	0
CO_2	13	0
Cu	768	768

System 1

System 1 was created to imitate a carbon dioxide reduction system. The electrodes in system 1 were made up of three layers of (100) copper, which equals exactly one single layer of copper FCC unit cells. In total there were 484 atoms in each of these electrodes. The experimental lattice constant of 3.6149 Å[66] was used to generate these lattices, a number chosen as a compromise between the lattice constants following from the set of copper SK-files (used in DFTB+) and the chosen copper force field (used in LAMMPS).

Between the electrodes, the space was filled with water molecules and approximately 1M of dissolved potassium bicarbonate. This was done to imitate a carbon dioxide reduction system in which potassium bicarbonate was dissolved. Next to carbon dioxide, the other modelled molecules were bicarbonate molecules and potassium ions, as these make up the largest percentage of solvation products.[67] Table 3.1 describes precisely how many atoms of what type were present in each of the systems.

System 2

System 2 differs from system 1 only in the type of molecules that were added to the electrolyte: the bicarbonate ions from system 1 were replaced by chloride ions, and the carbon dioxide molecules were removed.

3.1.2. Force fields

To model water, the SPC/E force field[1] was used in all simulations in this thesis. Both systems identified in the previous subsection, used their own specific set of parameters. Below the used parameters sets are defined for both systems. In all MD simulations performed, a cutoff distance of 10 Å was chosen. Bonds and angles are always modelled using harmonic potential functions, where the stiffness and equilibrium position/angle are parameters dependent on the specific bond/angle in the molecule. For larger molecules, the specific impropers and dihedrals potentials depended on the used force fields. These are described below.

Set 1

For system 1, multiple different force fields were tried. Each of these is given here.
The copper electrodes were modelled using the force field by Heinz et al.[68]. Carbon dioxide was modelled using the EPM2[69]force field, which was made flexible using the bond and angle parameters given by Cygan et al.[70], as was previously successfully used by Silvestri et al.[71].

For the bicarbonate molecules two force fields were attempted. The first attempt was with the force field provided by Demichelis et al.[72]. Because this force field did not allow for mixing between bicarbonate and other atoms types than those in water (as it contained Buckingham potential parameters), the mixing interaction was improvised by recycling the parameters of previously specified interactions of other species. For example, the interaction of the carbon atom in a bicarbonate molecule with potassium, was copied from the interaction between carbon in a carbon dioxide molecule, and potassium. In this attempt the Lorentz-Berthelot combination rules were applied, and all intramolecular non-bonded interaction was set to zero. Because of noticable clustering between the potassium and bicarbonate ions in this attempt, the potassium force field was changed from that given by Yagasaki et al.[73] to that

given by Joung and Cheatham[74].

Since this approach did not solve the clustering issues, the bicarbonate force field was changed. The new force field parameters were generated using the LigParGen parameter generator.[75] Using the SMILES code of bicarbonate as input, the LigParGen generator generated bond, angle, improper, di-hedral and Lennard-Jones OPLS-AA force field parameters, and assigned partial charges based on the CM1A charge model (N.B., the charge is not scaled by 1.14, since bicarbonate is a charged ion).[76][77] In correspondence with the OPLS-AA force field, the used mixing rules were changed to geometric. Furthermore, the intramolecular non-bonded interaction between atoms more than two bonds away (1,4 interaction) was changed from zero to being scaled by a factor of 0.5.

Set 2
Potassium and chloride were modelled using the force field given by Joung and Cheatham[74], and copper was modelled using using the force field given by Heinz et al.[68]. Geometric mixing was used (as prescribed by the copper force field) for the interaction between copper and the ions, and all in-tramolecular non-bonded interaction was set to zero.

3.1.3. Conditions
Before each simulation, the systems were first initialised in the same way. To avoid unphysically large accelerations as a result of atoms being initialised too closely together, an energy minimisation was performed in which the time step was slowly increased from 0.0001 fs to 1 fs. The force on the x,y-directions on all electrode atoms was set to zero and the force in the z-direction was set to be averaged over the whole of each individual electrode. This was done to prevent the atoms from moving rela-tive to each other, which was thought to ease the learning process of the neural networks to be used later in this thesis. A force, equivalent to a pressure of 1 atm, was applied on both electrodes in the z-direction in the relative direction of the electrolyte. After running for two nanoseconds, the two elec-trodes were fixed in space, and the initialisation was completed. The final system size of systems 1 and 2 is 28.92x28.92x80.85 Å.A visualisation of system 2 after the 'initialisation' step, can be found in figure 3.1.
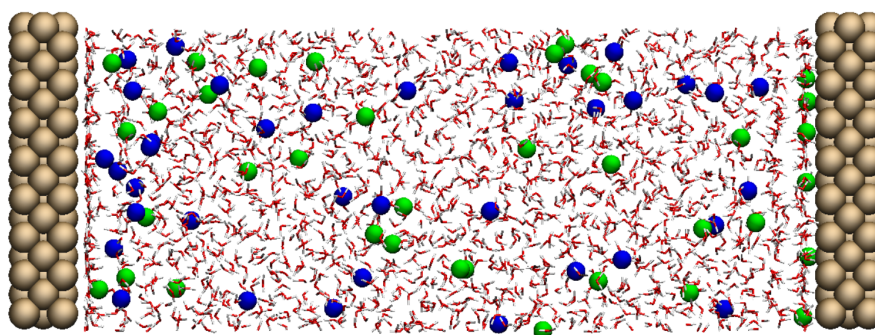

Figure 3.1: Depiction of system 2 after the 'minimisation' step. The copper electrodes are depicted with beige spheres, chloride and potassium are depicted by the blue and green spheres, respectively, and the oxygen and hydrogen in water are depicted using white and red tubes, respectively.

The time step that is acceptable was chosen based on the fastest relevant frequency in the simulated system. Often these are present in the covalent bonds. All hydrogen bonds within the water molecules were held rigid during every simulation using the SHAKE algorithm[78]. To ensure that a time step of 1 fs is allowed for the rest of the system, all present harmonic bond periods were calculated. By a rule of thumb, it was decided that if the found periods of oscillation divided by 25 was shorter than 1 fs, the bond should be held rigid with the SHAKE algorithm as well. The only bond that did not satisfy this condition, was the hydrogen bond in the bicarbonate molecules.

All MD calculations were carried out in the NVT ensemble using the velocity-Verlet integrator, using a

time step of 1 fs. Solely the electrolyte in each system was coupled to a Nosé-Hoover thermostat with a damping factor of 100 fs. Because the electrode atoms are fixed in space, the system in its entirety would be assigned an undesired temperature if the entire system was coupled to the thermostat. The PPPM solver[79] was used as the long-range solver in all simulations, with a set accuracy of 10^{-4}. To allow for a long-range solver to be used, the slab option was invoked with a volume factor of 3.0.

To obtain MD-only results, each initialised system was subsequently simulated for 30 ns. The way the initialised systems were used in the coupled QM-MD runs, will be described in section 3.3.

3.1.4. Data analysis
To further our understanding of the simulation and check its validity, multiple properties of the MD simulations are calculated from the MD output. This includes density profiles and molecule orientation, but also MSD measurements and pair correlation function plots. Below, a short explanation of the latter two is provided.

MSD
The mean squared displacement (MSD) is a measure of the displacement of the positions of a particle with respect to some reference position over time. The MSD is defined as[80]

$$\text{MSD} = \frac{1}{N} \sum_{i=1}^{N} \left| r^{(i)}(t) - r^{(i)}(0) \right|^2, \tag{3.1}$$

where N represents the total number of atoms in system. Using the MSD, a diffusion coefficient can be calculated. In terms of the diffusion coefficient D, the MSD is given as

$$MSD = 2n_{dim}Dt \tag{3.2}$$

where n_{dim} is the number of dimension taken into account in the MSD measurement. Since the diffusion coefficient is a force field dependent property, it can be used to provide insight into whether the system is behaving as it should. Furthermore, as will be explained in section 4.3, it can be used to get an idea about the average expected displacement of ions in the system.

Pair correlation function
A pair correlation function describes how a time-averaged density of a set of particles a varies as a function of the distance from a reference set of particles b. In essence, a pair correlation function counts the average number of particles in set a in a shell at distance r from a particle in set b, and represents this as a density. It can be given as

$$g_{ab}(r) = (N_a N_b)^{-1} \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} \langle \delta(|\mathbf{r}_i - \mathbf{r}_j| - r) \rangle \tag{3.3}$$

where N_a and N_b represent the total number of atoms in sets a and b. The pair correlation function provides information about the configuration of the system, which can be useful for mapping the differences between a system in different states.

3.2. DFT simulation
3.2.1. SCC-DFTB benchmarking
All SCC-DFTB calculations in this work were performed using the free software package DFTB+[81]. As previously described, the DFTB method requires an appropriate set of parameters to perform calculations. These parameters are stored in so-called Slater-Koster (SK) files. These contain diatomic

parameters, describing the interaction between two types of elements (Cu-Cu, Cu-H etc.). For copper, the matsci-0-3 set[82][83][84]) of SK-files is used. Various publications mentioned different sets of SK-files for copper. Unfortunately, these either do not share the parameterisations they made themselves[85][86], or refer to sets that do not include copper[87]. Parametrising a new set of SK-files was outside of the scope of this project.

The parametrisations contained in these files are often fitted under specific circumstances. For example, the SK-files for copper were made in the context of copper in zeolite, a type of mineral. There is no guarantee that these files will perform appropriately outside of their respective fitting contexts. The interest in this work solely goes out to polarisation predictions, meaning that only static single point calculations will be carried out. Therefore, the fitting context of the repulsive energy parameters is not relevant. Still, there is no guarantee that the available copper SK-files can appropriately reproduce the total polarisability of the system as it would be given by DFT calculations, the level of theory that SK-files are fitted to. This is simply because the SK-files probably were not fitted for this specific purpose.

Therefore, the performance of SCC-DFTB calculations using the matsci-0-3 SK-files set was benchmarked against to calculations on the DFT level. To do so, the method of Elliott et al.[14] is partially followed. A neutral (approximately) rectangular electrode is simulated at both the DFT and the SCC-DFTB levels of theory, with a point charge placed a few Ångström directly above the center of the electrode. The induced charge distributions are subsequently compared and the relative performance of the SCC-DFTB calculations is evaluated.

Because parametrising a new set of SK-files was outside of the scope of this project, a perfect overlap between the DFT and SCC-DFTB calculations was not expected. This was not seen as a requirement for obtaining insight into the effect of polarisation. The benchmarking served mainly as a way to show that the used set of SK-files provided a reasonable approximation of the DFT level of theory, and as a way to be able to disallow the set of parameters if their performance was deemed unsuitable to continue with.

The SK-files for graphene were provided by the mio-1-1 set[88]. For this set, a comparison of SCC-DFTB calculations and DFT calculations was already performed in the work of Elliott et al.[14]. Because the performance was deemed acceptable, no further investigations will be performed in this work.

3.2.2. Simulation details
The electrode systems were generated and exported using LAMMPS. Using Amsterdam Modelling Suite-Amsterdam Density Functional (AMS-ADF)[89][90], (static) single point calculations were performed. Since in AMS-ADF point charges could only be modelled in non-periodic systems, both DFTB+ and ADF calculations were performed using open boundary conditions. The added multipoles in ADF were represented as real point charges, not by a spherical Gaussian distribution. All electronic wave functions for copper were expanded in the STO-TZ2P basis set[91], and the GGA-PBE functional[92] was used to describe the electron XC-interaction. Both were previously used in copper DFT calculations.[86][85][93][94]

Scalar relativistic corrections were included using the default ZORA formalism[95], and the core was treated using frozen orbitals. All calculations were performed spin-unrestricted, meaning that the simulation has to take into account the spin state of each individual electron in the energy minimisation. For systems with an uneven number of copper atoms, an initial spin polarisation of 1.0 was set (one more up-electron than the number of down-electrons), while for all systems with an even number of copper atoms a spin polarisation of 0.0 was set.

As in SCC-DFTB simulations, AMS-ADF requires the system to be solved self-consistently, i.e., the computed density must define the same potential as from which the density itself was computed. Until convergence is reached, every cycle the electron density is recomputed. Convergence issues are frequently encountered with d-elements like copper.[96] In order to speed up and/or stabilise the SCF (self-consistent field, as it is called in AMS) procedure, and to avoid non-convergent oscillatory be-

haviour, various acceleration options were implemented.

In all calculations, the default mixed ADIIS+SDIIS acceleration method by (Hu and Wang, 2010)[97] was employed. Furthermore, the number of DIIS vectors used was increased from 10 to 20 or 25, depending on whether convergence could be reached. The number of 'DIIS vectors' denotes the number of expansion vectors that was used in the SCF acceleration procedure. They are part of the damping procedure, where using more vectors means taking into account more of the previous guesses for the electronic density in combination with the current one, when constructing the next estimate.

Except for the boundary conditions, the simulation details of DFTB+ were the same as used in the rest of the thesis, and can be found in section 3.3.2.

3.3. SCC-DFTB simulation

Density functional tight binding simulations were used in this project to enable computations on larger systems and larger timescales than are feasible in DFT, and still get reasonably good results. Whereas DFT in principle is an exact method (it only becomes non-exact when an approximation for the exchange-correlation functional is used), DFTB contains parameters to approximate the solution to the Schrödinger equation.[42] In terms of computational efficiency, DFTB lies between DFT and classical MD.[98] In general, DFTB codes are roughly three orders of magnitude faster than DFT, and three orders of magnitude slower than classical MD codes.[54] To portray the difference in computational efficiency: in DFT it is common practice to use systems of up to 100 atoms on a picosecond timescale. While classical molecular dynamics codes are not able to simulate systems with a macroscopic number of atoms ($\sim 10^{23}$), in general they are able to simulate systems of thousands of atoms on a timescale up to microseconds.[99]

3.3.1. QM-MD loop

In this thesis, SCC-DFTB calculations are coupled with an MD simulation. Using the SCC-DFTB method, the partial charges on the respective electrodes in the MD systems are computed. In this section the coupling is explained in detail. All SCC-DFTB calculation in this thesis were performed using the free software package DFTB+[81].

The MD systems, after being initiated as described in section 3.1.3, constituted the starting point of the QM-MD scheme. Only one of the two electrodes in each of the MD systems was quantum-mechanically modelled, the partial charges on the other electrode remained constant. The scheme used next consisted of four steps:

(Step I)
The Cartesian coordinates of the atoms in the electrode of interest were imported from the MD simulation. These served as the main input of the DFTB+ simulation, and the corresponding atoms were simulated properly. All atoms from the MD system not constituting the electrode of interest, served as a background potential. To this purpose, their respective Cartesian coordinates and partial charges were imported in DFTB+, where they were modelled as a set of point charges.
(QM step)
A static SCC-DFTB calculation was performed. The electronic structure of the electrode surface was computed, using the set of point charges as background potential. A Mulliken population analysis was performed to find the partial charges on each of the electrode atoms.
(Step II)
The computed partial charges were exported and used to update the partial charges in the MD simulation.
(MD step)
A short MD simulation was performed.

After step (4) was finished, the entire scheme was iterated. Processing steps I and II are performed

using the Python programming language[100]. The corresponding schematic representation of the QM-MD scheme can be found in figure 3.2.
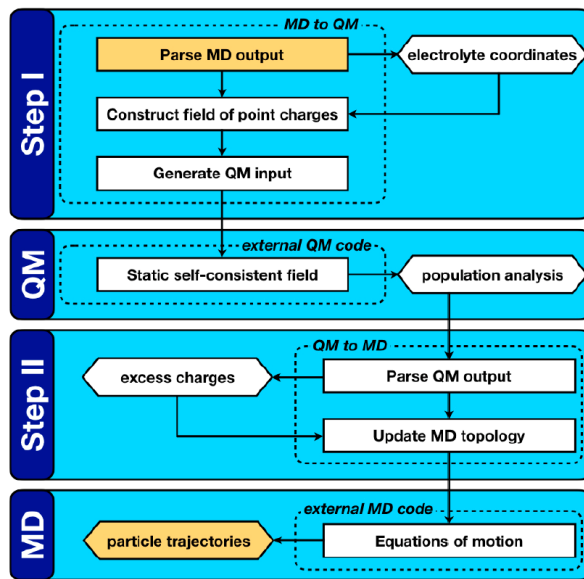

Figure 3.2: Schematic representation of the QM-MD workflow followed in this thesis, taken from the work of (Elliott et al, 2020)[14]. Key computable quantities are represented by hexagonal boxes, and square boxes represent computational processes. The two boxes colored gold link sequential iterations.

The partial charges of the electrode of interest were not computed at every time step of the MD simulation, as DFTB+ calculations were three to four orders of magnitude slower than the MD simulations. Therefore, they would have taken up the large majority of the total loop simulation time. Even though physically it would have made the most sense if the partial charges were computed every time step, the simulation of an increased number of time steps was permitted due to the time between QM calculations being increased.

When choosing the number of MD time steps that should be performed before every DFTB+ calculation, there is a trade-off between accuracy of the simulation and the practical viability of the loop. First of all, the choice should make physical sense. The QM computations must be used often enough that the representation of the polarisation is accurate and it is possible for interesting physically meaningful results to be obtained. At the same time, the total simulation time must be kept down. The total number of time steps that can be simulated must be large enough to allow some form of equilibrium to be reached, and then sufficient results must be obtainable from the remaining simulation time.

It can be expected that the electrolyte configuration does not change much in a few femtoseconds. This would mean that the partial charges also would not change much, such that a new DFTB+ calculation would not necessarily be required. However, on a nanosecond scale, the electrolyte configuration is likely to evolve to the point where the electronic configuration has changed beyond recognition. To be able to get a well suited estimate for the time between QM computations, first, multiple different test loops are performed, each with a different loop time. It is then measured how much the partial charges change, such that a suitable loop time can be selected. If this method does not yield conclusive results, the choice will be made based on MSD measurements of the ions in the electrolyte. These can be used to get an estimate of how fast the ions on average move near the electrode, and thus how fast the electrode charges are expected to undergo physically meaningful change.

3.3.2. SCC-DFTB simulation protocol

In this section the computational details of the performed DFTB+ calculations are provided.

As previously described, the SK-files required by DFTB+ to simulate copper are provided by the matsci-0-3 set[82][83][84].

Point charges in DFTB+ were modelled without any Gaussian delocalisation. d-orbitals were the orbitals with the highest angular momentum modelled for the copper simulations. All orbitals up to that angular momentum were included in the calculations. In all calculations, the default SCC-tolerance of $1 \cdot 10^{-5}$ e was used (maximum acceptable self-consistency difference in any charge between two SCC cycles). In each DFTB step, the modelled charges were initialised from the optimised charges from the previous DFTB step.

The system was modelled as a periodic supercell cluster. In the z-direction a large vacuum as added of 20 nm, to minimise interaction between periodic images in the non-periodic direction in the MD simulations.

In a periodic system like the one used here, integrals over real space (which is infinitely extended) can be replaced by integrals over the (finite) first Brillouin zone in reciprocal space. This is possible because the system deals with electrons in a periodic lattice, of which Bloch showed that the wave function could be written as a walking wave modulated with a periodic function related to the lattice.[101] By means of this periodicity, solutions can be characterised completely by their behaviour in the first Brillouin zone, or reversibly, the first Brillouin zone alone can provide information about the entire structure. Real space integrals over large systems, this way, become integrals over limited reciprocal space.

The integrals in reciprocal space are calculated by summing the function values of the integrand, like the charge density, at a finite number of points in the Brillouin zone, called k-points. There are many ways to choose a set of k-points. To obtain well converged results, a sufficiently dense mesh of well-placed k-points is required. To this purpose, convergence tests were performed for all systems simulated using SCC-DFTB. In all simulations a Monkhorst-Pack scheme[102] was chosen to generate k-points in the Brillouin zone. The Monkhorst-Pack scheme is a widely used scheme that distributes k-points homogeneously over the Brillouin zone, in a manner parallel to the reciprocal lattice vectors that span the Brillouin zone.

An even 2x2x2 Monkhorst-Pack scheme was ultimately chosen for all simulations. This resulted from a calculation of the total energy of a test system as a function of multiple k-meshes, for which either odd or even values were chosen. The k-mesh that was ultimately chosen was the one for which the difference in computed total energy between subsequent k-values (even compared with even, odd with odd) was below 1 meV per atom. Additionally, the polarisation results were compared between the final choice and results obtained using a much more dense scheme, to verify the negligible effect on the computed charges. The results of these tests can be found in appendix A.

3.4. NN implementation

In the QM-MD loop, the SCC-DFTB calculations take up the far majority of the total simulation time. Therefore, the QM part forms the bottleneck for the total simulation time. Many problems, however, require a time scale that is much longer than currently achievable using this approach. A recent paper of Di Pasquale et al.[34] have shown that, at least up to a certain level, an fully connected network is able to replicate/predict the results of non-periodic SCC-DFTB calculations on a single graphene layer. The computation time of such a network, once it is trained, is negligible compared to the computation time of SCC-DFTB calculations. Consequently, replacing the SCC-DFTB calculations by a trained neural network in the QM-MD loop, as given in figure 3.2, could possibly allow for a much longer simulation time scale to be achieved, while using an accurate description of the electric polarisation. In this final part of this thesis, it was explored whether the approach by Di Pasquale et al.[34] could be extended to the prediction of partial charges on a periodic copper electrode using a NN, by training it on the set of SCC-DFTB results obtained from the QM-MD loop.

Different kinds of neural networks were used in this thesis. Firstly, an FCN was implemented and trained. For this, the approach of Di Pasquale et al.[34] was mostly followed. The implementation of this type of network in this thesis is described in subsection 3.4.1. In subsection 3.4.2, the implementation of a second and novel approach to using neural networks in the context of electrolyte-electrolyte interfaces is described. In this approach, convolutional neural networks play a central role. All neural networks in this thesis were created using the PyTorch python library[103] (v.1.9.1).

3.4.1. Fully connected network

The architecture of the FCN applied consisted of four fully connected layers, each containing 1024 neurons, and ReLU activation functions. For each prediction, the output of the network consisted of a one-dimensional array containing the instantaneous partial charges on each atom on the electrode, such that in this multi-output approach all charges were simultaneously predicted. A schematic of the base architecture of the used FCN can be found in figure 3.3.
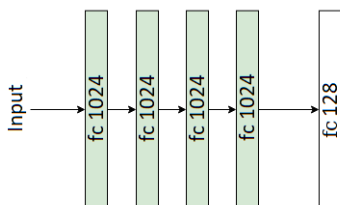
Figure 3.3: Schematic of the base architecture of the fully connected network used in this thesis, consisting of four fully connected layers, each containing 1024 neurons and ReLU activation functions, and a fully connected output layer. A fully connected layer with 1024 outputs is denoted as "fc 1024"

The input of the network, also represented in the form of a one-dimensional array, consisted of the Cartesian coordinates of all ions in the system, both the potassium and the chloride ions. For the particular case of graphene, Di Pasquale et al.[34] have shown that the ions' configuration formed a good descriptor of the system for training the NN. That the positions of the water molecules could seemingly be neglected, was attributed to the supposed correlation between the positions of the ions and the water molecules, such that the network seemed to be able to deduce the possible influence of water molecules from the positions of the ions.

In order to find the most optimal version of the FCN, many variants of the network described above were attempted. For example, the number of layers and the number of neurons were varied. Additionally, a variant was created in which the 50 water molecules closest to the electrode, which were assumed to be the most important ones in determining the polarisation, were included in the input, to find out whether the water molecules indeed did not influence the results significantly.

Furthermore, to be more in line with the implementation of Di Pasquale[34], additional variants of the network were created. In one variant, also the partial charges of the respective ions were used in the input. In another variant, the distribution of charges was divided into bins of a finite size. In this variant, the NN did predicted the label of the bin to which the partial charge on a particular atoms belongs, and not the actual value of the charge. The network was trained by keeping the the order of the coordinates in the input array sorted by atom type, and the order in the target output values sorted by position. This was important, as the network needs to be able to differentiate between the two different atom types and relate the input positions to the output.

3.4.2. Convolutional neural network

The FCN described above has some disadvantages. Firstly, the FCN is not invariant to permutation of identical particles. This is not a problem for the output, as the atoms in the electrode are fixed in space and are therefore defined solely by the order in the target output, such that no permutation of the copper atoms needs to be considered. However, this does present a problem for the input. Switching two potassium ions in the input, yields input data that is seen as unique by the FCN. Secondly, the FCN lacks transferability with respect to the molarity of the solution. The FCN, as described above, has a fixed number of inputs, and therefore cannot be used in situations where a larger or smaller number of ions are present in the electrolyte. Thirdly, the FCN treats atoms close together and atoms far apart from each other, on the same footing. In the context of the presented polarisation problem, this is most probably unnecessary, and the network could be simplified if it was able to take into account the context of every ion.

To deal with these three disadvantages, here it is proposed to transform the given problem into a computer vision one. To allow for this, the main idea is to divide the input space, the space occupied by the entire MD system, up into bins. When an atom happens to be present in a given bin, that bin gets assigned the value of the charge of that atom. This way, the input has become permutation invariant, as switching two ions of the same kind still yields the same input configuration. This approach also is a first step towards solving the issue with transferability with respect to the molarity of the solution, as the input space no longer changes when the concentration changes.

Moreover, by representing the input space as a three-dimensional matrix of values, state of the art computer vision techniques could be used in solving the polarisation of the electrode. Convolutional neural networks could be applied, which have been applied many times before to 3D vision problems, successfully. Finally, this could help solve the third issue mentioned, as convolutional neural networks inherently treat locations close-by and far apart on a different footing. Regrettably, in the context of the modelling the electrode polarisation, the proposed vision approach comes with its own disadvantages, and these all have to do with the handling of spatial information.

By binning the input space, spatial information is lost, due to computational limits to processing large 3D objects. Since the 3D matrix representing the system will be increasingly sparse for smaller bin sizes, one solution to this problem would be representing the system using a sparse matrix. Subsequently, a convolutional network made for processing sparse data could be used, for which multiple approaches have been described[104][105][106]. Unfortunately, due to severe time limitations it was not possible to implement these for this thesis.

In order to still be able to allow for an accurate representation of the input space, it was decided to scale the bin size, depending on the respective location in the input space. Effectively what was done, was choosing a finer z-coordinate binning, the closer the z-coordinate was to the electrode. This way, differences in the z-position of atoms close to the electrode were represented more accurately than those far away from the electrode. This was done, as it was seen that atoms close to the electrode have a relatively larger influence on the polarisation than atoms far away, such that differences in spatial coordinates have a relatively larger impact for atoms that are close by. Results that verify this, can be found in section 4.2.

In some ways, the translational equivariance of the convolutional layer might be beneficial to learning the features of the problem, especially in the earlier layers. For example, it allows the local regions surrounding the electrode atoms to be treated in the same way for every electrode atom, which is considered to be natural. However, the translational equivariance of convolutional layers might pose a problem if it denies the network from finding appropriate representations of individual regions in space, when these are required. In order to help the network to use the available spatial information as much as possible, padded and relatively deep CNNs were used. Recent papers have shown that CNNs are able to learn to use spatial information.[107][108][109] To allow a CNN to do so, (same) padding must be applied. This allows the network to place weights outside of the input space, that can allow information at specific location to be either taken into account or ignored.[107] Moreover, the network needs to be deep enough that the described padding effect can be used throughout the entire input space.

For this reason, the standard architecture of the used CNN in this thesis, was chosen to resemble the ResNet-18 network.[61] Kayhan et al.[107] showed that this network, while relatively shallow compared to other CNNs, can classify locations 128 pixels away from the boundary, which is sufficient for the purposes of this thesis. ResNet networks consist of multiple of the skip-connection blocks that were seen in figure 2.4 in tandem, and the number 18 refers to the combined number of layers in the network. Most of the dimensional reduction in a ResNet architecture is provided by strided-convolutions, which are thought to better suit the presented problem by not relying on pooling layers, that are thought to destroy spatial information. Furthermore, The standard ResNet-18 architecture was changed for the purposes of this thesis, by replacing the pooling layer that is normally present in the first layer of the network, by a strided-convolutional layer. The base architecture of the CNN used in this thesis can be found in figure 3.4.
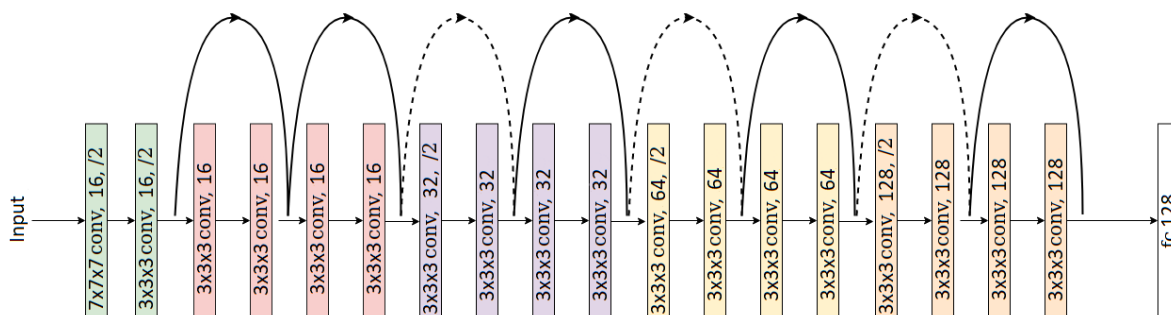
Figure 3.4: Schematic of the base architecture of the 3D ResNet-18-like convolutional neural network used in this thesis. Black arrows indicate connections between layers, with dashed connections indicating a strided convolutions that are used to allow for a skip connection between outputs of different sizes. Each depicted convolutional layer was followed by a batchnorm layer and a ReLU activation function (the latter after addition of the output from the skip connection), except for the first layer. A convolutional layer using a 3x3x3 kernel, a stride of 2 and an output containing 16 channels, is denoted as "3x3x3 conv, 16, /2", and a fully connected layer with 128 outputs as "fc 128". For the 2D version of this network, the 3D convolutions were replaced by 2D ones.

Since the CNN could possibly be benefited by access to additional spatial information, so that it can choose whether to keep the spatial equivariance, or discard it, the 'coordinate convolution' approach was applied in some variants of the described CNN. In a coordinate convolution layer, the spatial bin numbers of the input space in each dimension are appended to the input in the form of extra channels, i.e., for example, an x-channel is appended that contains a matrix where each element is assigned a value based on its x-position relative to the other elements, instead of containing a sparse matrix of charges.[110] This way, the network can directly choose to either incorporate the provided spatial information, or discard it (so choose whether to keep spatial equivariance, or discard it).

Both 2D and 3D versions of this network were created. In the 2D network every z-bin represented a channel in the input data, whereas in the 3D network the input space could be represented using one single channel. Like for the FCN, multiple variants of the described network and the input data were attempted, and will be discussed further in section 4.5. For example, training data was created in which the water molecules were taken into account. As described, in this approach this did not lead to an increase in the input dimensions.

3.4.3. Training procedure

To obtain training data, the configuration of the system in the QM-MD system was saved after every time an SCC-DFTB calculation was performed. To effectively increase the amount of available training data, the training data for the FCN could be permutated, as discussed before. Because the training time required scales with the number of training sets, this could not be done more than a few times, given time and computational constraints. Moreover, for both networks, certain symmetries in the system could be exploited. Depending on the symmetries present in the modelled electrode, the system could be mirrored and rotated to obtain new training data that is seen an unique by the networks. The only constraint to operations like these, is that the absolute positions of the copper atoms in the electrode remained the same.

To ease the training process of the FCN the copper electrode was fixed in space, like the graphene electrode was in the work of Pasquale et al.[34] was fixed in space. This allowed the networks to have to account for one less degree of freedom in the input.

During the training stage, the Adam optimiser was used with an initial learning rate of $1 \cdot 10^{-3}$. For training the FCN, this learning rate was multiplied by 10^{-1} after every 25 epochs. For training the CNN, the number of feasible epochs was lower, due to the inability to use sparse convolutions, and the learning rate was multiplied by 10^{-1} after every 10 epochs. The loss itself was calculated as

$$\mathcal{L} = \sum_{i=1}^{N_a} |y_i^{predicted} - \hat{y}_i^{target}| + \lambda |\sum_{i=1}^{N_a} y_i^{predicted} - \sum_{i=1}^{N_a} \hat{y}_i^{target}|, \qquad (3.4)$$

where N_a is the total number of atoms in the electrode. As can be seen in eq. (3.4), the loss function was made up of two separate contributions. The first term is the mean absolute error (MAE) loss between the predicted charges and the target charges, multiplied by N_a. The second term is the total difference between the sum of the predicted charge by the NN, and the sum of charge that was applied to the electrode. This forms a soft constraint on the prediction of the neural network. λ is an adjustable parameter that sets the strength of the penalisation, that was set to 1.

Setting this type of constraint is a way of regularisation, as it prevents the NN from learning models which do not predict a total charge charge equal to the one applied in the QM-MD loop. Additionally, weight decay was applied in all trained models. Lastly, the given loss function was varied, by individually scaling the before mentioned terms. Additionally, the use of an RMSE loss was attempted.

4

Results and discussion

In this chapter all results will be presented and discussed. In section 4.1, the process of selecting an appropriate MD system is shortly outlined. Furthermore, a general discussion of the used MD system is provided. Subsequently, in section 4.2, the benchmarking of the SCC-DFTB calculations against DFT are presented and discussed. Section 4.3 describes how the time coupling interval between the MD and SCC-DFTB simulations in the QM-MD loop was chosen. Section 4.4 contains the results of the QM-MD run, and the discussion of those results. This section is followed by individual discussions of the MD simulation and the SCC-DFTB simulations. Finally, in section 4.5 the results of the attempt to replicate the results of the SCC-DFTB simulations using neural networks is described.

4.1. Choosing an MD system

To create the QM-MD loop, first the system to simulate had to be chosen. Subsequently, it had to be confirmed that the chosen MD system behaved appropriately, i.e., the MD system did not display behaviour that would limit its usefulness in the evaluation of the QM-MD loop. Unfortunately, as already described in the method, the first few systems that were attempted, displayed exactly this kind of behaviour. Here, some results from these first runs are presented.

As described in the method, system 1 was created to mimic a carbon dioxide reduction system. An electrolyte was simulated that was bound by two copper electrode sheets and contained carbon dioxide molecules, and bicarbonate and potassium ions. Initially, the force field provided by Demichelis et al.[72] was used to model the bicarbonate molecules. As described in the method, mixing of the bicarbonate with other atom types was modelled using copies of interaction parameters between similar types of atoms in the system. This was considered to be an acceptable approach because bicarbonate was not thought to be crucial in determining the behaviour of both the carbon dioxide molecules and the positive potassium ions at the cathode. This approach, however, yielded clustering between the bicarbonate and potassium ions, as shown in figure 4.1.

Clustering may have resulted from the following. The electronic structure of molecules depends on their environment. Force fields of ions are often optimised in infinite dilution, so when a higher molarity is used, or other ions are added to the solution, the presumed electronic forces might no longer be correct. An attempt was made to reduce the amount of clustering by lowering the molarity and scaling down the ionic charges by means of the Electronic Continuum Correction (ECC) (without subsequently altering the LJ interacting parameters as is usually done).

In addition, the temperature of the system was increased from 298 K to 323 K. In some experiments the implemented potassium force field was also changed from that given by Yagasaki et al.[73] to that given by Joung and Cheatham[74]. Additionally, the potassium ions were replaced by sodium ions (using again the force field by Joung and Cheatham[74]), as for sodium the bicarbonate force field provided interaction parameters with the oxygen in the bicarbonate molecule (not the carbon or hydrogen).
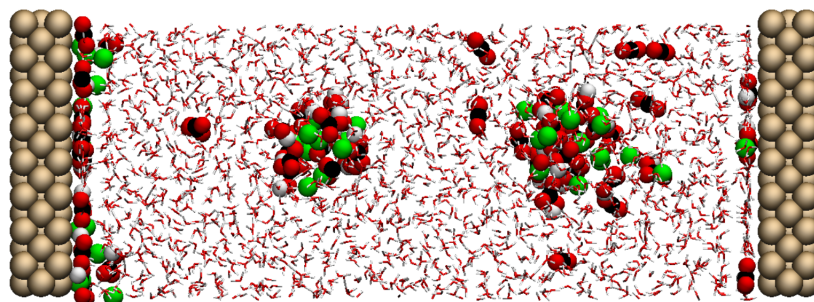
Figure 4.1: Visualisation of system 1, simulated using the bicarbonate force field provided by Demichelis et al.[72] The potassium ions are coloured green. Oxygen atoms are red, hydrogen white, and carbon is coloured black. Significant clustering is noticed between the oppositely charged ions.

All attempts failed to eliminate clustering.

The observed clustering is most likely a result of the implemented force fields of bicarbonate and potassium. These were mainly copied from the interaction of carbon dioxide with potassium, which were not adapted to bicarbonate. Since none of the attempts ultimately solved the clustering issues, the bicarbonate force field was changed. New force field parameters were generated using the online LigParGen parameter generator[75], as described in the method. A visualisation of a resulting run can be found in figure 4.2, and plots of the corresponding cumulative distribution functions can be found in figure 4.3.
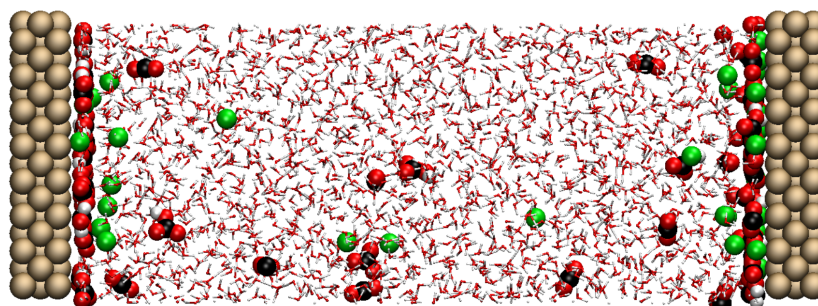
Figure 4.2: Visualisation of system 1, simulated using the bicarbonate force field generated using the online LigParGen parameter generator[75]. The potassium ions are coloured green. Oxygen atoms are red, hydrogen white, and carbon is coloured black. The majority of the ions can be observed to be in either one of the double layers on the respective electrode surfaces.

As shown, the bicarbonate in the system was primarily adsorbed to the copper electrode surfaces. This is thought to be mainly the result of the relatively strong pair interaction of copper with other atom types, which, for example, is significantly larger than that between water and other atoms types. In addition, a bicarbonate molecule consists of five atoms, so that each atom experiences an attractive force towards one of the electrodes.

Even though one the electrodes has a negative net charge (the cathode is on the left in the given figures), bicarbonate is still adsorbed on the surface. In the example given, even more bicarbonate is adsorbed on the negative electrode than on the positive electrode; this can be an effect of the attractive interaction between the two types of ions. Electrostatic effects seemed to play a minor role compared to the LJ interaction. Therefore, it was thought that if surface polarisation effects were to be included in the simulation of this system, it would only be able to play a relatively minor role. This system was considered to be non-ideal for a first study into the effect of surface polarisation. Additionally, the observed long adsorption times and resulting reduced dynamics of the system would make this system relatively unsuitable to train a neural network on, as this often requires a diverse training data set to properly generalise.

(a) Bicarbonate
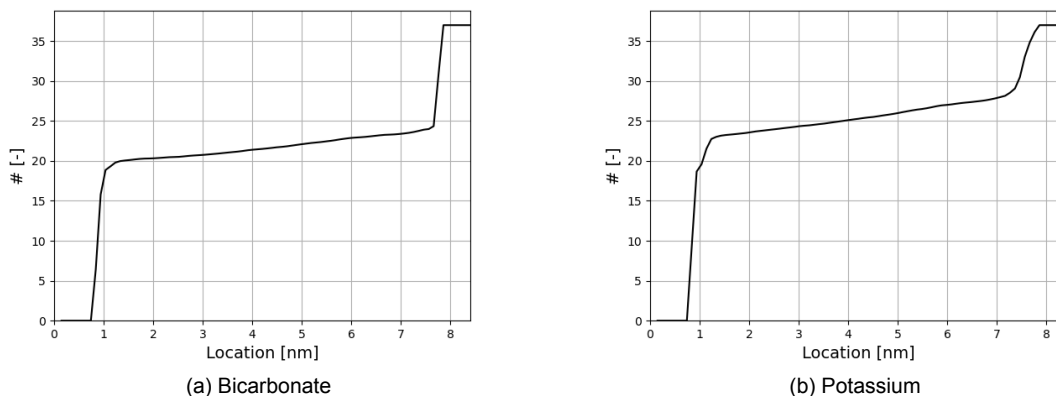
(b) Potassium

Figure 4.3: Plots of the cumulative distribution functions of the ions in system 1, bicarbonate in figure (a), and potassium in figure (b). In total, there were 37 ions of both types in the system. The left electrode was negatively charged, the right electrode positively. The majority of the bicarbonate atoms is observed to be adsorbed on either of the electrode surfaces. The majority of the potassium ions is part of one of the electric double layers existing on the electrodes.

Increasing the temperature of the system, again, to 323 K did not seem to have a significant effect on the bicarbonate adsorption. Moreover, it was attempted to artificially scale down the LJ interaction of the bicarbonate molecules with copper. Ultimately, however, it was decided to exchange the bicarbonate ions for chloride ions. This was done because a down scaling of the LJ interaction of the bicarbonate molecules would create an nonphysical situation, such that there would no longer be a point in choosing bicarbonate to be the anions in the system, over any other choice of anion. In addition, to further simplify the system, it was decided to remove the carbon dioxide molecules as well. Now system 2 was obtained, which serves as the main system under study in the rest of this thesis.

Using system 2, a classical MD run of 30 ns was performed. The results served as reference to compare the results of the QM-MD loop to, and will be presented together with the QM-MD results in section 4.4.

4.2. Benchmarking the SCC-DFTB calculations against DFT

The performance of the SCC-DFTB simulations, using the matsci-0-3 SK-files set, was benchmarked (primarily qualitatively) against calculations on the DFT level of theory, using the AMS-ADF software. Different non-periodic SCC-DFTB and DFT simulations were performed in which the charge distribution was computed on multiple sheets of (100) copper of different sizes, under the influence of one or multiple point charges at different locations. Additionally, to compare the effect of polarisation in both simulation techniques, the copper sheets were simulated in a vacuum. This allows for the polarisation effect to be quantified and compared. This, in turn, was done by subtracting the values of the partial charges q_0^i, as they were in the vacuum, from the partial charges q^i, as they are affected by a point charge. Consequently, a set of excess charges $\Delta q^i = q^i - q_0^i$ was obtained, which represents the effect of the polarisation.

The charge redistribution is visualised by plotting the sum of all excess charges as a function of the radial distance r from the center of the sheet ($\Delta q(r) = \sum_i \Delta q^i$ for all charges for which $r_{q^i} \leq r$). This provides more information than the average error, as this would simply be a value that decreases with the size of the sheet. Furthermore, in the plots that will be shown, the maximum error made by the SCC-DFTB simulation is always made in the prediction of the excess charge on the atom in the center of the sheet (and if there is no atom in the center of the sheet, on the the very first atoms from the center), and can therefore be read from the given plots.

4.2.1. Three layered sheet

The first idea was to use multi-layered (100) copper system, to mimic a real copper electrode. Therefore, the first simulated system consisted of three layers of (100) copper, with a total 216 atoms and a point charge (q = -1.0 e) at 3 Å from the center of the sheet. This distance was chosen as it was observed to be close to the adsorption distance of certain ions in the MD simulation. A depiction of the system is shown in figure 4.4, and the plot with the SCC-DFTB and DFT results can be found in figure 4.5.

(a) Top view.   (b) Side view.
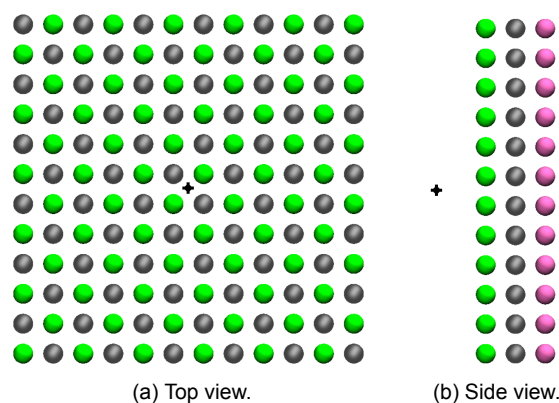
Figure 4.4: Top- and side view of the three-layered (100) copper sheet, consisting of 216 atoms in total. The atoms shown in green form the top layer. The gray atoms make up the second layer and the magenta atoms constitute the final layer. A point charge is simulated at the location of the black cross in the middle of the sheet, exactly three Å above the top layer. From the black cross outward, the computed excess charge localised on the atoms (at both the DFT and SCC-DFTB levels) is summed and plotted in figure 4.5.
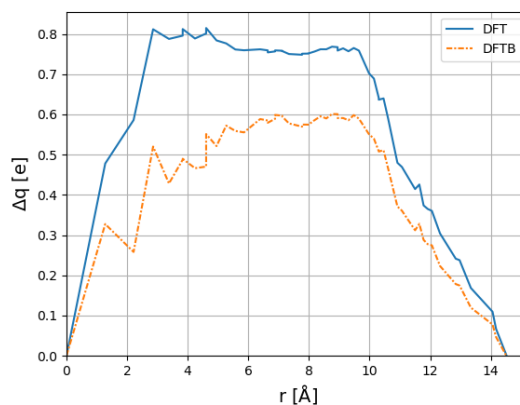
Figure 4.5: Plot depicting the DFT (solid) and SCC-DFTB (dash-dotted) summed excess charge Δq as a function of the radial distance r from the center of a three layered sheet of (100) copper, polarised by a point charge (q = -1.0 e) at a height of 3 Å above the sheet. The sheet is depicted in figure 4.4. Significant differences between the SCC-DFTB and DFT results can be observed. For example, the value of summed excess charge can differ more than 50%, and the radial distance at which it occurs is different.

In both the SCC-DFTB and the DFT simulation, the negative point charge can be seen to accumulate positive charge near the center of the sheet. From 3-4 Å the presence of the point charge is felt less, and the total amount of excess charge remains approximately the same. From 9.5 Å onwards the amount of excess charge decreases in order to compensate for the charge accumulation at the center.

Except for the accumulation of positive charge near the center and the depletion of the charge at the boundaries of the sheet, significant differences between the SCC-DFTB and DFT results can be observed. The maximum value of summed excess charge differs more than 25%, and the radial distance

at which it occurs is different. At a radial distance just over 2 Å, SCC-DFTB predicts a decrease in the total amount of excess charge, rather than an increase as predicted by DFT. The summed excess charge differs more than 50%.

The difference can be explained by looking at the predicted partial charges on the sheet in a vacuum. In a vacuum, DFT predicts a net positive charge on the front and back sheets, and a net negative charge on the middle sheet. SCC-DFTB, however, predicts the opposite: a net positive charge on the middle sheet, and a net negative charge on the front and back sheets. As a result, when under the influence of a negative point charge, SCC-DFTB predicts a larger negative polarisation of the middle sheet than DFT, leading to a decrease in predicted excess charge just over 2 Å in figure 4.5. Although effect of polarisation in both SCC-DFTB and DFT seems to behave similarly, the predicted partial charges differ significantly. This is a result of the difference in the predicted charged in a vacuum. Because the non-polarised charges do not correspond well between the different methods, the charge predictions under polarisation also do not match.

Figure 4.6 depicts the predicted partial charges by DFT and SCC-DFTB, in the case of polarisation by the negative point charge. These side views of the sheet, show that DFT and SCC-DFTB predict opposite signs for many of the partial charges. It should be noted that the colour ranges used for these figures, and all subsequent similar figures, do not necessarily correspond to the maximum and minimum colour values displayed in the figures. The purpose of these figures is solely to provide insight into how polarisation affects the charge distribution across the electrodes. What should be noted in these plots is the sign of the charges assigned to specific atoms, and there relative colour compared to the other atoms, with darker values corresponding to higher values.

In appendix B.1 more simulation results of three-layered sheets can be found. There it is shown that optimising the used lattice constants for either one of the the simulation methods, does not significantly change/improve the results. Furthermore, it is shown that taking into account colinear (up or down) spin polarisation using the spin constants provided by Yin et al.[86], also does not yield a significant improvement of the results. This was anticipated, as using these parameters was not recommended, since they were fitted using a different XC-functional than the SK-files. Moreover, for sheets with an even and odd total number of atoms, it was verified that in the DFT simulations the initial spin configuration (-2, -1, 0, 1 and 2), had no effect on the final outcome.

Based on these results, it was decided to attempt to use a copper sheet consisting of a single layer in this thesis.

(a) DFT   (b) SCC-DFTB
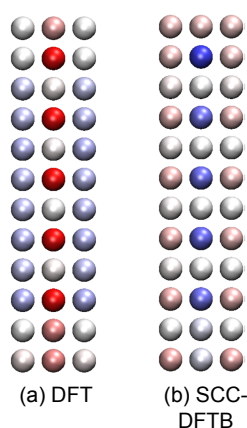
Figure 4.6: Side view of the polarised three layered (100) copper sheet. The atoms are coloured depending on the predicted partial charges, computed using DFT and SCC-DFTB. The same charge scale was used for both plots, from -0.15 e (red) to 0.15 e (blue). Charges outside of this range were capped, and were assigned the maximum colour values. In can be observed that DFT predicts charges opposite in sign compared to SCC-DFTB.

4.2.2. Single-layered sheet

Next, multiple single-layered sheets were simulated using both DFT and SCC-DFTB, for a similar negative point charge at a location 3 Å above the center of the respective sheets. For a (100) copper sheet consisting of 128 atoms, the plot of the summed excess charges, as a function of the radial distance from the center atom (center in this case chosen to be located on an atom), are shown in figure 4.7. Figures 4.8 and 4.9 show the sheet atoms, coloured using the values of the partial charges, and the excess charges, respectively. These results show that for a single layer, SCC-DFTB is able, to some extent, to capture the polarisation of the sheet, as predicted by DFT.

Furthermore, these results illustrate the (very) non-local nature of the charge redistribution as a result of polarisation by nearby charges. Charges are observed to be redistributed across the entire surface, with positive charge being accumulated near the point charge, and positive charge being depleted far away from the center. These results indicate the importance of charge transfer effects in correctly describing the charge redistribution under the influence of a polarising charge. Moreover, these result show that the charge induction effects can be sizable, confirming the importance of taking them into account in the study of metallic/semi-metallic electrode-electrolyte interfaces.
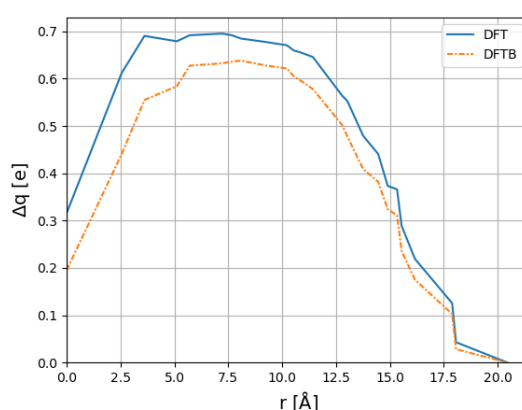
Figure 4.7: Plot depicting the DFT (solid) and SCC-DFTB (dash-dotted) summed excess charge Δq as a function of the radial distance r from the center of a single sheet of (100) copper, which is polarised by a point charge (q = -1.0 e) at a height of 3 Å above the center of the sheet. Up to a certain level, SCC-DFTB is shown to be able to capture the polarisation of the sheet, as it is predicted by DFT. SCC-DFTB can be seen to underestimate the total polarisability.

(a) DFT   (b) SCC-DFTB
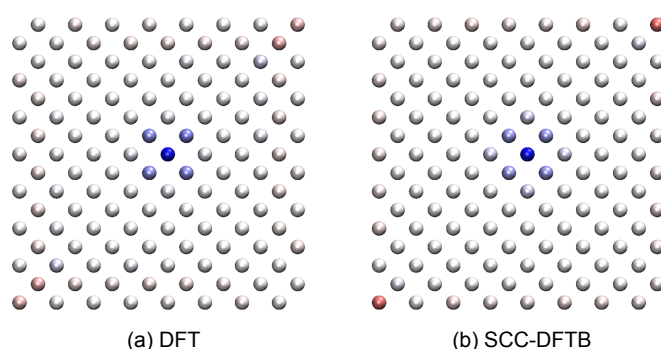
Figure 4.8: Visualisation of a single layered (100) copper sheet, coloured using the computed partial charges q^i by both DFT (left, a.) and SCC-DFTB (right, b.) The same charge scale was used for both plots, from -0.15 e (red) to 0.15 e (blue). Charges outside of this range were capped, and were assigned the maximum colour values. A difference between the predictions by SCC-DFTB and DFT can be observed at the right top and bottom left corners. DFT predicts a negative charge on the two outermost atoms on both sides of the sheet, while SCC-DFTB predicts that the charge is mainly localised on the single outermost atoms.

(a) DFT   (b) SCC-DFTB
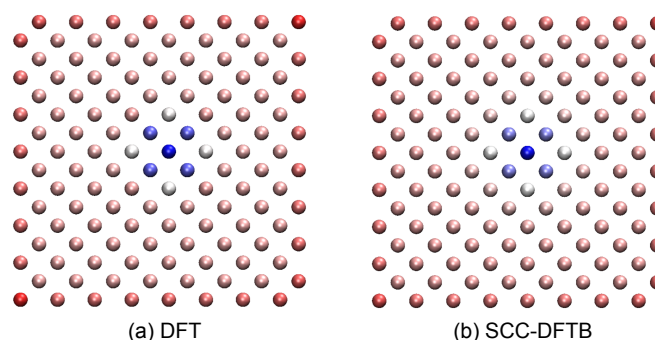
Figure 4.9: Visualisation of a single layered (100) copper sheet, coloured using the computed excess charges Δq^i by both DFT (left, a.) and SCC-DFTB (right, b.) The same charge scale was used for both plots, from -0.05 e (red) to 0.1 e (blue). Charges outside of this range were capped, and were assigned the maximum colour values. Being directly related for figure 4.7, SCC-DFTB here demonstrates being able to capture polarisation of the copper sheet by a point charge. DFT can be seen to polarise the sheet more than SCC-DFTB (darker blue colour around the center). Like figure 4.7, this figure illustrates the non-local nature of the charge redistribution as a result of polarisation by the point charge. Charge is observed to be redistributed across the entire surface, with positive charge being accumulated near the negative point charge, and positive charge being depleted far away from the center.

Still, differences between the predictions of DFT and SCC-DFTB can be observed. Figure 4.7 shows that SCC-DFTB underestimates the polarisability of the sheet by up to 20%. Furthermore, figure 4.8 shows that SCC-DFTB and DFT predict slightly different partial charges for the atoms in the top right corner and the bottom left corner. DFT predicts a negative charge on the two outermost atoms on both sides of the sheet, while SCC-DFTB predicts that the charge is mainly localised on only the outermost atoms. As previously described, this result follows from the predicted partial charges in a vacuum. Therefore, this difference is thought to be a result of the SCC-DFTB parameterisation, in the same way the predicted charge on the boundaries for the three layered sheet in a vacuum was different between SCC-DFTB and DFT. In this case, however, it does not lead to a large difference in the predicted partial charges or predicted polarisation.

The effect of the boundaries on the observed difference is validated by looking at the results of the simulations of smaller single layered sheets in appendix B.2. There it is shown that the larger the sheets become, the smaller the relative error becomes. Other, not observed, size effects could play a role here too. Therefore, it is recommended to perform these type of size dependent simulations when benchmarking SCC-DFTB performance.

The supposed boundary related errors are thought to be a minor issue in the application of SCC-DFTB in the rest of this thesis, as periodic boundaries are used throughout. Because periodic boundary simulations were not possible in DFT, additional non-periodic experiments were performed with multiple point charges of opposite charge. In this way, it could be seen whether SCC-DFTB and DFT predicted different behaviour in between opposite charges, a situation that might occur regularly in the QM-MD simulation in which the boundaries are not expected to play a significant role.

These experiments generally showed the same behaviour as observed in the other experiments on one layered sheets. Slight differences were observed in the predicted partial charges near the boundaries resulting from the vacuum configuration. Differences in the predicted polarisation were considered a consequence of the seemingly smaller polarisability of the SCC-DFTB simulations compared to DFT. An example of an experiment with multiple charges can be found in appendix B.3.

Lastly, figure 4.10 shows the result of performing the same type of simulations on a single sheet of (100) copper consisting of 113 atoms, for two different distances of the point charge. As anticipated, in both DFT and SCC-DFTB, the polarisation of the sheet increases as the point charge is moved closer to the sheet.
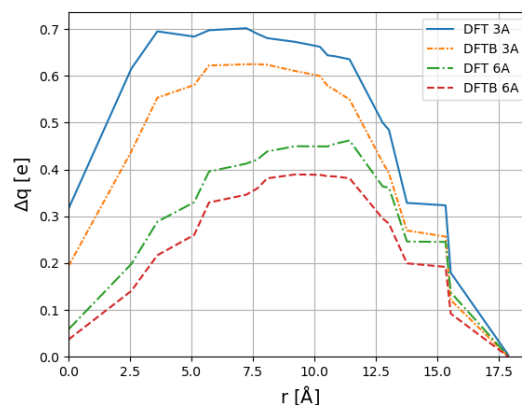
Figure 4.10: Plot depicting the DFT and SCC-DFTB summed excess charge Δq as a function of the radial distance r from the center of a single sheet of (100) copper, which is polarised by a point charge (q = -1.0 e) at heights of 3 and 6 Å above the center of the sheet. The results of both DFT and SCC-DFTB show that the polarisation of the sheet increases as the point charge is moved closer to the sheet.

Overall, boundary effects and an underestimation of the total polarisability in SCC-DFTB were taken to be the main contributors to the observed differences between the SCC-DFTB and DFT results. In the tight-binding approximation the interaction between different atoms is limited. Therefore, a priori and without looking at the parameterisation, it could be expected that SCC-DFTB would underestimate the total polarisability.

SCC-DFTB, using the matsci-0-3 set of SK-files, in comparison to DFT results, seemed to be able to capture the polarisation effects on a single sheet largely correctly. This is most important if the interest is to gain more insight into the effect of charge redistribution under the influence of a set of point charges. Moreover, other errors, for example, the errors attributed to boundary effects, were not found to influence the predicted partial charges to the point that the resulting configurations of DFT and SCC-DFTB contradicted each other. Therefore, the set of SK-files used with SCC-DFTB was deemed suitable to proceed with simulating the charge redistribution on a single sheet of (100) copper.

4.3. **Choosing the QM-MD coupling time interval**

Here it is described how the time interval between the MD and SCC-DFTB simulations was chosen. As described in the method, there is a trade-off between accuracy of the simulation and the practical viability of the loop. The QM computations must be used often enough that the representation of the polarisation is accurate and physically meaningful results are obtained. At the same time, the total simulation time must be kept down. Moreover, to to successfully train a neural network on the results, a diverse training data set is required.

First, a QM-MD test simulation was performed, in which the average and maximum charge differences were calculated between SCC-DFTB calculations, for varying time intervals between the calculations. This did not yield a suitable estimate for the time interval. Therefore, MSD measurements were performed on ions adsorbed on the wall, using which a suitable time interval was estimated. The QM-MD test simulation is described in section 4.3.1. The MSD measurements and the choice for the time interval are described in section 4.3.2.

4.3.1. **QM-MD test simulation**

To get a well suited estimate for the time interval between SCC-DFTB computations, a QM-MD test simulation was performed. This simulation consisted of multiple different consecutive smaller loops, each consecutive loop using a different time interval and initialised from the last. First, the system was simulated for 10 ps, with an SCC-DFTB calculation every 50 fs. Subsequently, the system was simulated for 1 ns, with a time interval between SCC-DFTB calculations of 5 ps. Then, the system was simulated for 2.5 ps, with again time intervals of 50 fs. Lastly, the system was simulated for 200 fs, with a time interval between SCC-DFTB calculations of 1 fs.

In the test simulation, there was a loop with a short time interval between SCC-DFTB calculations (1 fs), and a loop with a relatively long time interval between SCC-DFTB calculations (5 ps). If the entire test simulation was performed using the short time interval, the system would probably not have reached an equilibrium in the total simulated time. As a consequence, no data would be obtained about how the system would behave in equilibrium. It could be, for example, that initially the instantaneous change in the ion configuration, and thus the surface polarisation, is larger than when the system would have been allowed to reach an equilibrium. On the other hand, if the chosen test time interval had been chosen to be 5 ps for the entire simulation, the system might have been able to reach an equilibrium, but this equilibrium would have been influenced by the chosen (long) time interval. In additions, this way the two different loops with a time interval of 50 fs could be compared, potentially yielding information about changes in the system during.

The average and the maximum change in the partial charges relative to some other point in time was measured. To obtain a statistically more robust result from the relatively short runs, a 'sliding origin' was used to compute the change. This means that, instead of measuring the (average or maximum) change with respect to the configuration at $t = 0$ for every result in the entire run (from $t = 0 \rightarrow t = $ end), an new measurement was performed after every SCC-DFTB calculation over a shorter period of time (from $t = i \rightarrow t = $ end$_i \forall i$ for which end$_i \leq$ end). This shorter measurement time was chosen to be a fourth of the total run time, such that for the final fourth of the run time, no new measurements were started. To obtain the final results, the different sets of results were combined and averaged. The average change in the partial charges on the electrode, and the maximum change in a partial charge as a function of the simulation time, are plotted in figures 4.11 and 4.12 for the 5 ps and 1 fs interval times, respectively. The results of the 50 fs loops can be found in appendix C.

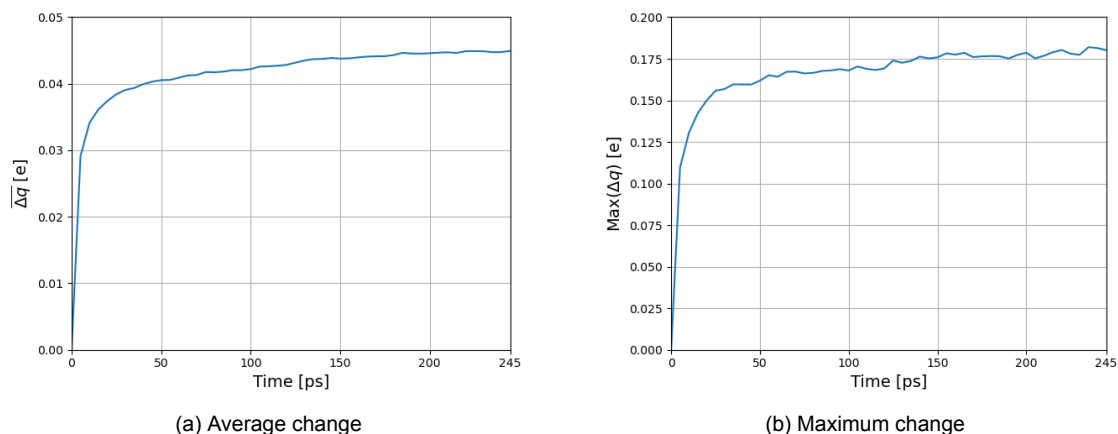(a) Average change

(b) Maximum change

Figure 4.11: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, using a time interval between SCC-DFTB calculations of 5 ps.
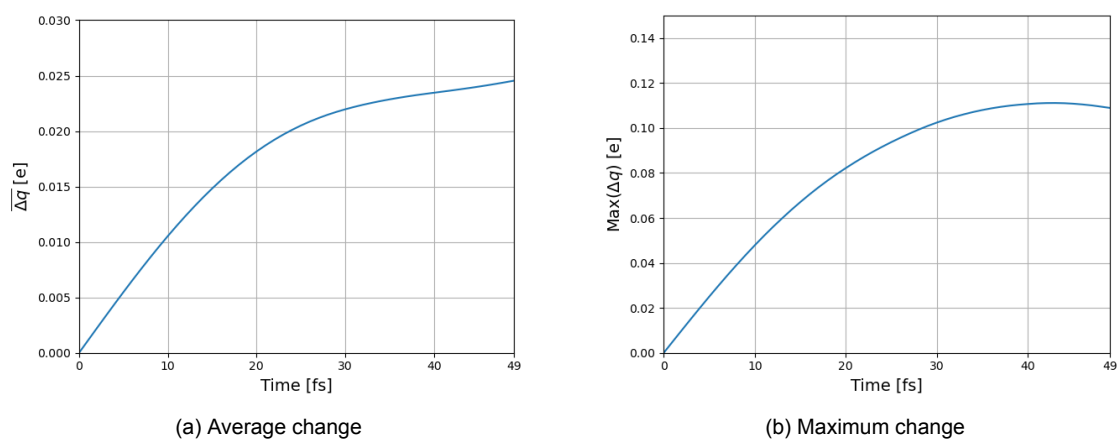

(a) Average change

(b) Maximum change

Figure 4.12: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, using a time interval between SCC-DFTB calculations of 1 fs. After 49 fs, both the average and the maximum change in the charge values, exceed 50% of the average and maximum change values observed after 245 ps, as seen in figure 4.11

As anticipated, both figures show that the average change increases as a function of time. By comparing the two figures, it is seen that within 49 fs, the average change in figure 4.12, exceeds a value of 50% of the largest average change observed after 245 ps in figure 4.11. The same holds for the maximum error observed in both plots. Consequently, if the decision of the time interval between SCC-DFTB calculations were to be based on these measurements, a time interval well below 49 fs would have to be chosen in order to construct a physically consistent loop. With a time interval below 49 fs, the total QM-MD loop simulation time would be severely limited. Therefore, it was decided that the time interval should be determined in a different way.

4.3.2. Estimating the time interval using MSD measurements

In order to obtain another estimate for a suitable time interval between SCC-DFTB calculations in the QM-MD loop, MSD measurements were performed using the potassium ions.

An MSD measurement was performed for 2.5 ns. The measurement was initialised from a state after 25 ns of simulation time of the previously performed MD run, of which it was observed that after 10 ns the MD system had reached an equilibrium (by checking density plots and pair correlation functions, using block averages of various lengths). Choosing the starting point after 25 ns, allowed for the evaluation of the final 5 ns of the simulation to choose a suitable set of atoms. The MSD was calculated for eight potassium atoms, that during the simulated 2.5 ns, remained adsorbed on the cathode. Like was done in the case of the previously presented results of the average and maximum charge change in the partial charges, a 'sliding origin' approach was followed in order to obtain a statistically more robust result.

The MSD of the selected atoms was calculated every 100 fs. Every 50 fs, a new MSD measurement was initialised, and each MSD measurement was performed for 50 ps. A plot of the result can be found in figure 4.13. Subsequently, a weighted ($w = 1/\sigma^2$) least squares fit was performed. Only data obtained after 1 ps of measuring was used for the fit, not the full time range. Data points from times below 1 ps fall into the ballistic regime, and only represented particle movement due to Brownian motion.[111]
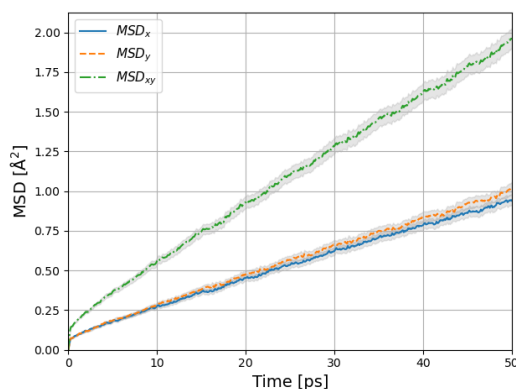
Figure 4.13: Plot of the averaged MSD trajectory of a selection of potassium ions, adsorbed on the cathode throughout the simulation of 2.5 ns. The corresponding standard errors ($\sigma/\sqrt{N_{\text{measurements}}}$) are plotted in grey around the measurements. The MSD is plotted for three directions: the x-, y- and x,y-direction, where the x,y-direction is simply the sum of the individual x- and y-results.

By means of the fit to the slope of the MSD measurements, an estimate for the diffusion coefficient in direction parallel to the wall was obtained using eq. (3.2), yielding a value of $7.39 \pm 0.03 \cdot 10^{-11}$ m^2/s. The diffusion coefficient provides information on the movement of particles in a system. By taking the square root of the obtained diffusion coefficient multiplied by a time of 1 ps (displacement = $\sqrt{D \cdot 1\text{ps}}$), an estimate was obtained for the expected average displacement of the potassium ions near the wall of $8.60 \pm 0.02 \cdot 10^{-2}$ Å/ps. The sought-after time interval was chosen based on the time the potassium ions near the wall would take to traverse one-tenth of the lattice constant of copper (3.6149 Å), yielding 4 ps after rounding down.

It is clear that, ultimately, the chosen time interval was based on several assumptions, and not derived rigorously. The effect of this choice is discussed further on, after the results of the QM-MD run have been presented.

4.4. QM-MD simulation

Here the results of the QM-MD simulation are presented and discussed. Using the previously chosen time interval of 4 ps between SCC-DFTB calculations, the QM-MD loop was simulated for 30 ns. The results presented here were all obtained using gathered data from the final 10 ns of the simulation.

As a result of the findings presented in section 4.2, the choice was made to only simulate the front layer of the copper cathode using SCC-DFTB. Consequently, the full net charge of the cathode (-4.0 e) was assigned to that layer only. For full consistency, the same net charge was assigned to the front layer of the cathode in the MD-only run (from here on out referred to as the 'classical-MD' (C-MD) simulation). This yielded a charge of $-4/128 = -0.03125$ e on every atom in that layer in the C-MD simulation. Figure 4.14 provides an impression of the charge distribution on the copper electrode during the final 10 ns of the QM-MD simulation, in the form of a histogram of the partial charges calculated in the corresponding SCC-DFTB simulations.

A significant redistribution of the charge on the cathode was observed, away from the average value of -0.03125 e. Both negative and positive charges were induced in the QM-MD simulation. The depicted charge distribution does not show a normal distribution with a peak around the average partial charge value, as was seen in the study of graphene using the QM-MD loop by Elliot et al.[14][34]. Instead, figure 4.14 shows a peak for partial charges more negative than the average value of -0.03125 e, and a second peak for partial charges larger than the average value, from which the primarily mentioned charges seem to have drawn their additional negative charge.
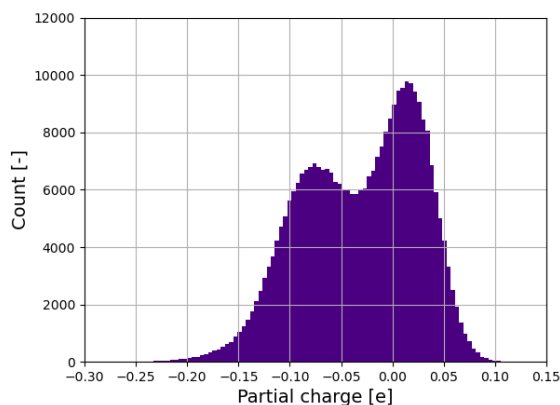
Figure 4.14: Histogram of the partial charges calculated in the SCC-DFTB simulations during the final 10 ns of the QM-MD simulation. A significant redistribution of charge on the cathode is observed, when compared to the average charge of -0.03125 e that was used in the C-MD simulation.

In the following, the effect of this charge redistribution is explored. To compare the results of the C-MD simulation with those of the QM-MD simulation, multiple different results are presented. Firstly, density plots, cumulative distribution function plots constructed from the measured densities, and radial distribution functions are presented related to the electrolyte of the system. Secondly, MSD measurements of the system are given, and lastly, histograms of the water molecule orientation angle are presented.

4.4.1. Electrolyte configuration

The pressure in the system was set by putting an inward force, equal to atmospheric pressure, on the two electrodes, as was described in the method. During the initialisation stage, this yielded fluctuations in the positions of the electrodes, and thus, fluctuations in the bulk density of the system. In both the C-MD simulation and the QM-MD simulation, the bulk density was too high. The bulk density of the aqueous potassium chloride solution, was measured to be $1049.0 \pm 4.6 \ \mathrm{kg/m^3}$ in both simulations, as compared to the expected value of $1041.5 \ \mathrm{kg/m^3}$.[112].

A plot of the electrolyte density, and a plot of the cumulative distribution function of water in the system, constructed from the measured water density, are provided in figure 4.15. As for all density plots in this section, a bin size of 0.1 Å was used to compute the profiles. Furthermore, the standard error in the computed densities was plotted in grey around the data. It should be noted, that this error is usually so small (relative to the data) that it cannot be observed in the plots.

(a) Density profile of the electrolyte   (b) Cumulative distribution function of water
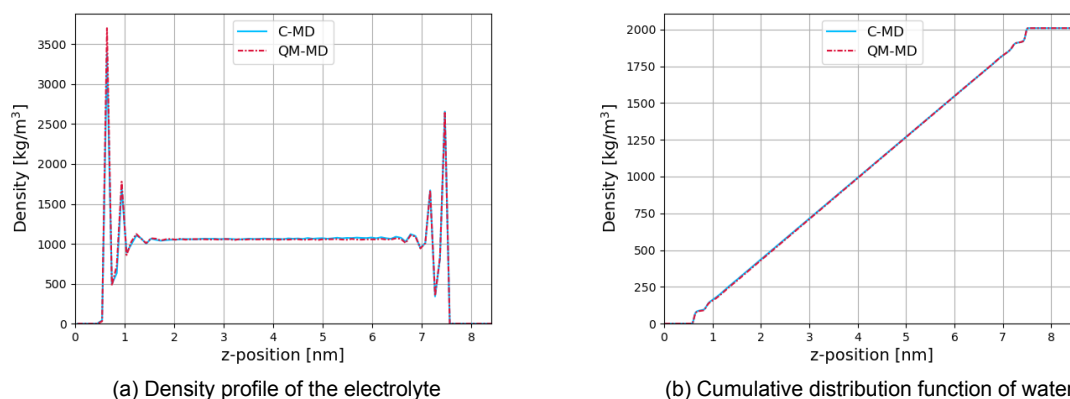
Figure 4.15: (a) Density plot of the electrolyte, plotted for both the C-MD simulation (solid line), and the QM-MD simulation (dash-dotted line). In both simulations, the bulk density was found to be higher than the expected value of $1041.5 \ \mathrm{kg/m^3}$. A difference between the C-MD results and the QM-MD results, with relative more weight being located near the cathode, which is located on the left, and relatively less in the bulk. (b) Plot of the cumulative distribution function of water. A total of 2009 water molecules were simulated.

The density plot, the cumulative distribution function plot and the radial distribution function plot of potassium can be found in figure 4.16. For chloride these are given in figure 4.17. The pair correlation functions of the respective ions can be found in figure 4.18. The cathode is located on the left side of the plots, with the front layer at a location of 3.6149 Å.

All figures clearly display a significant redistribution of the ions in the system as a result of the polarisation effects. On average, over 40% more potassium ions were adsorbed on the cathode surface. Correspondingly, on average, more chloride atoms were observed in the double layer near the cathode in the QM-MD run than in the C-MD run. Because more ions were drawn to the cathode, for both ion types this led to a relative decrease in the number of ions in the bulk of the system (at locations over 4 nm), as is shown in the density and cumulative distribution function plots.

(a) Density profile of potassium

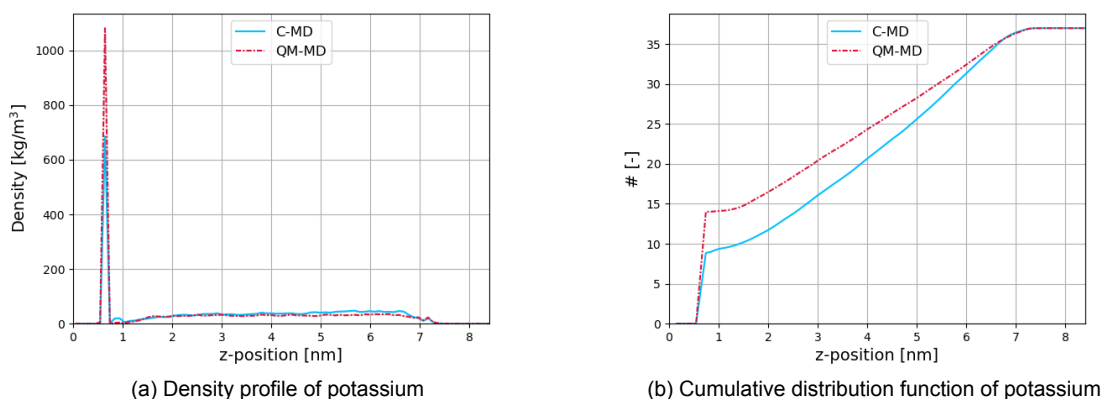(b) Cumulative distribution function of potassium

Figure 4.16: (a) Density plot of potassium, plotted for both the C-MD simulation (solid line), and the QM-MD simulation (dash-dotted line). (b) Plot of the cumulative distribution function of potassium. Both results show an increase in the number of adsorbed potassium ions at the cathode.
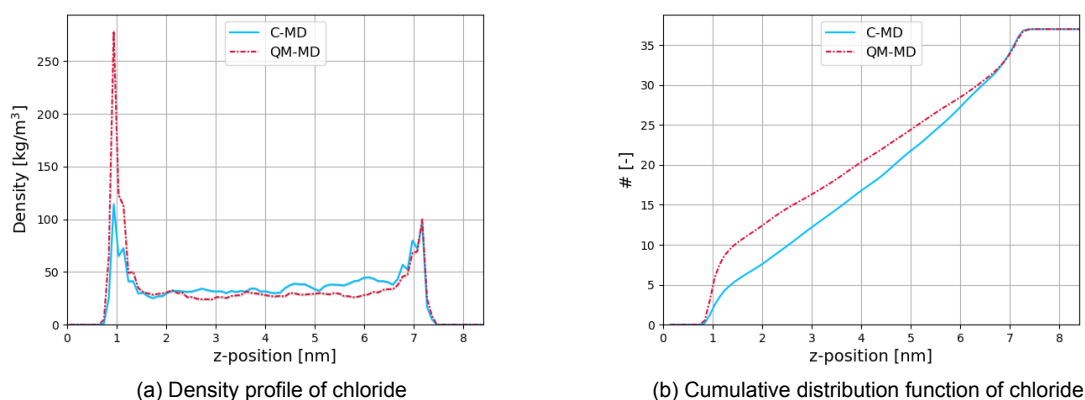

(a) Density profile of chloride

(b) Cumulative distribution function of chloride

Figure 4.17: (a) Density plot of chloride, plotted for both the C-MD simulation (solid line), and the QM-MD simulation (dash-dotted line). (b) Plot of the cumulative distribution function of potassium. Corresponding with the observed increase in the number of adsorbed potassium ions in figure 4.16, both results show an increase in the number of chloride ions in the double layer at the cathode.

(a) Potassium
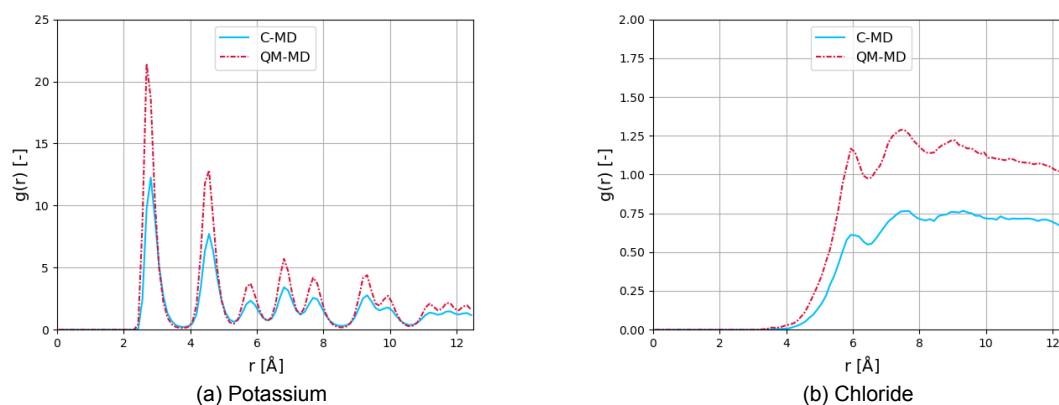
(b) Chloride

Figure 4.18: Pair correlation functions of potassium (a, left) and chloride (b, right), calculated with respect to the front layer of the cathode, and plotted for both the C-MD simulation (solid line), and the QM-MD simulation (dash-dotted line). The depicted pair correlations functions overlap with the increase of the number of ions in the double layer in Q-MD results, relative to the C-MD results, as seen in figures 4.16 and 4.17. In both plots, the ions can be seen to be drawn slightly more towards the cathode surface.

The plotted pair correlation functions show that polarisation effects only led to slight changes in the adsorption distance of potassium ions, drawing the potassium ions maximally 0.1 Å closer. In the QM-MD run, the potassium atoms were generally adsorbed in between four copper atoms of the front layer (and a copper atom in the second layer), as the snapshots given in figure 4.19 portray. This was observed in the C-MD run as well, but as the MSD measurements will show later on, the potassium atoms were much more immobilised in this configuration in the QM-MD run. This shows that, indeed, negative charge got accumulated where potassium ions were close to the surface. By taking into account polarisation effects, the cathode surface was able to adapt its potential energy surface by means of a redistribution of the charge. In this case, this seemed to lead to an increase in the number of adsorbed potassium ions in the QM-MD run over the C-MD run.

(a) C-MD   (b) QM-MD
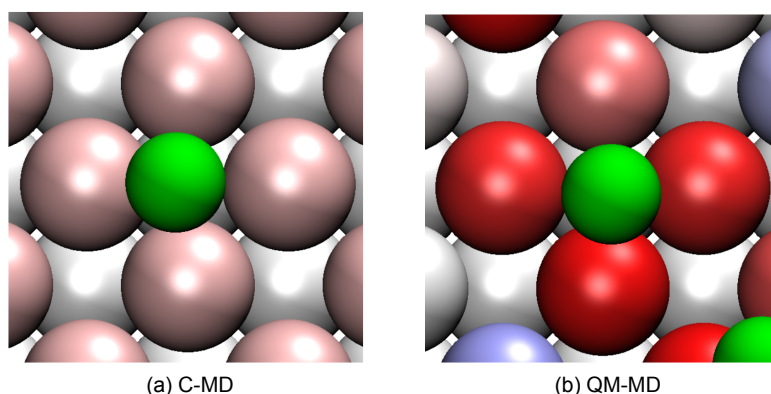
Figure 4.19: Snapshots of a potassium ion (green) adsorbed on the cathode surface, in both the C-MD run (a, left) and the QM-MD run (b, right). The copper sheet atoms are coloured based on their respective charge, with red indicating a negative charge and blue a positive charge. In the QM-MD simulation, a potassium ion adsorbed on the cathode surface could induce a negative charge on nearby copper atoms, that exceeded the average negative charge of the copper atoms in the C-MD simulation (as shown by the darker colour red). In turn, this would increase the attractive interaction between the copper sheet and the potassium ion. In figure b, Positive charge was induced on two nearby copper atoms as a result of local charge depletion by the surrounding negatively polarised atoms.

A potassium ion near the cathode surface in the C-MD run, experiences an attractive force due to LJ interaction and Coulombic interaction from the negative copper atoms, and a repulsive Coulombic force from the surrounding positive potassium ions. Ultimately, in the C-MD run on average 9 to 10 cations were adsorbed. The number of potassium ions adsorbed on the cathode can exceed the total negative charge on the cathode, due to the strength of the LJ interaction between the copper atoms and the potassium ions. The potassium ions that were not adsorbed, were further prevented from adsorbing on the cathode, because they were repulsed by the net positive charge they experience from surplus of positive charge near the cathode, either before they reached the surface, or after.

By allowing for the induction of negative charge due to the nearby presence of a positive potassium ion, the attractive interaction between the local cathode atoms and the ion was increased. This way, more positive ions could be adsorbed on the surface, without having assigned additional negative charge on the cathode. In the context of the configuration of copper atoms seen in figure 4.19, polarisation allowed for a deepening of the local potential well, where the potential wells in the C-MD simulation was not deep enough to hold an additional ion.

By drawing negative charge towards the adsorbed potassium ions, at times, positive charge is induced in between the adsorbed ions, as was seen in figure 4.14. This indicates a depletion of the local negative charge. The repulsive force the induced positive charge yields on potassium ions nearing the cathode from the bulk probably plays a role in why not more potassium ions are adsorbed. Because of the relatively small LJ interaction chloride ions experience compared to the other atom types in the system, they are generally not adsorbed on either electrode surface.

Bicarbonate was attracted to the oppositely charged electrode in the MD simulations shown in section 4.1. If bicarbonate was used in the QM-MD simulation, it is expected that this would have led to a local expulsion of negative charge. In turn, this could have led to increased attractive interaction of the cathode with positive ions, either due to delocalised negative charge on the cathode, or due to the extra negative charge of the bicarbonate molecule on the surface. Moreover, for anions with a LJ interaction magnitude below that of the bicarbonate molecules, but above that of chloride, it is expected that there is a point where the anion would not adsorb in a C-MD run, but would adsorb in the QM-MD run. Whether or not it would adsorb, would then also depend on how much positive charge is induced on the cathode. This would yield entirely different behaviour near the respective electrolyte. An extra accumulation of chloride ions is observed in this study, as a response to the increase in the number of adsorbed potassium ions on the cathode. For bicarbonate, a similar effect could be expected, in addition to increased adsorption times.

The same holds for cations. While the observed redistribution of cations in the simulated system was significant in itself, it is expected that for some types of ions the effect can be even more dramatic. For cations with smaller LJ interaction with the respective electrode than potassium, taking into account polarisation effects could lead to the adsorption of the cation where in a C-MD run it would, incorrectly, not adsorb.

Finally, figure 4.20 shows the net cumulative charge difference between the C-MD simulation and the QM-MD simulation. In this plot, the observed increase in the number of adsorbed potassium ions at the cathode translates into a maximum net charge difference of 5 e. Subsequently, the net charge difference goes to zero within the double layer, because of the corresponding increase in the number of present chloride ions.

Concluding, in this section it was seen that electrostatic effects played a significant role in the ion adsorption process in the system. Locally induced negative charges were seen to increase the attraction between potassium and the cathode. As a result, a significant redistribution of the ions in the system was found.
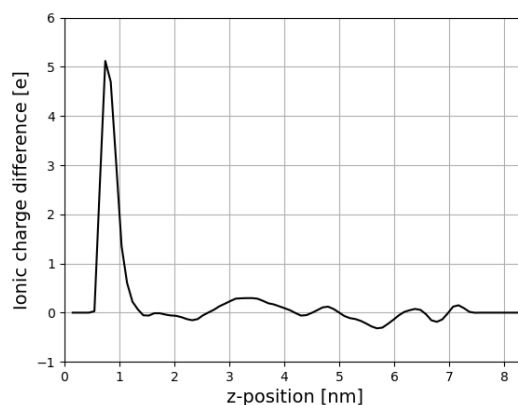
Figure 4.20: Plot of the net cumulative charge difference between the C-MD and QM-MD simulations. The increase in the number of adsorbed potassium ions on the cathode surface in the QM-MD simulation, relative to the C-MD simulation, leads to a maximum net charge difference of over 5 e near the cathode. The net charge difference goes to zero within the double layer, because of the corresponding increase in the number of present chloride ions.

4.4.2. MSD measurements

MSD measurements were performed using both the C-MD and QM-MD simulations. Potassium atoms adsorbed on the cathode, as well as potassium ions residing in the bulk, were used in separate MSD measurements. Subsequently, the differences between the to simulations were evaluated. Moreover, the measured MSD in the bulk was used to compute a local diffusion coefficient, that could be com-

pared with values known from literature. This served as a way of verifying that the system is functioning as it should.

The MSD results for the potassium ions adsorbed on the cathode in the C-MD simulation, that were used to obtain an estimate for a suitable interval time between SCC-DFTB calculations (i.e., 4 ps), were obtained in the previously described measurements. The other MSD measurements were performed in a similar manner. The total measurement time was 2.5 ns. 14 potassium atoms were selected for the bulk MSD measurements, in both the C-MD and QM-MD simulations. For the MSD measurements on the atoms adsorbed on the cathode, 12 potassium atoms were selected in the QM-MD simulation (more than in the C-MD experiment, as more were available/adsorbed), that during the simulated 2.5 ns remained adsorbed on the cathode. Again, the 'sliding origin' approach was applied, and all timescales were chosen to be the same as in the previously performed MSD measurement. Plots of the results can be found in figures 4.21 and 4.22, for the atoms in the bulk and the atoms adsorbed on the cathode, respectively. In appendix D the MSD trajectories of the individual atoms are given.

(a) C-MD   (b) QM-MD
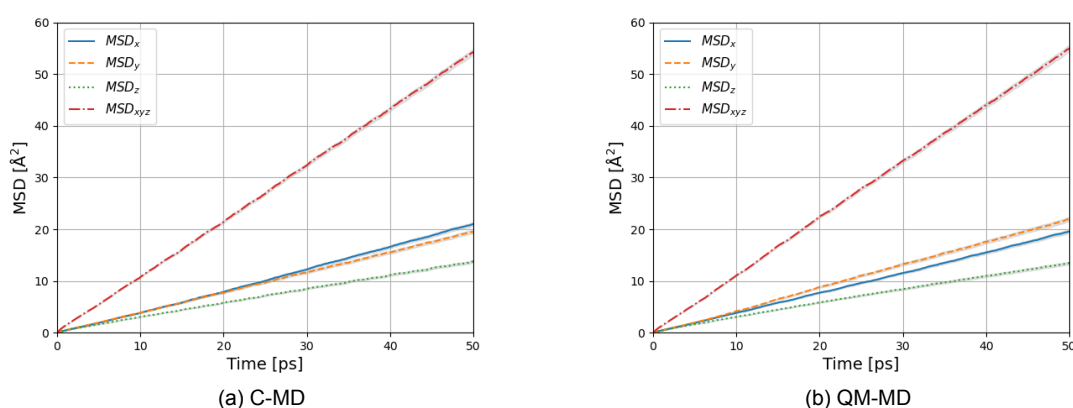
Figure 4.21: Results of the MSD measurements performed on atoms in the bulk, for both the C-MD simulation (a, left) and the QM-MD simulation (b, right). The corresponding standard errors ($\sigma/\sqrt{N_{\text{measurements}}}$) are plotted in grey around the measurements. The MSD in the x-, y- and z-directions are all found to be similar. The MSD in the x- and y-directions is significantly larger than the MSD in the z-direction, which is the direction bound by the two electrodes.

(a) C-MD   (b) QM-MD
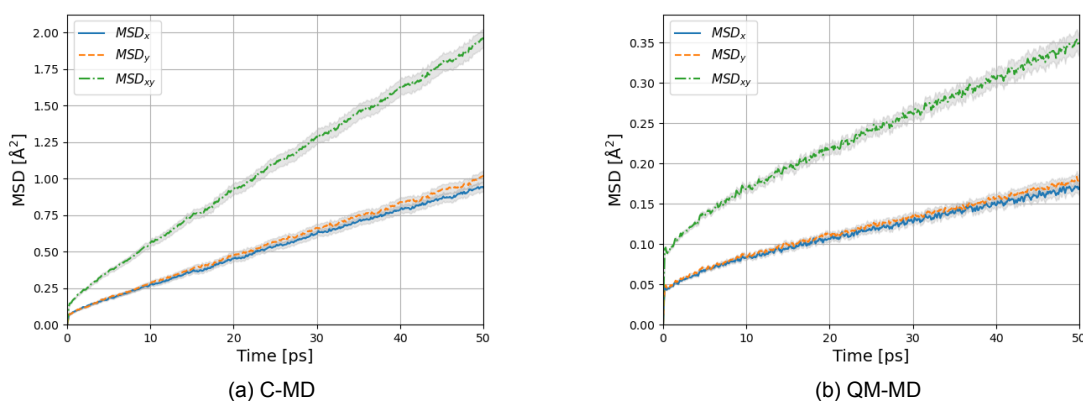
Figure 4.22: Results of the MSD measurements performed on atoms adsorbed on the surface of the cathode, for both the C-MD simulation (a, left) and the QM-MD simulation (b, right). The corresponding standard errors are plotted in grey around the measurements. The MSD in both directions of the measurement performed using the C-MD simulation, is found to be significantly larger than the MSD across the cathode surface in the QM-MD simulation. Previously, it was described how this could be a result of the induced negative charge on the cathode, yielding a situation as depicted in figure 4.19.

The MSD of the atoms in the bulk in the x-, y- and z-directions in the C-MD simulation were similar to those in the QM-MD simulation. This indicates that the polarisable cathode surface has no significant influence on the dynamics of atoms not near that surface, as was anticipated. The MSD in the x- and y-directions was significantly larger than the MSD in the z-direction. This was also anticipated, as this is the direction bound by the two electrodes. The differences between the x- and y-measurements in the C-MD and QM-MD simulations can probably be attributed to the limited simulation time of 2.5 ns that was used to obtain these results. The large correspondence between the two sets of bulk results is considered to provide sufficient confidence in the accuracy of the MSD measurements performed to allow them to be used in this thesis.

Taking into account polarisability of the cathode surface significantly altered the dynamics of the adsorbed cations. For the atoms adsorbed on the cathode surface the MSD in the C-MD simulation was significantly larger than the MSD of the adsorbed atoms in the QM-MD simulation, in both the x- and the y-direction. In section 4.4.1 (figure 4.19) it was explained how induced negative charges on the cathode surrounding the cation, yielded a potential well wherein a cation could be adsorbed. These results showed that, this way, the cations are comparatively trapped inside this structure, resulting in immobilisation of the cation and, correspondingly, a significant decrease in the measured MSD.

Diffusion coefficient

The MSD results obtained in the QM-MD simulation of the potassium ions residing in the bulk were processed using MATLAB[113]. Using a weighted ($w = 1/\sigma^2$) least squares fit, diffusion coefficients were computed for each of the measured directions. The bulk diffusion coefficient D_{xyz} was found to be $1.710 \pm 0.002 \cdot 10^{-9}$ m^2/s. This is lower than the 1.89 m^2/s described by the Joung and Cheatham force field[74][114], for 1M of dissolved potassium chloride in water modelled using the SPC/E force field.

This dissimilarity can primarily be attributed to the relatively lowered diffusion in the z-direction, for which a diffusion coefficient D_z was estimated to be $1.376 \pm 0.003 \cdot 10^{-9}$ m^2/s. The diffusion coefficient in the x- and y-directions D_{xy} was estimated to be $1.8768 \pm 0.005 \cdot 10^{-9}$ m^2/s, which is closer to the value of $1.89 \cdot 10^{-9}$ m^2/s from literature. Based on this estimate, it was concluded that the system behaved appropriately.

4.4.3. Angular orientation of the water molecules

Over the final 10 ns of the simulations, for water molecules near the cathode surface, histograms were made of the angular orientation of water molecules. For multiple distances from the cathode, for all water molecules present, the angle of the reversed dipole moment with respect to the normal of the cathode surface was computed. The resulting histograms can be found in figure 4.23.

The relative distances from the cathode, used to obtain the histograms, were chosen based on observed average distances of the water molecules from the cathode surface. The majority of the water molecules in the first superficial layer of adsorbed water were contained within 3.5 Å from the cathode surface, together with the adsorbed potassium ions. The distances of 2.9 Å and 3 Å were chosen to be in the beginning of the first superficial layer of adsorbed water. Within 5 Å, also the first water molecules within the next superficial layer of water were contained.

As depicted in figure 4.23, an angle of 90° corresponds to the parallel alignment of the water dipole with the surface of the cathode (90° with respect to surface normal). Smaller angles correspond to a tilting of the oxygen atom towards the cathode, whereas larger angles correspond to a tilting of the other end of the dipole (the hydrogen atoms) towards the surface. Both the C-MD and the QM-MD simulation results show a preferential angular orientation of smaller than 90°. This preference was larger in the case of the QM-MD results than in the case of the C-MD results. For the C-MD results, 61% of the observed water molecules at distances smaller than 3.5 Å had a angular orientation smaller than 90°, compared with over 80% in the QM-MD simulation.

(a) C-MD   (b) QM-MD
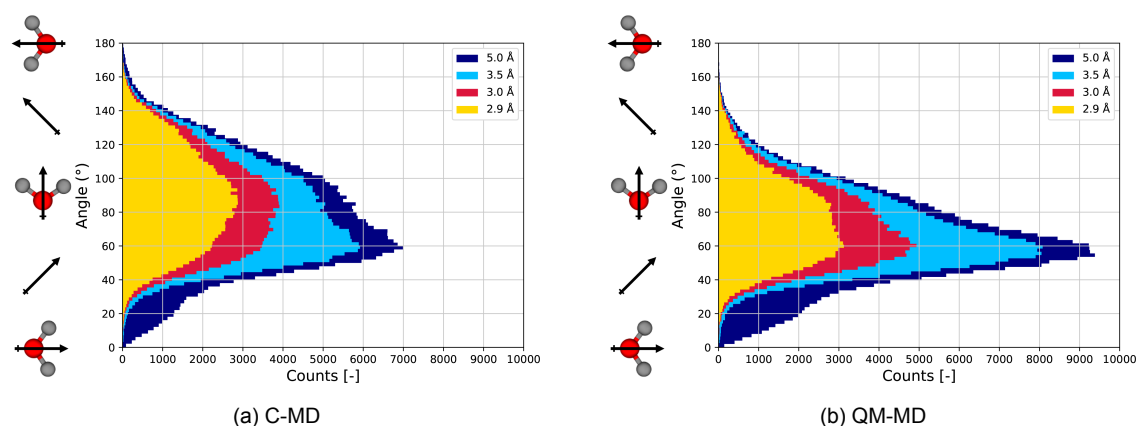
Figure 4.23: Histograms of the angular orientation of water molecules near the cathode, in the C-MD simulation (a, left) and the QM-MD simulation (b, right). Black arrows denote the orientation of the dipole for a given angle. The colours relate to the maximum distance from the cathode that water molecules were taken into account for a calculation. So, for 5.0 Å, for all water molecules were up to a distance of 5.0 Å from the cathode, the angle with respect of the normal of cathode was computed and stored in the dark blue histogram. Both the results of the C-MD and the results of the QM-MD simulation show a skewed profile, with a preferential angular orientation smaller than 90°. In the QM-MD simulation, this preference is relatively increased compared to the C-MD simulation.

This result was not fully anticipated. A priori, it was be expected that the water molecules would have a preferential angular orientation larger than 90°, due to the attraction between the two positive hydrogen atoms and the negative copper atoms. The results were verified by counting the number of cases in which the z-position of the oxygen atom was smaller than the average z-position of the hydrogen atoms, for all water molecules in used in the histogram computation. This number, indeed, overlapped exactly with the number of cases in which the angular orientation of the water molecules was smaller than 90°. In the following it will be attempted to rationalise these results.

The large cation concentration at the cathode, especially in the QM-MD simulations, significantly contributed to the observed preferred angular orientation. As a result of the electrostatic interaction between the two, it would be expected that the oxygen atom in a water molecule would be oriented towards the potassium ion. Due to Pauli repulsion, there is little to no space between the potassium ion and the copper atoms. Consequently, this effect would be expected to lead to a preferred angular orientation with an angle smaller than 90° in the first superficial layer of water adsorbed on the cathode.

The z-coordinates of both hydrogen atoms were compared to the z-coordinate of the corresponding oxygen atom in all water molecules. In the C-MD simulation, it was found that in more than 92% of the cases at least one of the hydrogen atoms was closer to the cathode than the corresponding oxygen atom. In the QM-MD simulation, this number decreased to 89%. Because the angle between the hydrogen atoms in a water molecule is modelled to be 109.47°, angular orientations of water molecules down to 35.265° can theoretically still have one hydrogen atom with a z-coordinate smaller than that of the corresponding oxygen atom. This finding makes the found preference in angular orientation of the water molecules somewhat less surprising. This way, the oxygen atom can be close to a nearby potassium ion, while simultaneously having at least one of the hydrogen atoms closer to a nearby negative charge.

Another factor that might have played a role in determining the preferential angular orientation of the water molecules is the fact that the Coulomb force is a product of charges. Since the average charge of the copper atoms is -0.03125 e, it is possible that a nearby potassium ion, with a charge of +1.0 e, could interact more strongly with a water molecule than with a copper atom that is closer by. This could be, for example, one of the reasons why induced negative charges near potassium ions in the QM-MD run do not seem to lead to a significant increase in the number of water molecules with an angular orientation larger than 90°.

Induced positive charges in the QM-MD could have contributed to the skewing of the histograms of the angular orientation of water molecules near the cathode, relative to the C-MD histograms. Positive charges have been observed to be induced in considerable numbers as a result of local charge depletion, in figure 4.14. Moreover, all induced partial charges larger than the average value of -0.03125 e could have contributed to turning water molecules away from high angles, relative to the C-MD simulation.

A relative increase in the number of water molecules with an angular orientation larger than 40°is observed between the two sets of results obtained for distances up to 5 Å away from the cathode. This increase is mainly attributed to the increase in strength of the local electric field, as a consequence of the increase in the number of present potassium and chloride ions in the double layer. It is demonstrated in appendix E that this relative increase in preference for one orientation persists further into the system, by showing the same results for a maximum distance of 7 Å away from the cathode.

Histograms of the angular orientation of the water molecules at the anode were made as well, for water molecule present within 3.5 Å away from the electrode surface. These results can be found in appendix E as well. The C-MD and the QM-MD results showed a similar skewed profile, with a preferential angular orientations of 90°or 120°. In contrast to the cathode results, this was expected. These results show that the water molecules, with barely any ions nearby or adsorbed on the anode, and only positive partial charges on the electrode surface, prefer to be oriented with the negatively charge oxygen atom towards the positive copper electrode (anode is on the right side of the system).

The effect of water-water interaction through hydrogen bonding is hard to evaluate, especially since the angles of the water molecules are spread over a wide range of values. Moreover, an experimentally determined value for the contact angle of water with copper is unknown, and thus can not be compared to the results obtained.

4.4.4. General discussion of the QM-MD simulation
Below, a general discussion of the simulations is provided pointing at several choices made, the weak and strong points of doing so and suggesting future research directions. It is recognised that many of the discussed topics do not come with a quantification of the discussed possible effects. Hopefully, the given discussion provides some insight into what effects could be studied, or should be taken into account, in a potential follow-up study.

(i) The QM-MD method as applied in this thesis, neglects possible charge transfer between the electrolyte and the copper surface. Previously, Zhan et al.[115] have studied this effect for graphitic interfaces. It was shown that the effect of charge transfer occurring between different ions (Cl^-, Na^+, K^+) and the graphitic interface was negligible compared to the overall Coulomb and Lennard-Jones non-bonded interactions at the reported adsorption distances (charge transfer of less than 0.04 e, for a potassium adsorption distance of approximately 3 Å). However, charge transfer was shown to be a very short-ranged interaction, that decreased significantly with distance. Because the adsorption distance of potassium in this thesis was observed to be smaller than 3 Å, potassium ions adsorbed at a distance of 2.5 Å, charge transfer between the potassium ions and the copper electrode might not have been negligible.

On the other hand, however, Li et al.[116], using DFT studies, showed that there was no charge transfer occurring between various ions (Cl^-, Na^+) and a charged silver electrode. Consequently, future studies should perform a separate (DFT) study to gain more insight into the occurrence/effect of charge transfer in the case of a copper electrode. In the case that charge transfer becomes a relevant phenomenon, Elliot et al.[14] noted that the effect could be incorporated in the simulation by means of altering the non-bonded interaction. Possible charge transfer could also be taken into account, by means of fitting the effective charge of the ions near the electrode, as was done in the work of Zhan et al.[115]. This fitting could be used to change the charge of the ion in the MD simulation, where the SCC-DFTB simulation could take this change explicitly into account.

(ii) To obtain an idea of how the system configuration would have changed if a lower time interval had been used, the original idea was to use trained neural networks. However, as will be seen in the following section, this turned out to not be possible. On average, a single static SCC-DFTB calculation of the partial charges on the 128 modelled copper atoms, under the influence of the set of point charges, took 40 seconds using two Xeon E5-2680 v4 processors (28 cores). In comparison, simulating the full MD system for 5 ps on the same system, on average took about 17 seconds on average. This way, it was not possible to decrease the time interval between SCC-DFTB calculations significantly, if the desire was to at least simulate the system for 20 ns.

The estimate of a suitable time interval between SCC-DFTB calculations was considered to be related to the dynamics of ions near the cathode. The choice to relate the time interval to the expected time it would take the average potassium ion to traverse one-tenth of the lattice constant across the cathode, was not based on any previously obtained results. Moreover, this estimation did not, for example, take into account ions that moved with significantly larger velocities towards the cathode, and did not account for the effect of polarisation by the water molecules. To perform the estimation in this way, allowed a physical basis to be ascribed to the choice of the time interval, whether this had physical significance or not. This way, this estimation serves as reference point, which similar future studies could take into account when comparing their results to those presented here.

The MSD measurements performed in the C-MD simulation on which the chosen time interval was based, proved to vastly overestimate the average MSD of the potassium ions adsorbed on the cathode in the QM-MD simulation. This way, the chosen time interval might provide a more accurate description of the surface polarisation than previously anticipated. Still, the choice for the time interval might have significantly influenced the obtained results. For future studies it is recommended to perform more in-depth experiments into the effect of the chosen time interval. These experiments could be based on the test simulation that was performed in section 4.3.1.

(iii) Elliott et al.[14] used a non-periodic system with open boundary conditions in their SCC-DFTB calculations, in combination with a periodic C-MD system. However, for this thesis in which also a periodic C-MD system was used, choosing a periodic system in the SCC-DFTB calculations seemed the logical choice: a non-periodic SCC-DFTB system would certainly result in significant boundary errors. As a consequence, however, self-interaction of the point charges in the SCC-DFTB simulations would form a source of error in this thesis. Therefore, to mitigate the self-interaction of the periodic images of the system, the system was chosen to be multiple nanometers wide, in both the x- and y-directions. In addition, to quantify the effect of self-interaction, results of systems of different sizes would have to be compared to each other. A first shortcoming of this thesis is that these type of comparisons were not performed. Therefore, the effect of self-interaction in the SCC-DFTB calculations cannot be quantified.

(iv) Polarisation effects of the fluid were not taken into account. In this thesis, this could have affected the measured angular orientations of the water molecules near the cathode. For depicting the significant impact of the surface polarisation on the average angular orientation, and the corresponding discussion on how this could affect the angular orientation, this lack of fluid polarisation was not thought to be important. However, for a study in which water polarisation is thought to be important, standard atom-centered polarisable models might be able to properly account for this effect, as significant charge transfer between water molecules and their surrounding is not expected.

(v) The given pair correlation plots showed that polarisation effects only led to slight changes in the adsorption distance of potassium ions, drawing the potassium ions on average 0.1 Å closer. Because the choice of MD force field can influence both the adsorption distance and the local potential gradients, the choice of force field has to be carefully considered in future studies. The choice of force field can make a difference when polarisation effects are taken into account, for example, for being able to correctly model hydration effects on ions, and corresponding ion adsorption behaviour.

(vi) As mentioned in the QM-MD loop description, a Mulliken population analysis[117] is performed at the end of the SCC-DFTB calculations to obtain a set of Mulliken charges, that are taken to represent the partial charges in the system. Essentially, a Mulliken population analysis is a summation of all

occupied atomic orbitals (squared) of each individual atom, to obtain the number of electrons localised on each particular atom. This method is sensitive to the adopted basis set of atomic orbitals. In some cases so much so, that some people in the field advise against putting too much reliance on population analysis.[118]

Still, this was the method of choice in this project, because, firstly, a Mulliken population analysis was inherent to the current implementation of SCC-DFTB in DFTB+. Secondly, a comparison with the DFT level (using a different basis set) was performed beforehand that should have disallowed the usage of the current implementation of SCC-DFTB, if it performed too badly. Thirdly, other methods would also have retained a basis-set dependent effect. Lastly, Elliott et al.[14] previously have shown that there is a full equivalence between on the one hand, the interaction of a point charge and a set of atoms in a DFTB simulation, and on the other hand, the Coulombic interaction between that same point charge and the Mulliken charges imposed on the same set of atoms in an MD simulation. This study was assumed to hold for copper as well.

(vii) Since hydrogen experienced no Pauli repulsion, as it was being modelled using the SPC/E force field[1], it could potentially have come unphysically close to the electrode. This, in turn, could have had a disproportional effect on the observed polarisation due to water molecules adsorbed on the surface. To quantify this effect, simulations should be performed in which water molecules were fully modelled, preferably using a polarisable force field.

(viii) Using only three layers of copper in the MD simulation, made the system less representative of a real system. Using more layers would have yielded a higher attractive force between the copper electrodes and the electrolyte, with likely an increase in both the number of adsorbed atoms and an increase in the adsorption time. Moreover, keeping the copper lattice rigid impacted the interaction between the copper electrodes, and the electrolyte. By allowing the copper atoms to move, the observed adsorptive behaviour would be altered. The choice for a rigid copper lattice, was made for the purpose of easing the training process of a neural network. If the training the networks would prove successful, the networks could be easily extended to incorporate the current location of the copper atoms in the electrode.

(ix) Only modelling the front layer of copper using SCC-DFTB, as modelling three layers turned out not to be possible using the available set of SK-files, has impacted the final results. Significant non-local changes in polarisation were observed in the benchmarking of SCC-DFTB. If the entire lattice could be polarised, charge that currently would be drawn solely from the front layer, could be drawn from multiple layers. Opposite charges induced by charge depletion at the surface, would be transported to the back of the lattice. This could lead to several changes. For example, induced positive charge might no longer be observed on the surface. Furthermore, the potassium ions trapped in between four copper atoms of the front layer, would probably also induce a negative charge at the nearby copper atom in the second layer. This could lead to an increase in the adsorption time, and a further reduction in the dynamics of potassium at the surface.

(x) The coupled QM-MD approach was shown to be a valid alternative to constant potential based methods for simulating an electrode-electrolyte interface. While the ideal conductor assumption, which underlies constant potential methods, might hold well inside a bulk copper electrode, it might not for other materials or structures. For example, Elliott et al.[14] showed that the ideal conductor assumption does not hold for graphene. It was shown recently that nanoscale copper electrocatalysts generally outperform their bulk counter parts, for a combination of reasons, including a relatively increased binding area.[7] In future studies, it would be interesting to see to what extent the underlying atomic structure of a copper electrode influences the validity of the ideal conductor assumption, and whether it would hold, for example, for copper nanostructures.

4.5. Neural networks

In this section the results of the attempt to use neural networks to accelerate the coupling between the SCC-DFTB level and classical MD are presented and discussed.

4.5.1. Input data used

In section 4.4.2 it was found that in the QM-MD simulation the movement of potassium ions adsorbed on the cathode was severely limited, as compared to their movement in the C-MD simulation. By visualising the system during the simulation it was seen that the potassium ions could be adsorbed for times longer than nanoseconds. This may have a detrimental effect on the performance of the neural networks since the training set generally needs to be very diverse to generalise properly to unseen data.

Anticipating this shortcoming of primary QM-MD data, six extra QM-MD simulations of 4 ns were performed. Three of these simulations were performed by simulating the system with downscaled LJ interaction between the ions and the electrodes and neutral charges on both the ions and the electrodes. in this way, the adsorbed ions were pushed into the bulk so that when the LJ interaction and the charges were set to normal again, the system would form a new configuration on the cathode. The other three additional simulations were performed while the primary QM-MD simulation was still in process, such that the corresponding data was not yet available. These three QM-MD simulations were initialised from the previously performed C-MD simulation, from three different time marks within the final 10 ns of that simulation.

The training set consisted of a total of 12750 individual training inputs and targets, of which 6750 were obtained during the first 27 ns of the primary QM-MD simulation, and 6000 during one of the additional QM-MD simulations performed. Following the approach of Di Pasquale et al.[34], the test set consisted of the 750 inputs and targets that were obtained during the final 3 ns of the QM-MD simulation.

The test data was obtained in the same simulation that was used to obtain training data. Since it was previously observed that the change in the position of adsorbed potassium ions could be very moderate, the choice of test data may have positively influenced the obtained test error, so that the test error may not be representative of the true error.

For generating the three-dimensional input matrix for the CNNs, a finer binning was used for z-coordinates close to the cathode than further away, as described in section 3.4.2. To decide what range of z-values should be taken into account, experiments were performed to study the influence of neglecting ions with z-values larger than some cut-off in SCC-DFTB. It was decided to place the cut-off around 2 nm from the cathode, as it was seen that ions after this cut-off did not have a significant effect on the polarisation anymore.

The binning in the z-dimension was chosen to start from the z-position of the ion (or the hydrogen atom if water was taken into account) measured to be the closest to the cathode in all obtained data. The first ten bins were chosen to have a spatial extend of 3.6149/40 Å, and included the majority of adsorbed ions. The next forty bins were chosen to have a spatial extend of 3.6149/20 Å, and the last 46 bins were chosen to have a spatial extend of 3.6149/10 Å. In the x- and y- dimensions, the binning was the same over the entire domain, and was chosen to be 3.6149/20 Å. This way, the input space consisted of 160 bins in the x- and y-dimensions, and 96 bins in the z-dimension.

4.5.2. Results and discussion of selected networks

As described in the method, many variants of the proposed networks were attempted, to find the most optimal architecture. For example, deeper, shallower, wider and narrower networks were attempted, and the used loss function was varied by scaling one of the two terms. Below, only the results of three different networks are presented and discussed. Each of these used the loss function given in eq. 3.4. The results of all other variant are mostly briefly mentioned here, as they all performed less well than the networks presented below. The constrained error, defined as the difference between the sum of

the predicted charges and the charge assigned to the cathode, was below $1 \cdot 10^{-2}$ for all results presented. It therefore did not significantly contribute to the total error. Not constraining the loss yielded a straying from the desired net predicted charge, as was also seen in the work of Di Pasquale et al.[34], and therefore is not recommended.

The best performing FCN, by a small margin, was the network described in section 3.4.1, only this network took into account the charge values of the ions in the input, in addition to the Cartesian coordinates of the ions. That this yielded improved performance was surprising, as the charges of the ions are constant and the networks should in principle be able to deduce these themselves. It could be that the network simply uses these values as bias terms, that are not related anymore to the original charge values. Using this network, a minimum mean absolute test error was obtained of 0.0378 e. Figure 4.24 gives the histogram of the predicted charge distribution of this network on the test set. Both the SCC-DFTB calculations and the FCN predict two peaks in the histogram, but the FCN histogram is, incorrectly, much more concentrated around its peaks. The FCN was not able to capture the full variance in the SCC-DFTB results, and consequently predicts fewer very negative and fewer positive charges.
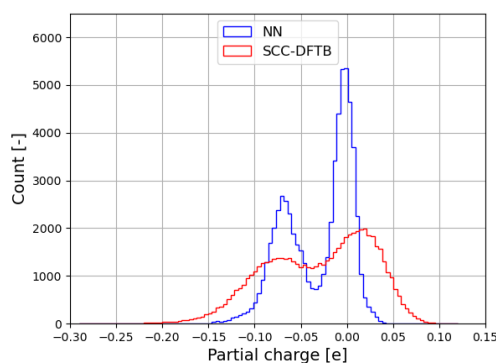
Figure 4.24: Histogram of the charge distribution as predicted by the fully connected network. Both the SCC-DFTB calculations and the FCN predict two peaks in the histogram, but the FCN incorrectly concentrates its predictions around these peaks.

Using the described CNN approach, an improvement in the performance was obtained, as compared to the FCN. The CNN presented here differed from the CNN described in the method only in the usage of leaky-ReLU activation functions (positive slope for input < 0). These yielded marginally improved performance over standard ReLU activation functions. The histograms of the predicted charge distribution on the test set of this network can be found in figure 4.25. Using this network, a minimum mean absolute test error was obtained of 0.0275 e, which represents the best performance of a network that was obtained on the test set in this thesis.

Like the FCN, the presented CNN overestimates the number of charges with values around 0 e, primarily at the expense of predicting more positive charges. A relative deviation of the NN results from the SCC-DFTB results is observed for both the most negative values and the most positive values, as was also observed in the work of Di Pasquale et al.[34]. Qualitatively, the charge distribution predicted by the CNN matches the SCC-DFTB result notably more than the FCN distribution did. The observed performance of the network indicates that it is possible to use a (convolutional) neural networks to replicate SCC-DFTB charge induction predictions on a metallic copper electrode, even if a completely satisfactory quantitative correspondence could not be obtained in this theses.

By investigating the average error per atom in the test set, it was found that the average error on the boundary atoms was not significantly larger than that for atoms more central in the electrode. This shows that the method of using neural networks to replicate SCC-DFTB charge induction predictions is not limited to periodic surfaces for which the method was shown by Di Pasquale et al.[34], but can also be used in the context of periodic surfaces. This result is exciting, as the periodic problem presented
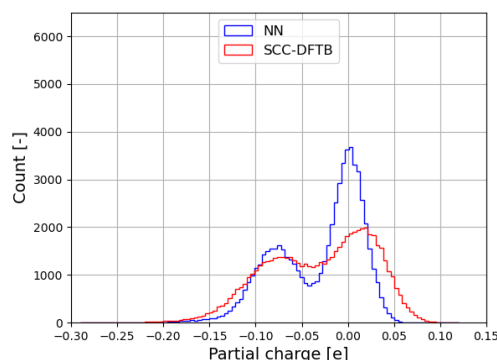
Figure 4.25: Histogram of the charge distribution as predicted by the convolutional neural network using leaky-ReLU activation functions. The CNN can be seen to overestimate the number of charges with values around 0 e. The performance of the CNN, however, is better than that of the FCN and indicates that neural networks can be used for learning problems like these.

here, with a significantly more non-local and significantly larger charge redistribution, and with copper being a d-type element with 11 electrons in its valence shell, was a lot more complex than the problem of a non-periodic graphene sheet.

The six additional QM-MD simulations performed to obtain more training data may not have provided representative training data. The simulated systems were not in a proper equilibrium after either resetting the LJ interaction and re-instantiating the charge values, or after restarting from the C-MD simulation. In addition, in the first 10 ns of the primary QM-MD simulation system was also not in equilibrium, even though the data from this period constitutes a third of the total data from the primary QM-MD simulation. Therefore, the amount of representative data of an equilibrated system used to train the networks was severely limited. This, in addition to the observed relatively moderate changes in the positions of adsorbed potassium ions on the cathode, suggests that the training set was one of the main limiting factors in the performance of the networks. For future studies, it is recommended to cleverly choose training data, suitable for the problem in question, rather than focus primarily on the quantity.

Unfortunately, no improvement on the presented results could be obtained by alterations of the CNN. The failure to improve on the presented results is not necessarily a consequence of the limitations of the application of neural networks in general in this context. Instead, it is primarily believed to be a consequence of the limitations of the used architectures in this thesis and the delicacy of the training procedure. A limited number of architectures could be attempted in this thesis, due to limited time and computational resources. Below, the results of a second attempted CNN setup are presented. Using this network as an example, the limitations of the used architectures in this thesis and the delicacy of the training procedure of the current approach are discussed.

The second CNN differed from the CNN setup described in the method in two ways. Firstly, it used pooling layers in two different locations. The second convolutional layer was replaced by a max pooling layer and an adaptive average pooling layer was applied before the final fully connected layer, as in a standard ResNet. These alterations were not thought to be able to yield improved performance over the previously described network, as the first pooling layer destroys spatial information and the inclusion of the adaptive pooling layer had already been observed to yield worse performance when testing other CNNs. Secondly, all water molecules in the system were included in the input of this CNN, instead of only the ions. In all similar networks, in which water was not included, the obtained minimum mean absolute test error was always over 0.037 e. However, by including water, a minimum mean absolute test error was obtained of 0.0308 e. This result seemingly indicated that the inclusion of the water molecules in the input can help the network with solving the problem.

After finding this result, it was first verified that replacing the second convolutional layer in the standard CNN by a pooling layer did not significantly affect the performance of the network. This result proves that the CNNs used in this thesis could not fully utilise the spatial information provided. This problem
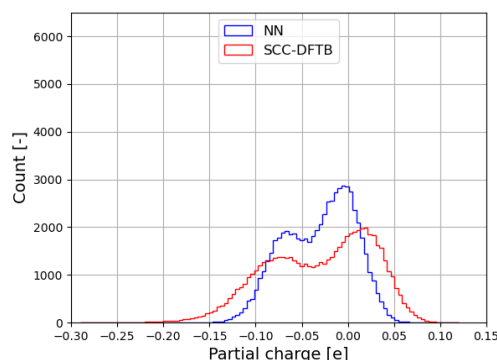
Figure 4.26: Histogram of the charge distribution as predicted by the described convolutional neural network using the water molecules in the system in the input in addition to using two pooling layers.

was anticipated and was precisely the reason why the inclusion of a coordinate convolutional layer was proposed in the method. In contrast, CNNs including a coordinate convolutional layer at the beginning of the networks performed worse than networks without a coordinate convolution. All CNNs in which a coordinate convolution was applied failed to train. They displayed a mean absolute test error of approximately 0.047 e, which they obtained already after one single epoch of training.

This kind of counter-intuitive behaviour was observed for two more extensions used for training the CNNs. Firstly, water molecules were included in the input space of the standard CNN network (and that using leaky-ReLU activation functions). This was proven be beneficial in the case of the previously presented network. For these networks, however, the inclusion of the water molecules led to the network failing to train and reverting back to a mean absolute test error of approximately 0.047 e. Secondly, in order to obtain more training data, certain symmetries of the modelled cathode were exploited. The single layered cathode depicted in figure 4.8 is the same one used in the QM-MD simulation. In this figure, three symmetries can be noticed: the electrode could be mirrored around the lines x=y and x=-y, and the system could be rotated by 180°, and the same electrode configuration would be obtained. The newly obtained system configuration would be considered unique by the networks. This way, three times as much training data was obtained. Counter-intuitively, the inclusion of the extra data did not lead to an improvement of the test error. Rather, the inclusion of the extra training data resulted in the networks displaying the same mean absolute test error of approximately 0.047 e after the first epoch and subsequently failing to train.

The failure of the networks to improve after the inclusion of these extensions, is not thought to be a consequence of mistakes that were made in the implementation of the extension. For the coordinate convolutional layer, two different implementations were attempted, both in PyTorch[119] and in Keras[120], but either one did not change the result. Furthermore, for the extra data sets that were obtained by mirroring and rotating, the test error that could be obtained using the FCN did ultimately revert back to the test error that was obtained using the original data set, making it unlikely that mistakes were made in the mirroring or rotating of the data set. However, it was observed that it took the FCN significantly more epochs to get to this test error, and it happened suddenly by a drop in the test error. This indicates that using the larger data set, containing more variance than before, led to significant changes in the training procedure. Beforehand, it was already seen that the training of the CNNs was very sensitive to the used optimiser and learning rate. For example, a too high learning rate led to some networks not being able to learn, even if the learning rate later decreased. Therefore, a delicate learning procedure is thought to be one of the two main reasons for the failure of the networks to improve after inclusion of the before-mentioned extensions.

The second main reason is that the chosen architectures of the CNNs were not suitable for capturing the complexity of the given problem. The networks stopped learning after increasing the complexity in the input space (coordinate convolution, water molecules or extra data), which could correspond to the

networks predicting values around the mean, and failing to learn the variance in the problem due to lacking complexity in their modelling. Although alterations to the CNNs were made, these always came down to adding a single layer, or removing one, but never a complete restructuring of the network. In potential follow-up studies, it is recommended to start with a small network and small data set, and work upwards. This way the capabilities of the network can be much better understood.

A NN/MD simulation was performed using the trained leaky-ReLU network. The cumulative distribution functions of potassium and chloride can be found in figure 4.27. The figure shows increased adsorption of potassium ions at the cathode, like the proper QM-MD simulation, and verifies that the neural network has learned how to replicate the predictions of SCC-DFTB calculations, up to a certain level. However, an increased concentration of ions was observed in the bulk region behind the double layer. It is unclear where this increase originated from.

(a) Potassium	(b) Chloride
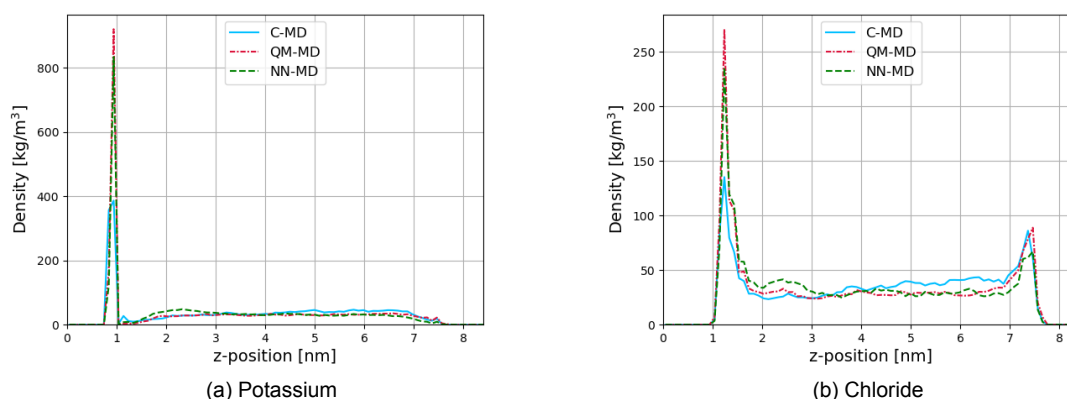
Figure 4.27: Density plot of the potassium (a, left) and chloride (b, right), plotted for the C-MD simulation (solid line), the QM-MD simulation (dash-dotted line) and the NN-MD simulation (dashed line).

In the work of Di Pasquale[34], the target charge values were quantised and binned. Consequently, the used FCN did not predict the exact charge on an atom in the electrode, but the corresponding bin. By following this approach, no significant improvement of the results of the FCN used in this thesis was found. Like the FCN, all CNNs overestimate the number of charges with a value around 0 e, and consequently underestimate of the number of the most negative and most positive values occuring in the SCC-DFTB calculations. Therefore, it would be interesting to see if this approach could be beneficial in training the CNNs for forcing the CNNs from predicting 0 e, which could currently function as a trivial solution, as for these networks this was not yet attempted.

While the periodicity of the problem did not explicitly show itself to be a problem, it contributed to making the problem harder. An idea to help the network with taking into acount, is using circular padding in the x,y-dimensions. In circular padding, the values from the other side of the input are padded. In PyTorch, different padding in different dimensions is not supported, and it was not implementation was written for this thesis.

4.5.3. Different approaches

In this section, three different approaches in using neural networks in the context of the presented problem are introduced.

Firstly, the described FCNs and CNNs could be replaced by a transformer. The transformer is a type of neural network that has long represented the state-of-the-art in natural language processing[121] and, since last year, in many vision tasks as well, by means of 'vision' transformers[122]. A transformer is not affected by many of the discussed disadvantages of both the FCN and the CNN. Since they apply the

mechanism of self-attention, in which basically all input is compared with all other input, transformers are permutation invariant. Furthermore, in the context of a vision transformer, positional embeddings of spatial information in the system can be used to directly incorporated.

There are two ways in which transformers could be applied in the context of the electrode polarisation problem. Firstly, a natural language processing approach could be followed, in which all ions (and possibly water molecules and copper atoms) are represented by one-dimensional arrays containing their positions and their charges. Secondly, a vision transformer approach could be followed, in which the previously described three-dimensional input space, used for training the CNNs, is divided up into patches in which extra positional data is encoded.

Secondly, the presented neural networks could be trained on smaller patches of an electrode, in order to simplify the problem, and allow for a prediction of the partial charges on electrodes of arbitrary size. By summing the partial charges, as computed in the QM-MD simulation, for regions of different sizes, it was found that the charge density on arbitrary patches comprising 56.25% of the copper electrode, could differ by up to 10%. This is attributed to the non-local nature of the charge redistribution on the metallic copper. However, similar calculations in which the system was replaced by that used in the work of Elliott et al.[14], showed that a patch comprising 25% of a graphene electrode consisting of 336 carbon atoms is already large enough to bring the local charge density within 1% of that of the charge density assigned to the entire electrode.

The observed local constraint on the charge density allows for graphene-like electrodes to be used in a radically different approach using the neural networks. The idea is to train a network to -only- predict the charge on such individual patches. As input, only ions in some specific radius away from the patch and the first few carbon atoms surrounding the patch could be used as these play the most important role in determining the local polarisation. By training on such smaller patches, the training procedure could be simplified, as predicting for such a smaller system is easier. Furthermore, much more data could be obtained, as many such patches are found in the system. Subsequently, by performing a convolution of such a trained network over the entirety of an electrode until an equilibrium in the charge distribution has been reached, this network could be used for predicting the charge on a periodic electrode of arbitrary size in an iterative way.

Thirdly and finally, neural networks could be used to speed-up the SCC-DFTB calculations, rather than serve as a complete replacement of the SCC-DFTB calculations in the coupled QM-MD workflow. The neural networks used in this thesis, as well as the fully connected network used in the work of Di Pasquale et al.[34], were shown to be unable to provide perfect descriptions for the problems given to them, after training on a set of SCC-DFTB calculations. Therefore, if the constraints on the accuracy of the predictions are very strict, these networks might not be able to provide satisfactory performance.

The SCC convergence procedure is what takes the most time when performing SCC-DFTB. As discussed in section 2.3, this procedure is to ensure that the computed orbital occupation and the Hamiltonian used to compute those occupations are consistent. It relies on a first guess for the orbital occupation, that usually is taken to be an even distribution of the charge of the system. In the QM-MD loop, this first guess coincided with the result of the previous computation, of 4 ps earlier in the simulation. It is hoped that this allows for fewer SCC iterations, thus reducing simulation time. In the same way, the predictions made by the trained neural networks could be used as initial guess for the new SCC-DFTB calculation. Ideally, this would allow for a significant decrease in the SCC-DFTB computation time, while keeping the same accuracy as from a full SCC-DFTB calculation.

Unfortunately, no evidence could be obtained for the supposed benefit of this method in this thesis. Here, it was found that using the trained leaky-ReLU network as suggested above did not lead to a decrease in the computation time, relative to initialising the SCC convergence procedure from the result of the previous SCC-DFTB calculation (up to 20% vs. up to 30% speed-up relative to no initial guess, respectively).

5

Conclusion

In this thesis, a proof of concept was established for using a novel coupled QM-MD approach to model metallic electrode-electrolyte interfaces, using copper specifically. SCC-DFTB calculations of the instantaneous electronic structure of a copper electrode were coupled to a classical MD simulation simulation of the electrode-electrolyte interface. This allowed the electrolyte configuration to evolve in response to changes in the electronic structure of the copper electrode due to charge induction effects, brought about by nearby ionic charges. The applied QM-MD method was described rigorously, and was used to investigate the compound distribution and dynamics at the copper electrode-electrolyte interface, relative to a classical MD simulation.

First, DFT calculations were used to benchmark the SCC-DFTB calculations. These results showed that for a single layer, SCC-DFTB is able, to some extent, to capture the polarisation of the sheet as predicted by DFT. Furthermore, these results demonstrated a sizable and very non-local charge redistribution as a result of polarisation by nearby charges. In the comparison between DFT and SCC-DFTB, it was found that experiments with multiple point charges and different sheet sizes, as well as a comparison of the partial charges next to one of the excess charges, are recommended.

The electrode-electrolyte system consisted of two copper electrodes, 3 nm wide, at both sides of a 1 M potassium chloride solution. During the QM-MD simulation, a significant charge redistribution from the mean charge was perceived, brought about by the presence of nearby charges in the solution. Density profiles showed a redistribution of ions in the system, with over 40% more potassium ions adsorbed in the QM-MD simulation than in a classical MD simulation. The increased attraction between potassium and the cathode after polarisation indicated that electrostatic effects played a significant role in the ion adsorption process. Furthermore, taking into account surface polarisability of the cathode surface was observed to significantly immobilise cations adsorbed on the cathode. Lastly, the preferential angular orientation of the water dipoles was found to be away from the cathode, as a result of the influence of adsorbed cations. This preference was increased in the case of the QM-MD simulation, due to local polarisation of the electrode and increased adsorption of cations.

Altogether, polarisation effects brought about significant changes in the compound distribution and dynamics at the copper electrode-electrolyte interface, relative to a classical MD simulation. For future similar QM-MD simulations, it is recommended to more closely investigate the effects of the time interval between SCC-DFTB calculations and charge transfer between the electrolyte and the electrode.

The secondary goal of this project was to explore to what extent are neural networks able to replace periodic SCC-DFTB calculations of the electronic structure of a copper electrode in the coupled QM-MD approach.

A vision approach to the problem was found capable of yielding improved results over a fully connected network approach using Cartesian coordinates as input. Qualitatively, the observed performance of the used networks indicated that it is possible to use (convolutional) neural networks to replicate SCC-DFTB charge induction predictions on a metallic electrode, even if a completely satisfactory quantitative correspondence could not be obtained in this thesis. In addition, it was found that the applicability of neural networks was not limited to non-periodic surfaces, but can also be used in the context of periodic surfaces.

Evidence was provided that neural networks can be used to replicate SCC-DFTB predictions, even for periodic and metallic surfaces. However, for future applications of neural networks in this context, it is recommended to use a transformer approach to the problem. Furthermore, different network architectures, more suitably chosen the training data increased attention for the training procedure may improve the predictions.

Bibliography

[1]   H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma. "The missing term in effective pair potentials". In: *The Journal of Physical Chemistry* 91.24 (1987), pp. 6269–6271. DOI: `10.1021/j100308a038`.

[2]   William Humphrey, Andrew Dalke, and Klaus Schulten. "VMD – Visual Molecular Dynamics". In: *Journal of Molecular Graphics* 14 (1996), pp. 33–38.

[3]   V. Masson-Delmotte et al. *Climate Change 2021: The Physical Science Basis*. IPCC, 2021. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA., 2021. DOI: `10.1017/9781009157896`.

[4]   David Raciti and Chao Wang. "Recent Advances in CO2 Reduction Electrocatalysis on Copper". In: *ACS Energy Letters* 3.7 (2018), pp. 1545–1556. DOI: `10.1021/acsenergylett.8b00553`.

[5]   Phil De Luna et al. "What would it take for renewably powered electrosynthesis to displace petrochemical processes?" In: *Science* 364.6438 (2019), eaav3506. DOI: `10.1126/science.aav3506`.

[6]   Alejandro J. Garza, Alexis T. Bell, and Martin Head-Gordon. "Mechanism of CO2 Reduction at Copper Surfaces: Pathways to C2 Products". In: *ACS Catalysis* 8.2 (2018), pp. 1490–1499. DOI: `10.1021/acscatal.7b03477`.

[7]   David Raciti and Chao Wang. "Recent Advances in CO2 Reduction Electrocatalysis on Copper". In: *ACS Energy Letters* 3.7 (2018), pp. 1545–1556. DOI: `10.1021/acsenergylett.8b00553`.

[8]   Stephanie Nitopi et al. "Progress and Perspectives of Electrochemical CO2 Reduction on Copper in Aqueous Electrolyte". In: *Chemical Reviews* 119.12 (2019). PMID: 31117420, pp. 7610–7672. DOI: `10.1021/acs.chemrev.8b00705`.

[9]   Marilia Moura de Salles Pupo and Ruud Kortlever. "Electrolyte effects on the electrochemical reduction of CO2". English. In: *ChemPhysChem: a European journal of chemical physics and physical chemistry* 20.22 (2019), pp. 2926–2935. ISSN: 1439-4235. DOI: `10.1002/cphc.201900680`.

[10] Leanne D. Chen et al. "Electric Field Effects in Electrochemical CO2 Reduction". In: *ACS Catalysis* 6.10 (2016), pp. 7133–7139. DOI: `10.1021/acscatal.6b02299`.

[11] H. Helmholtz. "Ueber einige Gesetze der Vertheilung elektrischer Ströme in körperlichen Leitern mit Anwendung auf die thierisch-elektrischen Versuche". In: *Annalen der Physik* 165.6 (1853), pp. 211–233. DOI: `https://doi.org/10.1002/andp.18531650603`.

[12] Dimiter N Petsev, Frank van Swol, and Laura J D Frink. *Molecular Theory of Electric Double Layers*. 2053-2563. IOP Publishing, 2021. ISBN: 978-0-7503-2276-8. DOI: `10.1088/978-0-7503-2276-8`.

[13] Guilherme Colherinhas, Eudes Eterno Fileti, and Vitaly V. Chaban. "The Band Gap of Graphene Is Efficiently Tuned by Monovalent Ions". In: *The Journal of Physical Chemistry Letters* 6.2 (2015). PMID: 26263467, pp. 302–307. DOI: `10.1021/jz502601z`.

[14] Joshua D. Elliott, Alessandro Troisi, and Paola Carbone. "A QM/MD Coupling Method to Model the Ion-Induced Polarization of Graphene". In: *Journal of Chemical Theory and Computation* 16.8 (2020). PMID: 32644791, pp. 5253–5263. DOI: `10.1021/acs.jctc.0c00239`.

[15] Heather J. Kulik, Eric Schwegler, and Giulia Galli. "Probing the Structure of Salt Water under Confinement with First-Principles Molecular Dynamics and Theoretical X-ray Absorption Spectroscopy". In: *The Journal of Physical Chemistry Letters* 3.18 (2012). PMID: 26295887, pp. 2653–2658. DOI: `10.1021/jz300932p`.

[16] Tuan Anh Pham et al. "Salt Solutions in Carbon Nanotubes: The Role of Cation−π Interactions". In: *The Journal of Physical Chemistry C* 120.13 (2016), pp. 7332–7338. DOI: 10.1021/acs.jpcc.5b12245.

[17] Suji Gim, Hyung-Kyu Lim, and Hyungjun Kim. "Multiscale Simulation Method for Quantitative Prediction of Surface Wettability at the Atomistic Level". In: *The Journal of Physical Chemistry Letters* 9.7 (2018). PMID: 29558139, pp. 1750–1758. DOI: 10.1021/acs.jpclett.8b00466.

[18] S. Kondrat et al. "A superionic state in nano-porous double-layer capacitors: insights from Monte Carlo simulations". In: *Phys. Chem. Chem. Phys.* 13 (23 2011), pp. 11359–11366. DOI: 10.1039/C1CP20798A.

[19] S Kondrat and A Kornyshev. "Corrigendum: Superionic state in double-layer capacitors with nanoporous electrodes". In: *Journal of Physics: Condensed Matter* 25.11 (Feb. 2013), p. 119501. DOI: 10.1088/0953-8984/25/11/119501.

[20] Céline Merlet et al. "On the molecular origin of supercapacitance in nanoporous carbon electrodes". In: *Nature Materials* 11.4 (Apr. 2012). DOI: 10.1038/nmat3260.

[21] VS Sandeep Inakollu et al. "Polarisable force fields: what do they add in biomolecular simulations?" In: *Current Opinion in Structural Biology* 61 (2020). Theory and Simulation ● Macromolecular Assemblies, pp. 182–190. ISSN: 0959-440X. DOI: https://doi.org/10.1016/j.sbi.2019.12.012.

[22] Céline Merlet et al. "Computer simulations of ionic liquids at electrochemical interfaces". In: *Phys. Chem. Chem. Phys.* 15 (38 2013), pp. 15781–15792. DOI: 10.1039/C3CP52088A.

[23] Laura Scalfi et al. "Charge fluctuations from molecular simulations in the constant-potential ensemble". In: *Phys. Chem. Chem. Phys.* 22 (19 2020), pp. 10480–10489. DOI: 10.1039/C9CP06285H.

[24] Dorothea Golze et al. "Simulation of Adsorption Processes at Metallic Interfaces: An Image Charge Augmented QM/MM Approach". In: *Journal of Chemical Theory and Computation* 9.11 (2013). PMID: 26583423, pp. 5086–5097. DOI: 10.1021/ct400698y.

[25] Thomas Dufils et al. "Simulating Electrochemical Systems by Combining the Finite Field Method with a Constant Potential Electrode". In: *Phys. Rev. Lett.* 123 (19 Nov. 2019), p. 195501. DOI: 10.1103/PhysRevLett.123.195501.

[26] Matt K. Petersen et al. "A Computationally Efficient Treatment of Polarizable Electrochemical Cells Held at a Constant Potential". In: *The Journal of Physical Chemistry C* 116.7 (2012), pp. 4903–4912. DOI: 10.1021/jp210252g.

[27] J. Ilja Siepmann and Michiel Sprik. "Influence of surface topology and electrostatic potential on water/electrode systems". In: *The Journal of Chemical Physics* 102.1 (1995), pp. 511–524. DOI: 10.1063/1.469429.

[28] Stewart K. Reed, Oliver J. Lanning, and Paul A. Madden. "Electrochemical interface between an ionic liquid and a model metallic electrode". In: *The Journal of Chemical Physics* 126.8 (2007), p. 084704. DOI: 10.1063/1.2464084.

[29] Arjan van der Vaart et al. "Are Many-Body Effects Important in Protein Folding?" In: *The Journal of Physical Chemistry B* 104.40 (2000), pp. 9554–9563. DOI: 10.1021/jp001193f.

[30] Upendra Adhikari et al. "Computational Estimation of Microsecond to Second Atomistic Folding Times". In: *Journal of the American Chemical Society* 141.16 (2019), pp. 6519–6526. DOI: 10.1021/jacs.8b10735.

[31] Ke Xu et al. "Pulse Dynamics of Electric Double Layer Formation on All-Solid-State Graphene Field-Effect Transistors". In: *ACS Applied Materials & Interfaces* 10.49 (2018), pp. 43166–43176. DOI: 10.1021/acsami.8b13649.

[32] Elliot Schmidt et al. "Characterization of the Electric Double Layer Formation Dynamics of a Metal/Ionic Liquid/Metal Structure". In: *ACS Applied Materials & Interfaces* 8.23 (2016). PMID: 27213215, pp. 14879–14884. DOI: 10.1021/acsami.6b04065.

[33] Sarah Hocine et al. "How Ion Condensation Occurs at a Charged Surface: A Molecular Dynamics Investigation of the Stern Layer for Water-Silica Interfaces". In: *The Journal of Physical Chemistry C* 120.2 (2016), pp. 963–973. DOI: 10.1021/acs.jpcc.5b08836.

[34] Nicodemo Di Pasquale et al. "Dynamically Polarizable Force Fields for Surface Simulations via Multi-output Classification Neural Networks". In: *Journal of Chemical Theory and Computation* 17.7 (2021). PMID: 34197102, pp. 4477–4485. DOI: 10.1021/acs.jctc.1c00360.

[35] H. A. Lorentz. "Ueber die Anwendung des Satzes vom Virial in der kinetischen Theorie der Gase". In: *Annalen der Physik* 248.1 (1881), pp. 127–136. DOI: https://doi.org/10.1002/andp.18812480110.

[36] Loup Verlet. "Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules". In: *Phys. Rev.* 159 (1 July 1967), pp. 98–103. DOI: 10.1103/PhysRev.159.98.

[37] J. M. Sanz-Serna. "Symplectic integrators for Hamiltonian problems: an overview". In: *Acta Numerica* 1 (1992), pp. 243–286. DOI: 10.1017/S0962492900002282.

[38] Robert G. Parr and Yang Yang Weitao. "Appendix A, Functionals". *Density-Functional Theory of Atoms and Molecules*. Oxford University Press, Incorporated, Oxford, New York, 1994. ISBN: 9780195357738.

[39] M. Born and R. Oppenheimer. "Zur Quantentheorie der Molekeln". In: *Annalen der Physik* 389.20 (1927), pp. 457–484. DOI: https://doi.org/10.1002/andp.19273892002.

[40] Jörg Meyer. "Ab initio Modeling of Energy Dissipation during Chemical Reactions at Transition Metal Surfaces". In: (2012).

[41] N. M. Harrison. *An Introduction to Density Functional Theory*. Imperial College of Science, Technology, and Medicine, 2002.

[42] Chang Liu. "Le Fantôme de l'Opéra - Studies on Atoms and Electrons Beneath". In: *Doctoral Thesis in Theoretical Physics at Stockholm University, Sweden* (2019). DOI: ISBN978-91-7797-863-3ISBN978-91-7797-864-0. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:su:diva-174700.

[43] P. Hohenberg and W. Kohn. "Inhomogeneous Electron Gas". In: *Phys. Rev.* 136 (3B Nov. 1964), B864–B871. DOI: 10.1103/PhysRev.136.B864.

[44] W. Kohn and L. J. Sham. "Self-Consistent Equations Including Exchange and Correlation Effects". In: *Phys. Rev.* 140 (4A Nov. 1965), A1133–A1138. DOI: 10.1103/PhysRev.140.A1133.

[45] F.M. Bickelhaupt and E.J. Baerends. "Kohn-Sham Density Functional Theory: Predicting and Understanding Chemistry". English. In: *Reviews in computational chemistry* 15 (2000), pp. 1–86. ISSN: 1069-3599. DOI: 10.1002/9780470125922.ch1.

[46] Ralf Stowasser and Roald Hoffmann. "What Do the Kohn−Sham Orbitals and Eigenvalues Mean?" In: *Journal of the American Chemical Society* 121.14 (1999), pp. 3414–3420. DOI: 10.1021/ja9826892.

[47] "Introduction to density-functional theory". Ed. by Julien Toulouse. Paris: Laboratoire de Chimie Th´eorique, Sorbonne Université and CNRS, 2019. URL: https://www.lct.jussieu.fr/pagesperso/toulouse/enseignement/introduction_dft.pdf.

[48] John P. Perdew et al. "Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation". In: *Phys. Rev. B* 46 (11 Sept. 1992), pp. 6671–6687. DOI: 10.1103/PhysRevB.46.6671.

[49] A. D. Becke. "Density-functional exchange-energy approximation with correct asymptotic behavior". In: *Phys. Rev. A* 38 (6 Sept. 1988), pp. 3098–3100. DOI: 10.1103/PhysRevA.38.3098.

[50] P.G. Jambrina and J. Aldegunde. "Chapter 20 - Computational Tools for the Study of Biomolecules". *Tools For Chemical Product Design*. Ed. by Mariano Martín, Mario R. Eden, and Nishanth G. Chemmangattuvalappil. Vol. 39. Computer Aided Chemical Engineering. Elsevier, 2016, pp. 583–648. DOI: https://doi.org/10.1016/B978-0-444-63683-6.00020-4.

[51] Pekka Koskinen and Ville Mäkinen. "Density-functional tight-binding for beginners". In: *Computational Materials Science* 47.1 (Nov. 2009), pp. 237–253. ISSN: 0927-0256. DOI: 10.1016/j.commatsci.2009.07.013.

[52] M. Elstner et al. "Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties". In: *Phys. Rev. B* 58 (11 Sept. 1998), pp. 7260–7268. DOI: 10.1103/PhysRevB.58.7260.

[53] Mervyn Roy. "The Tight Binding Method". New Brunswick, New Jersey: Rutgers–New Brunswick, Department of Physics & Astronomy, 2015.

[54] Fernand Spiegelman et al. "Density-functional tight-binding: basic concepts and applications to molecules and clusters". In: *Advances in Physics: X* 5.1 (2020). PMID: 33154977, p. 1710252. DOI: 10.1080/23746149.2019.1710252.

[55] D. Porezag et al. "Construction of tight-binding-like potentials on the basis of density-functional theory: Application to carbon". In: *Phys. Rev. B* 51 (19 May 1995), pp. 12947–12957. DOI: 10.1103/PhysRevB.51.12947.

[56] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 978-0387-31073-2.

[57] Vinod Nair and Geoffrey E. Hinton. "Rectified Linear Units Improve Restricted Boltzmann Machines". In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. Haifa, Israel: Omnipress, 2010, pp. 807–814. ISBN: 9781605589077.

[58] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. Adaptive computation and machine learning. MIT Press, 2016. ISBN: 9780262035613.

[59] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2014. DOI: 10.48550/ARXIV.1412.6980.

[60] Sunitha Basodi et al. "Gradient amplification: An efficient way to train deep neural networks". In: *Big Data Mining and Analytics* 3.3 (2020), pp. 196–207. DOI: 10.26599/BDMA.2020.9020004.

[61] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/ARXIV.1512.03385.

[62] Wen Xu et al. "Advances in Convolutional Neural Networks". *Advances and Applications in Deep Learning*. Ed. by Marco Antonio Aceves-Fernandez. Rijeka: IntechOpen, 2020. Chap. 2. DOI: 10.5772/intechopen.93512.

[63] Rikiya Yamashita et al. "Convolutional neural networks: an overview and application in radiology". In: *Insights into Imaging* 9.4 (2018). DOI: 10.1007/s13244-018-0639-9.

[64] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. DOI: 10.48550/ARXIV.1502.03167.

[65] Steve Plimpton. "Fast Parallel Algorithms for Short-Range Molecular Dynamics". In: *Journal of Computational Physics* 117.1 (1995), pp. 1–19. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1995.1039. URL: lammps.org.

[66] M. E. Straumanis and L. S. Yu. "Lattice parameters, densities, expansion coefficients and perfection of structure of Cu and of Cu–In α phase". In: *Acta Crystallographica Section A* 25.6 (Nov. 1969), pp. 676–682. DOI: 10.1107/S0567739469001549.

[67] Divya Bohra et al. "Modeling the electrical double layer to understand the reaction environment in a CO2 electrocatalytic system". In: *Energy Environ. Sci.* 12 (11 2019), pp. 3380–3389. DOI: 10.1039/C9EE02485A.

[68] Hendrik Heinz et al. "Accurate Simulation of Surfaces and Interfaces of Face-Centered Cubic Metals Using 12−6 and 9−6 Lennard-Jones Potentials". In: *The Journal of Physical Chemistry C* 112.44 (2008), pp. 17281–17290. DOI: 10.1021/jp801931d.

[69] Jonathan G. Harris and Kwong H. Yung. "Carbon Dioxide's Liquid-Vapor Coexistence Curve And Critical Properties as Predicted by a Simple Molecular Model". In: *The Journal of Physical Chemistry* 99.31 (1995), pp. 12021–12024. DOI: 10.1021/j100031a034.

[70] Randall T. Cygan, Vyacheslav N. Romanov, and Evgeniy M. Myshakin. "Molecular Simulation of Carbon Dioxide Capture by Montmorillonite Using an Accurate and Flexible Force Field". In: *The Journal of Physical Chemistry C* 116.24 (2012), pp. 13079–13091. DOI: 10.1021/jp3007574.

[71] Alessandro Silvestri et al. "Wetting Properties of the CO2–Water–Calcite System via Molecular Simulations: Shape and Size Effects". In: *ChemRxiv* (2019). DOI: 10.26434/chemrxiv.9809834.v1.

[72] R. Demichelis et al. "Stable prenucleation mineral clusters are liquid-like ionic polymers". In: *Nat Commun* 2.590 (2011). DOI: 10.1038/ncomms1604.

[73] Takuma Yagasaki, Masakazu Matsumoto, and Hideki Tanaka. "Lennard-Jones Parameters Determined to Reproduce the Solubility of NaCl and KCl in SPC/E, TIP3P, and TIP4P/2005 Water". In: *Journal of Chemical Theory and Computation* 16.4 (2020). PMID: 32207974, pp. 2460–2473. DOI: 10.1021/acs.jctc.9b00941.

[74] In Suk Joung and Thomas E. Cheatham. "Determination of Alkali and Halide Monovalent Ion Parameters for Use in Explicitly Solvated Biomolecular Simulations". In: *The Journal of Physical Chemistry B* 112.30 (2008). PMID: 18593145, pp. 9020–9041. DOI: 10.1021/jp8001614.

[75] Leela S. Dodda et al. "LigParGen web server: an automatic OPLS-AA parameter generator for organic ligands". In: *Nucleic Acids Research* 45.W1 (Apr. 2017), W331–W336.

[76] William L. Jorgensen and Julian Tirado-Rives. "Potential energy functions for atomic-level simulations of water and organic and biomolecular systems". In: *Proceedings of the National Academy of Sciences* 102.19 (2005), pp. 6665–6670. DOI: 10.1073/pnas.0408037102.

[77] Leela S. Dodda et al. "1.14*CM1A-LBCC: Localized Bond-Charge Corrected CM1A Charges for Condensed-Phase Simulations". In: *The Journal of Physical Chemistry B* 121.15 (2017). PMID: 28224794, pp. 3864–3870. DOI: 10.1021/acs.jpcb.7b00272.

[78] Jean-Paul Ryckaert, Giovanni Ciccotti, and Herman J.C Berendsen. "Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes". In: *Journal of Computational Physics* 23.3 (1977), pp. 327–341. ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(77)90098-5.

[79] R.W. Hockney and J.W Eastwood. *Computer Simulation Using Particles (1st ed.)* Boca Raton, US: CRC Press, 1988. ISBN: 9780367806934.

[80] Daan Frenkel and Berend Smit. "Chapter 4 - Molecular Dynamics Simulations". *Understanding Molecular Simulation (Second Edition)*. Ed. by Daan Frenkel and Berend Smit. Second Edition. San Diego: Academic Press, 2002, pp. 63–107. ISBN: 978-0-12-267351-1. DOI: https://doi.org/10.1016/B978-012267351-1/50006-7.

[81] B. Hourahine, B. Aradi, and et al. "DFTB+, a software package for efficient approximate density functional theory based atomistic simulations". In: *The Journal of Chemical Physics* 152.12 (2020), p. 124101. DOI: 10.1063/1.5143190.

[82] Johannes Frenzel et al. "Structural and Electronic Properties of Bulk Gibbsite and Gibbsite Surfaces". In: *Zeitschrift für anorganische und allgemeine Chemie* 631.6-7 (2005), pp. 1267–1271. DOI: https://doi.org/10.1002/zaac.200500051.

[83] Luciana Guimarães et al. "Imogolite Nanotubes: Stability, Electronic, and Mechanical Properties". In: *ACS Nano* 1.4 (2007). PMID: 19206688, pp. 362–368. DOI: 10.1021/nn700184k.

[84] N. Jardillier Universite Montpellier II. "PhD Thesis". 2006. URL: online:http://nicolas.jardillier.free.fr.

[85] Chang Liu et al. "Stability and Effects of Subsurface Oxygen in Oxide-Derived Cu Catalyst for CO2 Reduction". In: *The Journal of Physical Chemistry C* 121.45 (2017), pp. 25010–25017. DOI: 10.1021/acs.jpcc.7b08269.

[86] Di Yin et al. "A Molecular Dynamics Study of Carbon Dimerization on Cu(111) Surface with Optimized DFTB Parameters". In: *Acta Physico-Chimica Sinica* 34.10, 1116 (2018), p. 1116. DOI: 10.3866/PKU.WHXB201801151.

[87] Hasan Kurban, Mustafa Kurban, and Mehmet Dalkılıç. "Density-functional tight-binding approach for the structural analysis and electronic structure of copper hydride metallic nanoparticles". In: *Materials Today Communications* 21 (2019), p. 100648. ISSN: 2352-4928. DOI: https://doi.org/10.1016/j.mtcomm.2019.100648.

[88] M. Elstner et al. "Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties". In: *Phys. Rev. B* 58 (11 Sept. 1998), pp. 7260–7268. DOI: 10.1103/PhysRevB.58.7260.

[89] G. te Velde et al. "Chemistry with ADF". In: *J. Comput. Chem.* 22.9 (2001), pp. 931–967. ISSN: 1096-987X. DOI: 10.1002/jcc.1056.

[90] SCM, Theoretical Chemistry, Vrije Universiteit, Amsterdam, The Netherlands. *ADF*. Version 2021.104. URL: http://www.scm.com.

[91] E. Van Lenthe and E. J. Baerends. "Optimized Slater-type basis sets for the elements 1–118". In: *Journal of Computational Chemistry* 24.9 (2003), pp. 1142–1156. DOI: https://doi.org/10.1002/jcc.10255.

[92] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. "Generalized Gradient Approximation Made Simple". In: *Phys. Rev. Lett.* 77 (18 Oct. 1996), pp. 3865–3868. DOI: 10.1103/PhysRevLett.77.3865.

[93] C.-L. Nies and M. Beilstein Nolan. "DFT calculations of the structure and stability of copper clusters on MoS2". In: *Beilstein J. Nanotechnol.* 11 (2020), pp. 391–406. DOI: 10.3762/bjnano.11.30.

[94] Zhi-Xin Hu, Haiping Lan, and Wei Ji. "Role of the dispersion force in modeling the interfacial properties of molecule-metal interfaces: adsorption of thiophene on copper surfaces". In: *Scientific Reports* 4.5036 (2014). DOI: 10.1038/srep05036.

[95] E. van Lenthe, E. J. Baerends, and J. G. Snijders. "Relativistic regular two‐component Hamiltonians". In: *The Journal of Chemical Physics* 99.6 (1993), pp. 4597–4610. DOI: 10.1063/1.466059.

[96] "SCF convergence guidelines for ADF". Software for Chemistry and Materials B.V. URL: https://www.scm.com/doc/ADF/Rec_problems_questions/SCF.html.

[97] Xiangqian Hu and Weitao Yang. "Accelerating self-consistent field convergence with the augmented Roothaan–Hall energy function". In: *The Journal of Chemical Physics* 132.5 (2010), p. 054109. DOI: 10.1063/1.3304922.

[98] Qiang Cui and Marcus Elstner. "Density functional tight binding: values of semi-empirical methods in an ab initio era". In: *Phys. Chem. Chem. Phys.* 16 (28 2014), pp. 14368–14377. DOI: 10.1039/C4CP00908H.

[99] Fernando J. A. L. Cruz and José P. B. Mota. "Conformational Thermodynamics of DNA Strands in Hydrophilic Nanopores". In: *The Journal of Physical Chemistry C* 120.36 (2016), pp. 20357–20367. DOI: 10.1021/acs.jpcc.6b06234.

[100] Python Software Foundation. *Python Language, version 3.9.* URL: http://www.python.org.

[101] Felix Bloch. "Über die Quantenmechanik der Elektronen in Kristallgittern". In: *Zeitschrift für Physik* 52 (7 July 1929). DOI: 10.1007/BF01339455.

[102] Hendrik J. Monkhorst and James D. Pack. "Special points for Brillouin-zone integrations". In: *Phys. Rev. B* 13 (12 June 1976), pp. 5188–5192. DOI: 10.1103/PhysRevB.13.5188.

[103] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.

[104] Benjamin Graham and Laurens van der Maaten. *Submanifold Sparse Convolutional Networks*. 2017. DOI: 10.48550/ARXIV.1706.01307.

[105] Ben Graham. *Sparse 3D convolutional neural networks*. 2015. DOI: 10.48550/ARXIV.1505.02890.

[106] Baoyuan Liu et al. "Sparse Convolutional Neural Networks". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, June 2015, pp. 806–814. DOI: 10.1109/CVPR.2015.7298681.

[107] Osman Semih Kayhan and Jan C. van Gemert. "On Translation Invariance in CNNs: Convolutional Layers can Exploit Absolute Spatial Location". In: *CoRR* abs/2003.07064 (2020).

[108] Md. Amirul Islam, Sen Jia, and Neil D. B. Bruce. "How Much Position Information Do Convolutional Neural Networks Encode?" In: *CoRR* abs/2001.08248 (2020).

[109] Rui Xu et al. "Positional Encoding as Spatial Inductive Bias in GANs". In: *CoRR* abs/2012.05217 (2020).

[110] Rosanne Liu et al. "An Intriguing Failing of Convolutional Neural Networks and the CoordConv Solution". In: *CoRR* abs/1807.03247 (2018).

[111] Fenna Westerbaan van der Meij. "Dynamic Surface Charge Distribution". In: *MSc thesis, Delft University of Technology* (2020).

[112] Darren Rowland. *Density of NaCl(aq)*. [Online; accessed 07-06-2022]. 2021. URL: https://advancedthermo.com/electrolytes/density_nacl_jun2021.html.

[113] MATLAB. *version R2019b*. Natick, Massachusetts: The MathWorks Inc., 2010.

[114] In Suk Joung and Thomas E. Cheatham. "Molecular Dynamics Simulations of the Dynamic and Energetic Properties of Alkali and Halide Ions Using Water-Model-Specific Ion Parameters". In: *The Journal of Physical Chemistry B* 113.40 (2009). PMID: 19757835, pp. 13279–13290. DOI: 10.1021/jp902584c.

[115] Cheng Zhan et al. "Specific ion effects at graphitic interfaces". In: *Nature Communications* 10.1 (2019). DOI: 10.1038/s41467-019-12854-7.

[116] Lang Li et al. "Unraveling molecular structures and ion effects of electric double layers at metal water interfaces". In: *Cell Reports Physical Science* 3.2 (2022), p. 100759. ISSN: 2666-3864. DOI: https://doi.org/10.1016/j.xcrp.2022.100759.

[117] R. S. Mulliken. "Electronic Population Analysis on LCAO☐MO Molecular Wave Functions. III. Effects of Hybridization on Overlap and Gross AO Populations". In: *The Journal of Chemical Physics* 23.12 (1955), pp. 2338–2342. DOI: 10.1063/1.1741876.

[118] Ira Levine. *Quantum Chemistry (4th ed.)* Englewood Cliffs, N.J.: Prentice Hall, 1991.

[119] walsvid. *CoordConv*. https://github.com/walsvid/CoordConv. 2018.

[120] titu1994. *keras-coordconv*. https://github.com/titu1994/keras-coordconv/commits/master. 2018.

[121] Thomas Wolf et al. "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

[122] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2020. DOI: 10.48550/ARXIV.2010.11929.

A
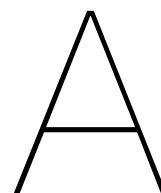
Choosing the k-mesh

Multiple periodic systems, as they would occur in the QM-MD simulation, were simulated in DFTB+. K-space sampling was performed using various different k-meshes. Uneven Monkhorst-Pack grids were used, ranging from 1x1x1 to 7x7x7, as well as even schemes, ranging from 2x2x2 to 8x8x8.

Ultimately, a Monkhorst-Pack grid of 2x2x2 was chosen. Compared with the most dense grid, the 8x8x8 grid, this yielded an average error in the measured total energy of the systems, of 0.3 meV per simulated atom in the system. The maximum measured error in the predicted charges between the 2x2x2 and 8x8x8 grids, was $2.3 \cdot 10^{-4}$ e. The average error between predicted charges, as computed using the two different grids, was $9 \cdot 10^{-5}$ e. Since the average magnitude of the charges in the simulated system was -0.3125 e, both the values for the maximum and the average error were found to be acceptable.

B

Additional results DFT

B.1. Three layers

Figure B.1 gives results of DFT and SCC-DFTB calculations on the three layered sheet of 216 atoms. Results are given in which the sheet was constructed using the original lattice parameter of 3.6149 Å are given, which have already been depicted earlier in figuer 4.5. Additionally, to two extra results, obtained using SCC-DFTB, are given: the result of the original problem but taking into account colinear spin polarisation using the spin constants by Yin et al.[86], and the result of a system constructed using the SCC-DFTB optimised lattice parameter of 3.71 Å. Clearly, these alterations did not significantly improve the SCC-DFTB result.
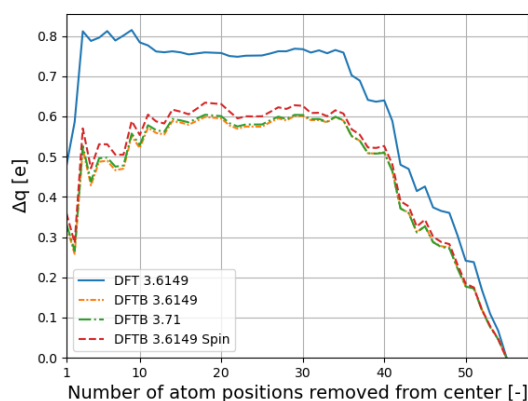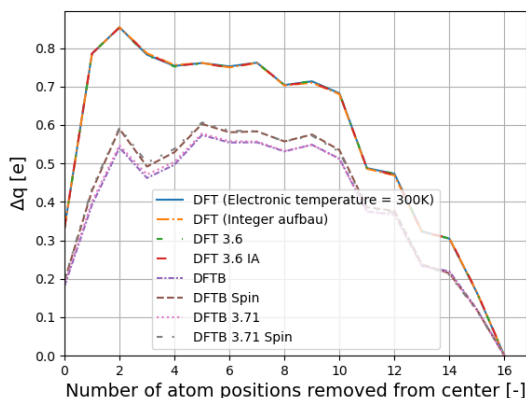

Figure B.1: Plot depicting the DFT and SCC-DFTB summed excess charge Δq as a function of the radial distance r from the center of a three layered sheet of (100) copper consisting of 216 atoms, which is polarised by a point charge (q = -1.0 e) at a height of 3 Å above the center of the sheet. The number in the legend gives the lattice constant used to construct the sheet. 'Spin' denotes the incorporation of a colinear treatment of spin. The latter can be seen to increase the polarisability of the sheet somewhat, but neither of the alterations is able to significantly improve the SCC-DFTB result.

To be able to perform more experiments on three layered sheets given time constraints, the system size was reduced to 122 atoms in order to decrease the simulation time. Figure B.2 gives the results for multiple experiments performed on the resulting system. It was checked whether specifying an electronic temperature of 300 K (which speeds up the computation) or the usage of the 'Integer aufbau' option (sometimes a warnign was received about partially occupied orbitals in the final result, this option could help solve that) had a significant impact on the result. Additionally, the lattice constant used to construct the system was changed to the AMS-ADF optimised value of 3.6 Å. Using SCC-DFTB calculations, the same experiments were performed as previously done for the larger system, given in figure B.1. As can be seen, none of the extra DFT options changed the final result significantly. The extra DFTB options were also not able to significantly improve the result, just like before.

Figure B.2: Plot depicting the DFT and SCC-DFTB summed excess charge Δq as a function of the radial distance r from the center of a three layered sheet of (100) copper consisting of 122 atoms, which is polarised by a point charge (q = -1.0 e) at heights of 3 Å above the center of the sheet. 'Spin' denotes the incorporation of a colinear treatment of spin, and '3.71' denotes the use of the SCC-DFTB optimised lattice constant to construct the sheet. As can be seen, none of the extra options significantly changed the final result.

B.2. Single layer

Like was done for the three layered sheet, of some extra options in both DFT and SCC-DFTB it was checked whether these changed the final results. The result of these calculation can be found in figure B.3. Like in the case of the three layered sheet too, none of the extra options significantly changed the final results.
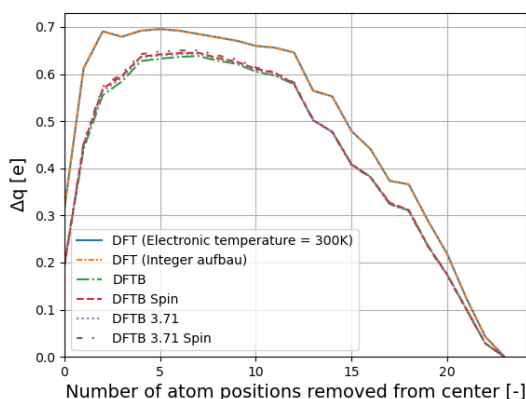

Figure B.3: Plot depicting the DFT (solid) and SCC-DFTB (dash-dotted) summed excess charge Δq as a function of the radial distance r from the center of a single sheet of (100) copper consisting of 128 atoms, which is polarised by a point charge (q = -1.0 e) at heights of 3 and 6 Å above the center of the sheet. 'Spin' denotes the incorporation of a colinear treatment of spin, and '3.71' denotes the use of the SCC-DFTB optimised lattice constant to construct the sheet. As can be seen, none of the extra options significantly changed the final result.

Figure B.4 shows the charge redistribution results for four different single layered sheet sizes. It is observed that the relative error decreases with an increase of the system size.
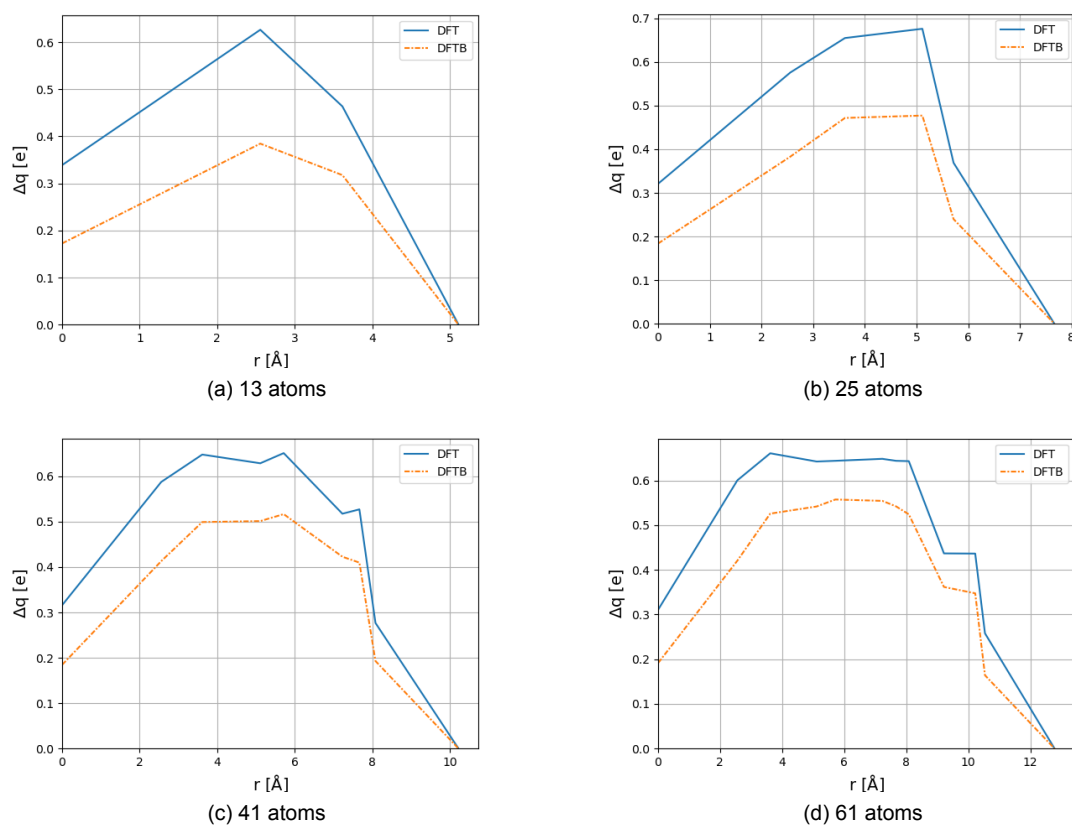

(a) 13 atoms

(b) 25 atoms

(c) 41 atoms

(d) 61 atoms

Figure B.4: Plots depicting the DFT (solid) and SCC-DFTB (dash-dotted) summed excess charge Δq as a function of the radial distance r from the center of single sheets of (100) copper of different sizes, which are polarised by a point charge (q = -1.0 e) at heights of 3 and 6 Å above the center of the sheets. The relative error between the DFT and SCC-DFTB calculations can be seen to decrease with an increase of the system size.

B.3. Multiple point charges

Lastly, figure B.5 gives a visualisation of the charge redistribution results on a single sheet of copper (consisting of 128 atoms), polarised using four different point charges. At a location just above the center of the sheet, two point charges of charge +1.0 e and -1.0 e are placed at 3 Å and 6 Å from the sheet, respectively. Similarly, opposite point charges of charge -1.0 e and +1.0 e are placed at 3 Å and 6 Å from the sheet, just below the center, the same distance from the center as the first location. The atoms closest to the point charges can be seen to be polarised similarly using both DFT and SCC-DFTB. However, on closer inspection, it turns out that, at these point charge distances, SCC-DFTB predicts a different sign of the excess charge for the surrounding atoms than DFT. This can be seen in figure B.6a By lowering the point charges in the SCC-DFTB simulation, it turns out that the signs of the surrounding atoms change, as depicted in figure B.6b. Therefore, it is suspected that the observed error in predicted sign, is a consequence of the reduced polarisability of SCC-DFTB.

(a) DFT   (b) SCC-DFTB
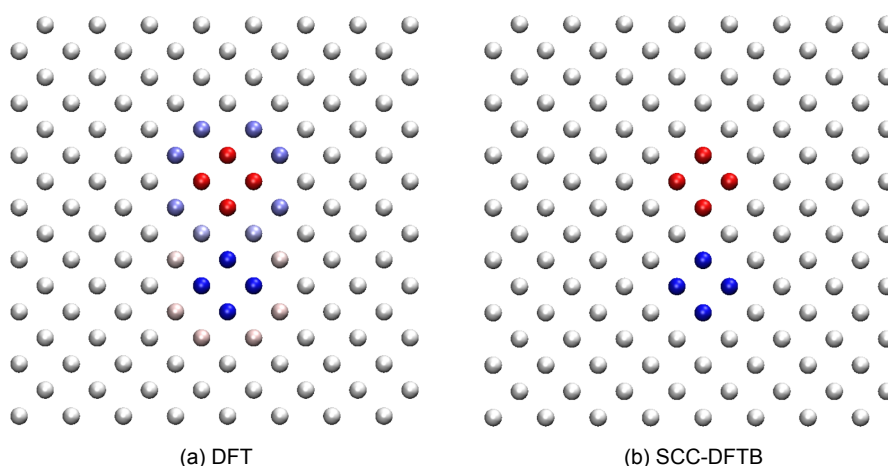
Figure B.5: Visualisation of a single layered (100) copper sheet, coloured using the computed excess charges Δq^i by both DFT (left, a.) and SCC-DFTB (right, b.), for the case where the sheet was polarised using four different point charges. At a location just above the center of the sheet, two point charges of charge +1.0 e and -1.0 e were placed at 3 Å and 6 Å from the sheet, respectively. Similarly, opposite point charges of charge -1.0 e and +1.0 e are placed at 3 Å and 6 Å from the sheet, just below the center at the same distance from the center as the first location. Looking at the figure, the charges were places exactly in the centers of the two groups of four atoms that are polarised the most. The same charge scale was used for both plots. Charges outside of the specified range were capped, and assigned the maximum colour values.
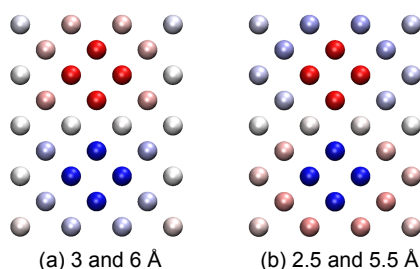

(a) 3 and 6 Å   (b) 2.5 and 5.5 Å

Figure B.6: Visualisation of the center part of the copper sheet given in figure B.5b, simulated using SCC-DFTB, for different distances of teh four point charges. The atoms are coloured using the computed excess charges Δq^i. The same charge scale was used for both plots, from -0.01 e (red) to 0.01 e (blue). Charges outside of this range were capped, and were assigned the maximum colour values. It can be observed that when the point charges are moved closer to the sheet, the sign of the predicted excess charges of the atoms surrounding the atoms nearest to the point charges, changes.
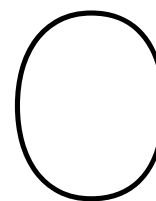
QM-MD time interval experiments

Here all results of the (chained together) testing loop, used for determining a suitable time interval between SCC-DFTB calculations in the QM-MD simulation and described in section 4.3, are given. The results of the 5 ps and 1 fs time interval loops are given again here, to allow for easy comparison with the 50 fs time interval loops. The figures are given in the order at which they occurred in the total loop.
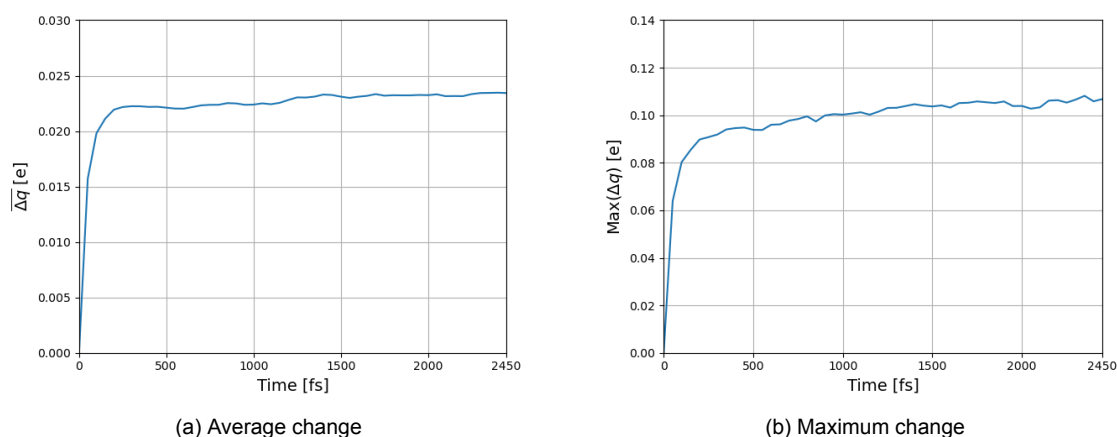

(a) Average change (b) Maximum change

Figure C.1: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, of the first loop in which a time interval between SCC-DFTB calculations of 50 fs was used.

The difference in the results of the two individual runs using a time interval of 50 fs, underscores why it was decided to use the presented simulation using loops with multiple values for the time interval. Apparently, the system configuration changed during the 5 ps time interval loop, which would have led to unrepresentative results if only the first 50 fs loop would had been considered. Only considering the first loop might have led to a different choice for the time interval between SCC-DFTB calculations in the QM-MD loop. The difference between the two 50 fs runs can be explained in multiple ways. Firstly, after the 5 ps run, more potassium ions had adsorbed on the cathode surface, which could lead to a relative increase in the polarisation effect. General changes in the double layer could also account for this. The reduced sample size, of only 50 calculations, could play a role in the observed difference as well. It can be seen that the average error, as compared to some initial configuration, only exceeded a value of 0.03 e after 5 ps, in the case of a used time interval of 5 ps. In comparison, when using a time interval of 50 fs, the average error already exceed a value of 0.03 e after 500 fs. In how far this can be attributed to relative changes in the double layer, or to the effect of the smaller time interval, is unclear. This could, at least partly, be a sign that using a longer time interval between SCC-DFTB calculations, lead to less accurate results.
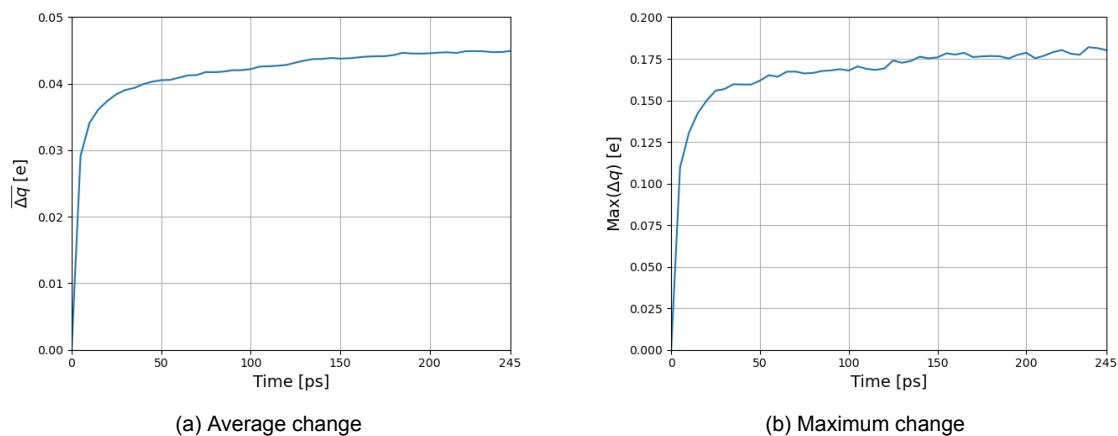
(a) Average change

(b) Maximum change

Figure C.2: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, using a time interval between SCC-DFTB calculations of 1 fs.
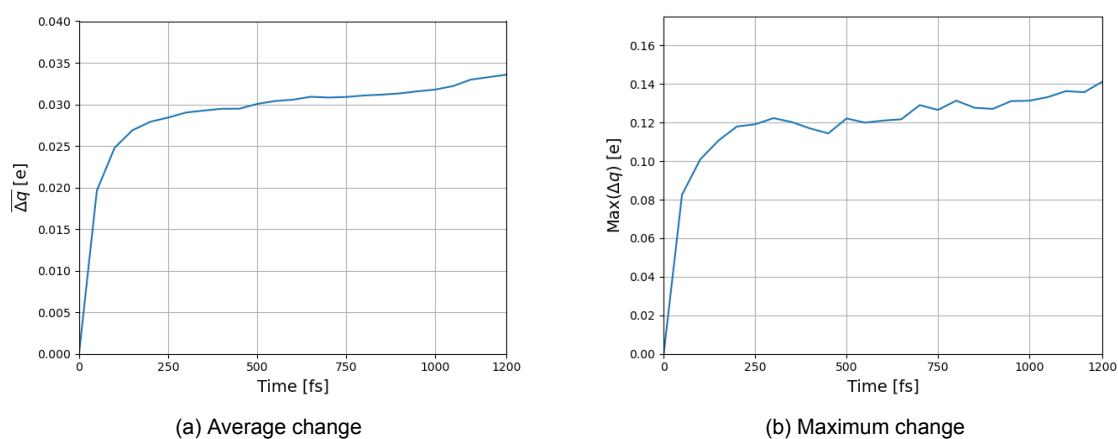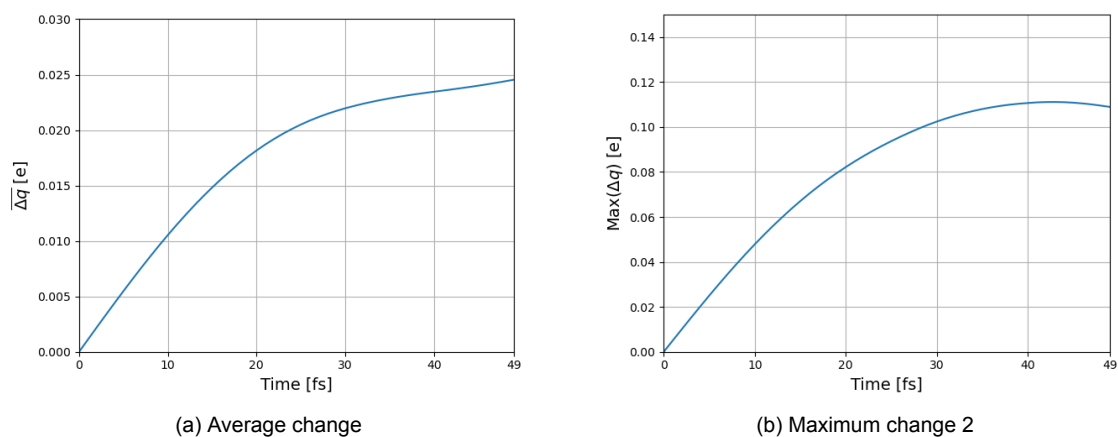

(a) Average change

(b) Maximum change

Figure C.3: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, of the second loop in which a time interval between SCC-DFTB calculations of 50 fs was used.

(a) Average change

(b) Maximum change 2

Figure C.4: Plots depicting the average change and the maximum change in the partial charges, as compared to some starting configuration, as a function of time, using a time interval between SCC-DFTB calculations of 1 fs. As discussed previously, after 49 fs, both the average and the maximum change in the charge values, exceed 50% of the average and maximum change values observed after 245 ps, as seen in figure 4.11

D
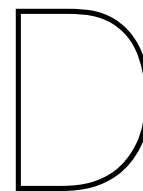
MSD trajectories

Figures D.1 and D.1 show plots of the individual trajectories of the traced atoms used in the MSD measurements, for atoms in the bulk and atoms adsorbed on the cathode surface, respectively.
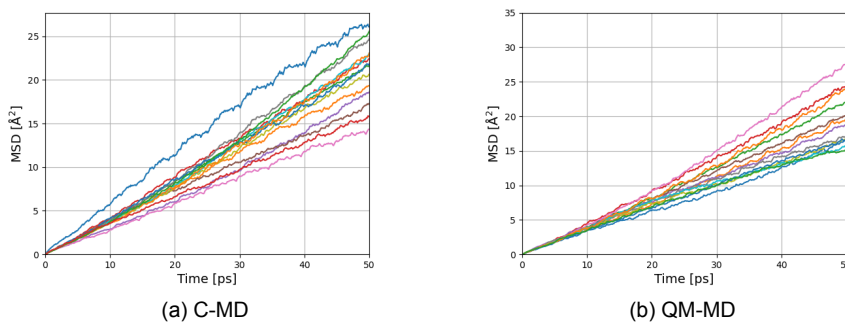

(a) C-MD

(b) QM-MD

Figure D.1: MSD results of individual trajectories of atoms in the bulk, for both the C-MD simulation (a, left) and the QM-MD simulation (b, right).
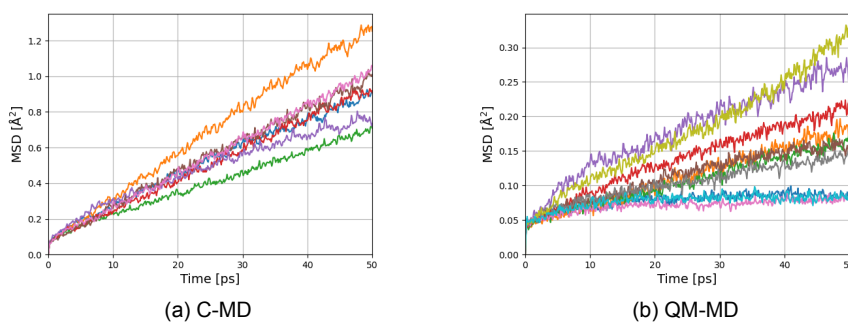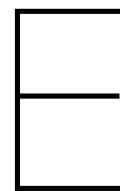

(a) C-MD

(b) QM-MD

Figure D.2: MSD results of the individual trajectories of atoms adsorbed on the cathode surface, for both the C-MD simulation (a, left) and the QM-MD simulation (b, right).

Additional results of the angular orientation of water

Here, an expanded version of figure 4.23 is provided, where also a histogram is plotted of the angular orientation of water molecules up to a distance of 7 Å away from the cathode. The dipole orientation is the same as given before in figure 4.23.
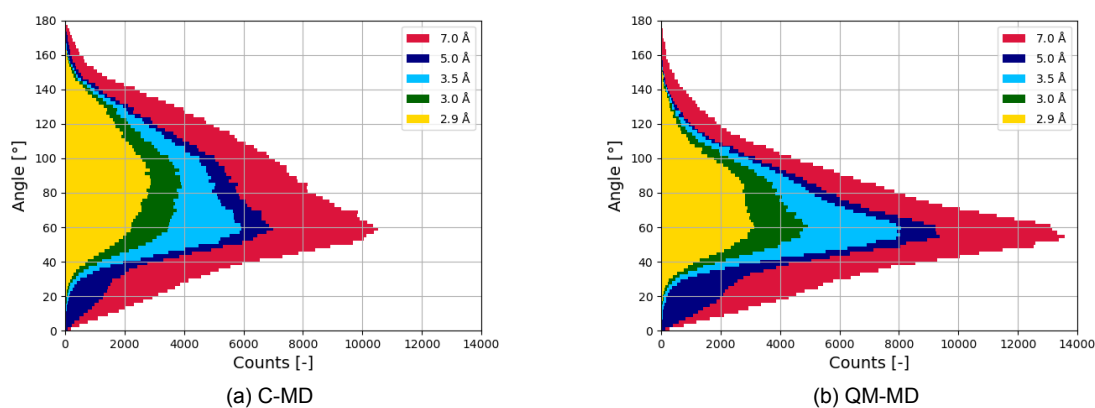

(a) C-MD   (b) QM-MD

Figure E.1: Histograms of the angular orientation of water molecules near the cathode, in the C-MD simulation (a, left) and the QM-MD simulation (b, right). The colours relate to the maximum distance from the cathode that water molecules were taken into account for a calculation. So, for 5.0 Å, for all water molecules were up to a distance of 5.0 Å from the cathode, the angle with respect of the normal of cathode was computed and stored in the dark blue histogram. Both the results of the C-MD and the results of the QM-MD simulation show a skewed profile, with a preferential angular orientation smaller than 90°. In the QM-MD simulation, this preference is relatively increased compared to the C-MD simulation. In this figure, the relatively increased preference for the angular orientation in the QM-MD simulation, is shown to persist at least up to distances of 7 Å away from the copper surface (more counts are added for angular orientations below 90°, than for angular orientations over 90°).

Figure E.2 shows the angular orientation of water molecules at a maximum distance of 3.5 Å from the anode surface.
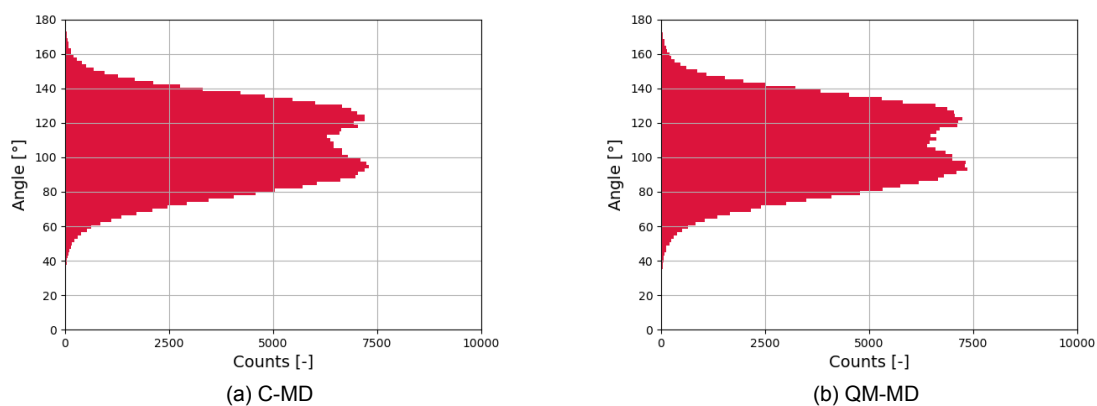

(a) C-MD

(b) QM-MD

Figure E.2: Histograms of the angular orientation of water molecules near the anode, in the C-MD simulation (a, left) and the QM-MD simulation (b, right), for molecules at a maximum distance of 3.5 Å from the anode surface. The results of the C-MD and the results of the QM-MD simulation show a similarly skewed profile, with a preferential angular orientation larger than 90°. With barely any ions nearby or adsorbed on the anode, and the positive net charge on the surface nearby, this would be expected (anode is on right side).