universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„The predicted influence of temperature on knock-on damage in graphene“

verfasst von / submitted by

Alexandru Ionut Chirita Mihaila BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2018 / Vienna, 2018

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066876
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Physik
Betreut von / Supervisor:	Assoc. Prof. Dr. Jani Kotakoski
Mitbetreut von / Co-Supervisor:	Dr. Toma Susi

Contents

Abstract

Beobachtungen auf atomarer Ebene von Materialien werden oft unter Verwendung eines Transmissionselektronenmikroskops durchgeführt. Abhängig von der Beschleunigungsspannung der Elektronen, kann der Strahl Defekte in der Probe induzieren. Dies geschieht wenn der Impuls der einfallenden Elektronen hoch genug ist um Kohlenstoffatome aus dem Gitter herauszuschlagen. Atomistische Simulationen sind eine gute Möglichkeit solche Ereignisse zu testen und vorherzusagen. Hier untersuchen wir, wie sich die Temperatur auf den Schwellwert für Atomverschiebungen auswirkt und wie dieser Effekt bei der Berechnung des Wirkungsquerschnitts berücksichtigt wird mithilfe einer auf Dichtefunktionaltheorie basierenden Tight Binding-Methode.

Chapter 1

Introduction

1.1 Motivation

Atomic level observations of materials, are often carried out using a transmission electron microscope (TEM). In a TEM, a beam of accelerated electrons is transmitted through the sample, interacting with it as it passes through. Depending on the acceleration voltage of the electrons, the beam may induce defects in the sample. This happens when the momentum of the incoming electrons is high enough to knock out carbon atoms from the lattice. Hence, the electron beam of the TEM can alter the morphology of graphene, a carbon based two-dimensional material.

For example, it has been shown that the electron beam can be used to transform graphene into single-atom carbon chains and carbon nano-ribbons [1, 2] or to transform a graphene flake into a fullerene [3]. Another example of electron beam induced manipulation was presented in [4], where graphene turned into an amorphous two-dimensional carbon membrane under exposure to the electron beam.

In scanning transmission electron microscopy (STEM) [5], a highly focused electron beam can be placed on one atom at a time. This small probe along with the tunable electron energy have even made it possible to manipulate individual atoms (like Si) in the graphene lattice, as was shown in [6].

Atomistic simulations are a good way to test and predict such events. In such simulations, one needs a way to describe the interactions between atoms, for example, using analytical potentials. These potentials are not computationally expensive, so they can be run on systems with a large number of atoms. However, there are also more precise ways to test whether the predictions using analytical potentials are trustworthy or not, for example with density functional theory (DFT) or

density functional tight-binding (DFTB), which are computational modeling methods based on quantum mechanics.

DFT and DFTB [7] have previously been used to simulate displacements of carbon atoms in graphene. In order to eject an atom out of the lattice, the energy transferred to the atom needs to exceed the displacement threshold of it. Here, we study how the temperature affects the displacement threshold and how to include this effect when calculating the displacement cross section. An accurate description of the displacement cross section helps us understand how often these displacements occur. Hence, this master's thesis will investigate the irradiation effect of the electron beam on carbon atoms in graphene at various temperatures by means of atomistic simulations.

However, up to date, there is no theoretical model to predict the role of temperature on the displacement cross-section of graphene. This master's thesis will investigate the irradiation effect of the electron beam on carbon atoms in graphene at various temperatures by means of atomistic simulations.

1.2 Graphene

Along the timeline of material science there have been some major breakthroughs, like the discovery of polymers (1830), semi conductors (1833) and plastic (1941). In 2004, Andre Geim, a physics professor at the University of Manchester, and his Ph.D. student, Konstantin Novoselov, isolated the first two-dimensional (2D) material ever, a one-atom thick layer of carbon. Physicists had speculated about such a material, called graphene, even though it had been argued in 1935 and 1937 [8, 9] that 2D materials were thermodynamically unstable because the thermal fluctuations would become comparable to interatomic distances. So it was thought that graphene would be unstable at room temperature. However, after a few unlucky attempts to publish their discovery, in October 2004 their publication was accepted [10] and six years later they received the Nobel prize in physics for it. Since then, graphene has been hailed as a miracle material.

Graphene is made out of a single layer of carbon atoms. These carbon atoms are tightly bound in a 2D hexagonal lattice (Figure 1.1), this being the building block for other graphitic materials. The lattice vectors can be written as

$$\vec{a}_1 = \frac{a}{2}\left(3, \sqrt{3}\right), \quad \vec{a}_2 = \frac{a}{2}\left(3, -\sqrt{3}\right) \tag{1.1}$$

where $a \approx 0.142$ nm is the nearest neighbor carbon-carbon distance. This structure is built up of two triangular sublattices, A (blue colored atoms) and B (red-colored). The three nearest-

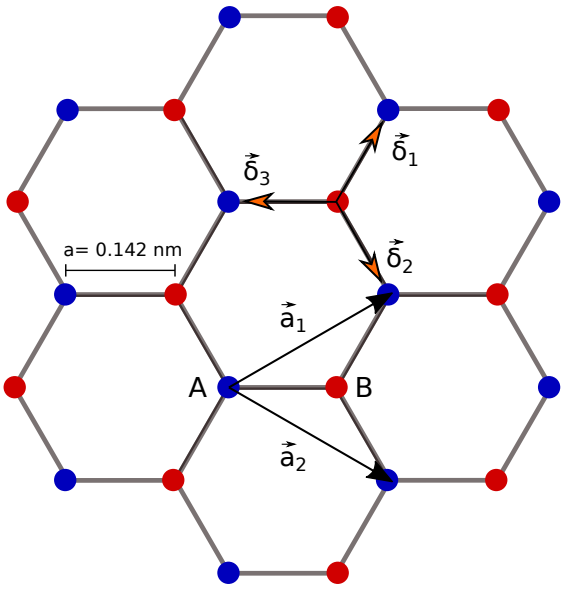

Figure 1.1: Lattice structure of graphene.

neighbour vectors are

$$\vec{\delta}_1 = \frac{a}{2}\left(1, \sqrt{3}\right), \quad \vec{\delta}_2 = \frac{a}{2}\left(1, -\sqrt{3}\right), \quad \vec{\delta}_3 = a\left(-1, 0\right). \tag{1.2}$$

The reciprocal lattice vectors $\vec{b}_1$ and $\vec{b}_2$ are given by

$$\vec{b}_1 = \frac{2\pi}{3a}\left(1, \sqrt{3}\right), \quad \vec{b}_2 = \frac{2\pi}{3a}\left(1, -\sqrt{3}\right). \tag{1.3}$$

The important physics of graphene takes place at the corners of the Brillouin zone (Figure 1.2), namely the two points of high symmetry K and K':

$$\vec{K}' = \left(\frac{2\pi}{3a}, \frac{2\pi}{3\sqrt{3}a}\right), \quad \vec{K} = \left(\frac{2\pi}{3a}, -\frac{2\pi}{3\sqrt{3}a}\right). \tag{1.4}$$

They are called Dirac points because the electron transport in their vicinity can be described by the massless Dirac equation in two-dimensions [11]. This means that one can think of the electrons as free massless Dirac particles moving at an effective velocity v_F instead of interacting with a periodic potential.

The electronic structure of one isolated C atom is $(1s)^2 (2s)^2 (2p)^4$. Each carbon atom has a total of six electrons; two on the inner shell and four on the outer shell. In graphene, three of the outer electrons take part in chemical bonding, where the orbitals hybridize in such a way that every carbon atom is strongly bonded to its neighbors by σ bonds. The σ-bonds consist of the overlap of sp^2 hybridized orbitals. The sp^2 hybridization is the result of the combination of the $2s$-orbital with two p-orbitals, the $2p_x$ and the $2p_y$ orbital as seen in Figure 1.3. These orbitals have an angle of 120° between them and are responsible for the hexagonal structure of graphene.

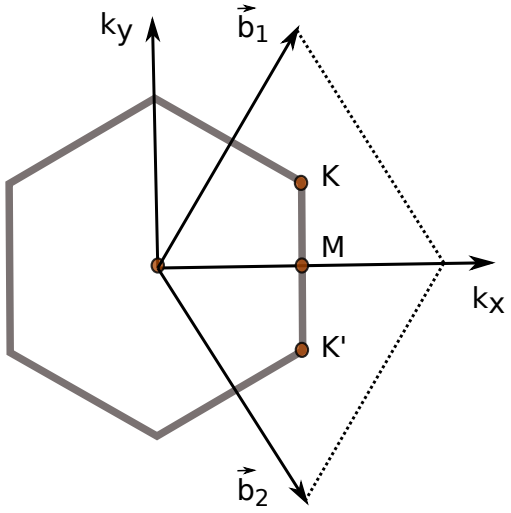

Figure 1.2: Reciprocal lattice. First Brillouin zone with corresponding symmetry points.

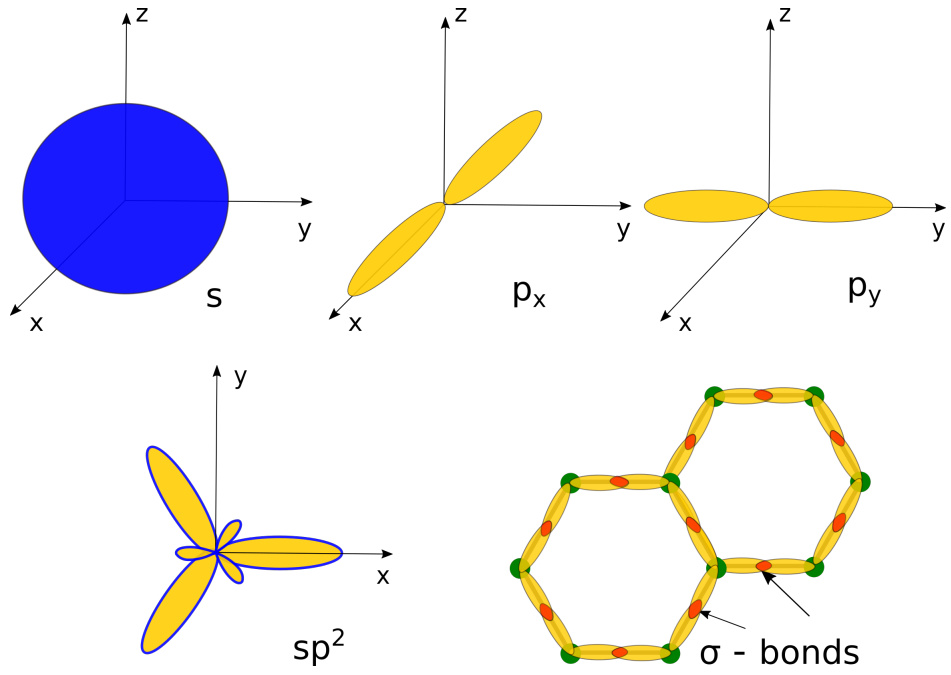

Figure 1.3: Hybridization of carbon in the graphene lattice.

This leaves one free unhybridized orbital in the out-of-plane position, the $2p_z$ or (π orbital) which determines the electronic structure, because it allows the hopping of electrons between the orbitals. Each carbon atom contributes to one $2p_z$-orbital as shown in Figure 1.4.

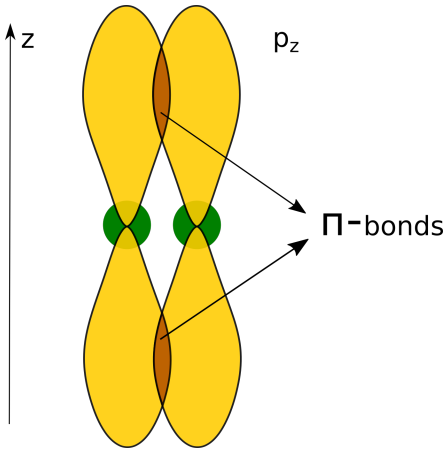

Figure 1.4: Hybridization of the out-of-plane orbitals and the π bonds.

One can imagine that in a real crystal, these $2p_z$-orbitals merge in such a way, that they create a sort of "cloud" beneath and above the graphene sheet.

Of particular importance for the electronic properties of graphene is the π-orbital. This single carbon orbital, which is left unfilled by the electrons and which is oriented normal to the plane, can

shelter two electrons with spin $\pm \frac{1}{2}$. The basis of electron states contains two π states belonging to the atoms from sublattices A and B. In the nearest-neighbor approximation, considering only hopping processes within the sublattices, the tight binding Hamiltonian for electrons in graphene is described by the 2x2 matrix :

$$\hat{H}(\vec{k}) = \begin{pmatrix} 0 & tS(\vec{k}) \\ tS^{\star}\vec{k} & 0 \end{pmatrix}, \tag{1.5}$$

where $\vec{k}$ is the wave vector, t the nearest-neighbor hopping energy and

$$S(\vec{k}) = \sum_{\delta} e^{i\vec{k}\vec{\delta}} = 2\exp\left(\frac{ik_x a}{2}\right)\cos\left(\frac{k_y a\sqrt{3}}{2}\right) + exp\left(-ik_x a\right). \tag{1.6}$$

The energy bands, which describe the range of energies an electron may have, are derived from this Hamiltonian and have the form [12]:

$$E(\vec{k}) = \pm t|S(\vec{k}) = \pm t\sqrt{3 + f\left(k\right)} \tag{1.7}$$

with

$$f\left(k\right) = 2\cos\left(\sqrt{3}k_y a\right) + 4\cos\left(\frac{\sqrt{3}}{2}k_y a\right)\cos\left(\frac{3}{2}k_x a\right), \tag{1.8}$$

where t is the approximated at about 2.8 eV. The plus sign corresponds to the upper π and the minus sign to the lower π^* band. One can calculate that $S(\vec{K}) = S(\vec{K'}) = 0$, which means that the bands are crossing.

Because of the network of sp^2 hybridized bonds, graphene is the strongest material ever measured [13]. It has a tensile strength of 130 GPa and a 2D elastic constant of about 300 N/m. Its thermal properties are equally remarkable. Graphene has a thermal conductivity of up to 5000 W/m K at room temperature [14], which is about 20 times higher than that of copper. Also it's thermal expansion coefficient is negative and large [15], almost up to 10 times larger than graphite. Graphene's quality unveils itself also through it's electronic properties. It is a zero-gap semiconductor and the charge carriers have an electron mobility that can exceed 15,000 $\mathrm{cm}^2/\mathrm{Vs}$ even at room temperature [10, 16]. However, these remarkable electronic properties are mostly determined by the defect-free pristine structure. So the electronic and transport properties are sensitive to changes in the structure, such as having defects or impurities [17]. These deviations from the pristine structure can help reshape the electronic properties and achieve new functionalities. The electron beam of a TEM can be used not only for the purpose of imaging graphene, but also to alter the morphology and therefore change its properties.

1.3 (Scanning) transmission electron microscopy

In order to visualize atoms one can not use visible light because of its long wavelength. Light microscopes can resolve images in the order of hundreds of nm, but this is not enough to get a glimpse at the atomic structure of any material. To get a better understanding of the structure of materials, electron microscopes have been designed [18]. As the name says, they use electrons, which have wavelenghts in the order of pico meters (10^{-12} m), small enough to analyze materials at the level of single atoms. Transmission electron microscopes (TEMs) and scanning transmission electron microscopes (STEMs) [5] are powerful tools to image materials at atomic resolution. This was made possible thanks to the advances made in instrumentation to suppress electron optic aberrations [19–21]. A schematic comparison of both TEM and STEM is presented in Figure 1.5.

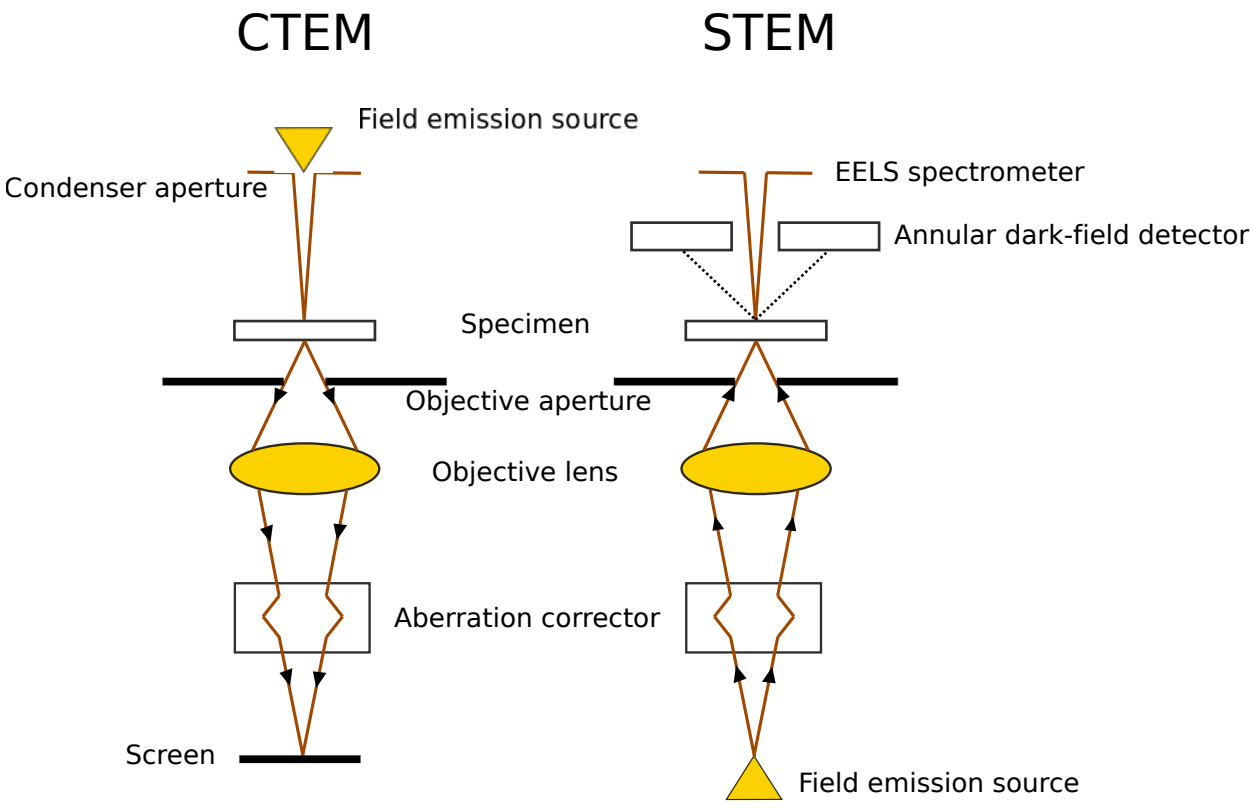

Figure 1.5: Comparison between a conventional transmission electron microscope (CTEM) and a scanning transmission electron microscope (STEM)

As is presented in Figure 1.5 in a conventional TEM the electrons produced are collected in a condenser aperture and focused as a parallel beam on the specimen. After the electrons are transmitted through the specimen, the objective aperture is used to exclude high-angle scattered

electrons. The beam gets collected by the objective lens, which is used to focus and magnify the image. Since the lenses bend electrons of different energies (different wavelengths), it will result in different focal lengths and this gives rise to chromatic aberrations. Also the electrons going through the periphery of the lens will be refracted more than those passing along the axis, and they will create spherical aberrations. Aberrations result in the blurring of the image and thus loss of resolution. Because the aberrations of electromagnetic lenses are always positive it was quite a challenge to overcome them. However, sextupole and octupole magnets can nowadays be used to correct them. After the corrector, the electrons will hit a screen where the image is recorded.

In Figure 1.5 one can see that in a STEM the focusing of the electron beam happens before it hits the specimen. The field-emission gun provides with an electron beam (low energy spread), which goes through the aberration corrector first and then is focused by the objective lens onto the specimen. The elastically scattered electrons are collected by the annular dark-field detector (ADF) and the image is formed. The high angle ADF detector makes it possible to form high resolution images, where the contrast of an atom is directly related to the atomic number Z. In terms of signal collection efficiency this gives an advantage and permits the main beam (inelastically scattered electrons) to pass on to the electron energy loss spectrometer (EELS). The EELS spectrometer makes the elemental identification possible for low atomic number and light elements such as C, O, N and others. In contrast to a TEM, where the sample is illuminated by a parallel electron beam over the whole field of view, in a STEM a very precise, highly focused electron beam scans over the designated area providing a resolution of ca. 1 Å. This makes the STEM one of the most adequate tools to image graphene and other low dimensional solids with an unprecedented quality.

Having such a small electron probe along with the tunable energy, the STEM is the right tool to controllably manipulate atoms throughout the graphene lattice [6]. Electron bombardment can prove to be useful. First, given that the transfer of energy can be tuned just around the threshold value to create defects, individual defects may be generated. Second, the electron beam is a sub-nanometric probe, allowing to controllably manipulate single atoms in low dimensional solids. Thus, it is of scientific and technological importance to understand the effects of electron irradiation.

1.4 Irradiation effects on solids

Radiation damage is one of the key limitations of transmission electron microscopy. Almost any material could be resolved with the current instrumentation, but the concern is whether the sample remains stable until the image is obtained.

When a highly energetic particle such as an electron hits the sample, different events of energy or momentum transfer may take place. The most significant are:

- ionization of individual atoms,
- collective electronic excitations such as plasmons,
- bond breaking,
- phonon excitation,
- atom displacements (knock-on effect).

The energy of the incoming electrons is of particular importance since different events have different energy dependencies. When we consider radiation effects in carbon-based materials, it is useful to divide them into those that cause displacement of atoms and those that do not (excitations). With increasing electron energy, excitations become less important whereas knock-on effects rise.

1.4.1 Knock-on atom displacements

Transferred energies

When highly energetic electrons scatter from the electrostatic potential of the nuclei of the atoms in the specimen, atom displacements may occur. These knock-on collision events are the most notable effect of electron irradiation in carbon nanostructures. They lead to defect creation, such as Stone-Wales [22] defects, and may help changing the morphology in such a manner that the electronic and magnetic properties can be tailored [23].

Due to the great difference in mass between an electron and a nucleus, the amount of transferred energy is limited. The momentum transfer is a result of the change in the direction of the electron. The angular dependence of the transferred energy E to the nucleus is

$$E(\theta) = E_{\max} \cos^2(\theta) \tag{1.9}$$

where θ is the angle between the initial direction of the electron and the direction in which the nucleus gets scattered and $E_{\max}$ is the maximum energy which is transferred to the nucleus by a head-on collision ($\theta = 0$). The geometry of scattering an electron to a nucleus is presented in Figure 1.6.

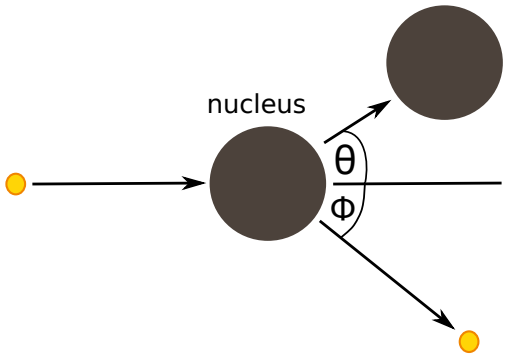

Figure 1.6: Schematic of an electron-nucleus collision

If the energy transferred to the atom is large enough to produce a vacancy in the lattice which does not spontaneously recombine, then the target atom is considered displaced (knocked out). This minimum transferred energy needed to knock out an atom is called the displacement threshold energy E_{d}.

In a TEM experiment it is worth considering the vibration of the atoms out-of-plane. Due to momentum conservation, if an atom is hit by an electron while it happens to move parallel to the incoming beam it can acquire a higher maximum transferred energy $E_{\max}$ than if it were at rest [24]. To illustrate this situation, we need to consider a relativistic scattering process between an electron and a nucleus. The electron (mass m, energy E_{e}, momentum p_{e}), a relativistic projectile, collides with a moving, non-relativistic target, the nucleus (mass M, energy E_{n}, momentum p_{n}). Taking into account that the energy and momentum are conserved we have the equations

$$E_{\mathrm{e}}^{\mathrm{tot}} + E_{\mathrm{n}}^{\mathrm{tot}} = \tilde{E}_{\mathrm{e}}^{\ \mathrm{tot}} + \tilde{E}_{\mathrm{n}}^{\ \mathrm{tot}} \tag{1.10}$$

and

$$\vec{p_e} + \vec{p_n} = \vec{\tilde{p}_{\mathrm{e}}} + \vec{\tilde{p}_{\mathrm{n}}} \tag{1.11}$$

where the quantities $\tilde{E}_{\mathrm{e}}^{\ \mathrm{tot}}$, $\tilde{E}_{\mathrm{n}}^{\ \mathrm{tot}}$ and $\vec{\tilde{p}_{\mathrm{e}}}$, $\vec{\tilde{p}_{\mathrm{n}}}$ represent the total energies and the momenta after the collision for the electron and nucleus, respectively. We need to express the maximum transferred energy to the nucleus $\tilde{E}_{\mathrm{n}}^{\ \mathrm{tot}}$ as a function of the kinetic energy of the electron E and the velocity of the nucleus v. As can be seen in Figure 1.6 the electron and the nucleus scatter under different

angles, ϕ and θ respectively, after the collision. Considering a head-on collision along the x-axis in a two dimensional plane we can express the momentum conservation Equation 1.11

$$p_{\mathrm{e_x}} + p_{\mathrm{n_x}} = |\tilde{p}_{\mathrm{e}}| \cdot \cos\phi + |\tilde{p}_{\mathrm{n}}| \cdot \cos\theta \tag{1.12}$$

and

$$p_{\mathrm{e_y}} + p_{\mathrm{n_y}} = |\tilde{p}_{\mathrm{e}}| \cdot \sin\phi - |\tilde{p}_{\mathrm{n}}| \cdot \sin\theta. \tag{1.13}$$

In a pure head-on collision it is assumed that the nucleus only vibrates parallel to the direction of the electron beam, neither having a component of momentum, and we can write $p_{\mathrm{e_y}} + p_{\mathrm{n_y}} = 0$. Therefore we can express the nucleus scattering angle θ in terms of ϕ

$$\sin\theta = \frac{|\tilde{p}_{\mathrm{e}}| \cdot \sin\phi}{|\tilde{p}_{\mathrm{n}}|}. \tag{1.14}$$

Now, using the trigonometric identity $\cos^2\theta + \sin^2\theta = 1$ and substituting in Equation 1.12 we get

$$p_{\mathrm{e_x}} + p_{\mathrm{n_x}} = |\tilde{p}|_{\mathrm{e}} \cdot \cos\phi + |\tilde{p}_{\mathrm{n}}| \cdot \sqrt{1 - \frac{|\tilde{p}_{\mathrm{e}}|^2 \cdot \sin^2\phi}{|\tilde{p}_{\mathrm{n}}|^2}}. \tag{1.15}$$

We can express the energy in terms of momentum according to the equation

$$E_{\mathrm{e}}^{\mathrm{tot}} = E_{\mathrm{e}} + mc^2 = \sqrt{p_{\mathrm{e_x}}^2 c^2 + m^2 c^4}, \tag{1.16}$$

where E_{e} is the kinetic energy of the incoming electron, to get

$$p_{\mathrm{e_x}} = \sqrt{\frac{E_{\mathrm{e}}\left(E_{\mathrm{e}} + 2mc^2\right)}{c^2}}. \tag{1.17}$$

We can express the momentum $p_{\mathrm{n_x}}$ of the nucleus classically since the mass M is much larger ($\sim 10^4$ times) than that of the electron as

$$p_{\mathrm{n_x}} = \sqrt{2ME_{\mathrm{n}}}, \tag{1.18}$$

where $E_{\mathrm{n}} = \frac{Mv^2}{2}$ is the kinetic component of the energy of the nucleus before collision.

Since the rest energies of both the electron and the nucleus do not change in the collision, we can use the energy conservation relationship (Equation 1.10) to express the energy of the electron after collision using only kinetic energies as

$$E_{\mathrm{e}} + \cancel{mc^2} + E_{\mathrm{n}} + \cancel{\frac{Mv^2}{2}} = \tilde{E}_{\mathrm{e}} + \cancel{mc^2} + \tilde{E}_{\mathrm{n}} + \cancel{\frac{Mv^2}{2}} \tag{1.19}$$

Then we can use the relativistic energy-momentum relationship Equation 1.16 to express $|\tilde{p}_\mathrm{e}|$ as

$$|\tilde{p}_\mathrm{e}| = \sqrt{\frac{\tilde{E}_\mathrm{e}\left(\tilde{E}_\mathrm{e} + 2mc^2\right)}{c^2}} = \sqrt{\frac{\left(E_\mathrm{e} + E_\mathrm{n} - \tilde{E}_\mathrm{n}\right)\left(\left(E_\mathrm{e} + E_\mathrm{n} - \tilde{E}_\mathrm{n}\right) + 2mc^2\right)}{c^2}}. \tag{1.20}$$

The kinetic energy of the electron $E_\mathrm{e} = eU$ (e.g 80 keV), where e is the elementary charge and U the acceleration voltage, is much higher than that of the nucleus before the collision $E_\mathrm{n} \sim 0.04\,\mathrm{eV}$ or after $\tilde{E}_\mathrm{n} \sim 10\,\mathrm{eV}$. In earlier studies [24] and [25], the assumption that $E_\mathrm{e} + E_\mathrm{n} - \tilde{E}_\mathrm{n} \approx E_\mathrm{e} + E_\mathrm{n}$ was done. This approach is not well justified since $E_\mathrm{n} < \tilde{E}_\mathrm{n}$. One should either approximate $E_\mathrm{e} + E_\mathrm{n} - \tilde{E}_\mathrm{n} \approx E_\mathrm{e} - \tilde{E}_\mathrm{n}$, for which the equation can only be solved numerically, or

$$E_\mathrm{e} + E_\mathrm{n} - \tilde{E}_\mathrm{n} \approx E_\mathrm{e}, \tag{1.21}$$

which delivers an algebraically solvable equation. Let us proceed to solve the better approximation. Substituting for $p_{\mathrm{e_x}}$, $p_{\mathrm{n_x}}$, $\tilde{p}_\mathrm{e}$ and $\tilde{p}_\mathrm{n}$ in Equation 1.15 we end up with this expression for the momentum conservation

$$\frac{1}{c}\sqrt{E_\mathrm{e}\left(E_\mathrm{e} + 2mc^2\right)} + \sqrt{2ME_\mathrm{n}} = \frac{\cos\phi}{c}\sqrt{E_\mathrm{e}\left(E_\mathrm{e} + 2mc^2\right)} + \sqrt{2M\tilde{E}_\mathrm{n} - \frac{E_\mathrm{e}\left(E_\mathrm{e} + 2mc^2\right)\sin^2\phi}{2M\tilde{E}_\mathrm{n}c^2}}. \tag{1.22}$$

Our goal is to solve Equation 1.22 for $\tilde{E}_\mathrm{n}$, which corresponds to the maximum transferred energy to the nucleus. To simplify the calculation we make the following substitutions:

$$r = \frac{1}{c}\sqrt{E_\mathrm{e}\left(E_\mathrm{e} + 2mc^2\right)} + \sqrt{2ME_\mathrm{n}} \quad \text{and} \quad t = \frac{1}{c}\sqrt{E_\mathrm{e}\left(E_\mathrm{e} + 2mc^2\right)}. \tag{1.23}$$

Equation 1.22 can thus be written as:

$$r = t\cos\phi + \sqrt{2M\tilde{E}_\mathrm{n} - \frac{t^2\sin^2\phi}{2M\tilde{E}_\mathrm{n}}} \tag{1.24}$$

Bringing the $\cos$ term to the left side of the equation and squaring both sides leads to

$$r^2 - 2rt\cos\theta + \cos^2\phi t^2 = 2M\tilde{E}_\mathrm{n} - \frac{t^2\sin\phi^2}{.}2M\tilde{E}_\mathrm{n} \tag{1.25}$$

We can further make the following substitution $a = r^2 - 2r\cos\theta t + \cos^2\theta t^2$ and then bring Equation 1.25 to a common denominator, from where we can solve the second order polynomial for $\tilde{E}_\mathrm{n}$

$$4M^2\tilde{E}_\mathrm{n}^2 - 2M\tilde{E}_\mathrm{n}a - t^2\left(1 - \cos^2\theta\right) = 0. \tag{1.26}$$

Given the fact that the electron is backscattering, the angle $\phi = \pi \ (180°)$ and therefore we get the solution for $\tilde{E}_{\mathrm{n}_{1,2}}$

$$\tilde{E}_{\mathrm{n}_{1,2}} = \frac{2Ma \pm \sqrt{4M^2a^2 + 16M^2t^2\left(1 - \cos^2\phi\right)}}{8M^2}. \tag{1.27}$$

If we set $\phi = \pi \ (180°)$, one of the solutions will be equal to zero, and we are left with

$$\tilde{E}_{\mathrm{n}} = \frac{2Ma + 2Ma}{8M^2} = \frac{4M\left(r^2 + 2rt + t\right)}{8M^2} = \frac{(r+t)^2}{2M}. \tag{1.28}$$

Substituting r and t back in Equation 1.28 we get the maximum energy that an electron can transfer to a nucleus of mass M that is moving with velocity v as

$$E_{\max}\left(E_{\mathrm{e}}, v\right) = \frac{\left(2\sqrt{E_{\mathrm{e}}\left(E_{\mathrm{e}} + 2mc^2\right)} + Mvc\right)^2}{2Mc^2}, \tag{1.29}$$

where the kinetic term of the electron can be directly linked to the acceleration voltage U and the elementary charge e, $E_{\mathrm{e}} = eU$. For the case where the target atoms is at rest, we set v=0 and see that by doing so we get

$$E_{\max}\left(E_{\mathrm{e}}\right) = \frac{2E_{\mathrm{e}}\left(E_{\mathrm{e}} + 2m_{\mathrm{e}}c^2\right)}{Mc^2}, \tag{1.30}$$

which coincides with the equation derived in [26].

A comparison between the approximations and what energy values they deliver, can be seen in Figure 1.7. One can tell from the plots, that the difference in the output these approximations produce is negligible.

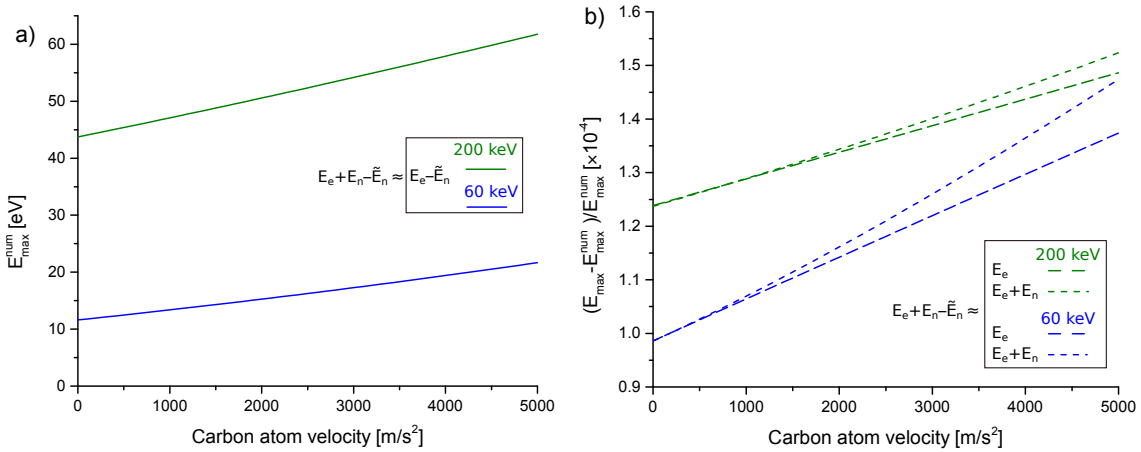

Figure 1.7: (a) The maximum transferred energy $E_{\max}^{\text{num}}$ to a carbon atom for various atom velocities v at 60 keV and 200 keV, calculated with the numerical approximation. At this scale, the other two approximations would be indistinguishable from $E_{\max}^{\text{num}}$. (b) The relative difference between the other two approximations for the maximum transferred energy and the numerical one.

Displacement cross section

A way to predict how often knock-on collisions occur [26] given a certain beam current density j is to calculate the displacement rate p of each atom

$$p = \sigma j, \tag{1.31}$$

where σ is the displacement cross section. The cross section for Coulomb scattering between a relativistic electron and a corresponding target nucleus was derived by N. Mott [27] as a solution to the Dirac equation [28]. McKinley and Feshbach extended this work [29] to accurately describe cross sections of ions up to a medium Z (atomic number)

$$\sigma(\theta) = \sigma_{\text{R}} \left[1 - \beta^2 \sin^2(\theta/2) + \pi \frac{Ze^2}{\hbar c} \beta \sin(\theta/2) \left(1 - \sin(\theta/2)\right) \right], \tag{1.32}$$

where $\beta = v/c$ is the ratio between the electron speed and the speed of light and σ_{R} is the classical Rutherford scattering cross section

$$\sigma_{\text{R}} = \left(\frac{Ze^2}{4\pi\epsilon_0 2m_0c^2} \right)^2 \frac{1-\beta^2}{\beta^4} \csc^4(\theta/2), \tag{1.33}$$

where ϵ_0 is the vacuum permittivity and m_0 is the mass of the electron. Considering the notations of Equation 1.9, we can write Equation 1.32 as a function of the energy E as in [28]

$$\sigma(E) = \left(\frac{Ze^2}{4\pi\epsilon_0 2m_0c^2} \frac{E_{\max}}{E} \right)^2 \frac{1-\beta^2}{\beta^4} \left[1 - \beta^2 \frac{E}{E_{\max}} + \pi \frac{Ze^2}{\hbar c} \beta \left(\sqrt{\frac{E}{E_{\max}}} - \frac{E}{E_{\max}} \right) \right]. \tag{1.34}$$

In a monoatomic gas, the energy transferred by the electron is converted to kinetic energy. The same happens in a crystalline system whether or not the transferred energy exceeds the bonding energy of the atom. Displacements can only occur if the transferred energy between the electron and the atom is larger than the displacement threshold E_{d} [28]. Transferred energies lower than displacement threshold are converted into vibrational energy of the lattice.

In order to quantify experimental irradiation, the total displacement cross section needs to be considered. This is obtained by integrating the cross section from Equation 1.34 inside the energy domain S where the displacement conditions are satisfied

$$\sigma_{\mathrm{d}} = \int_{S(E_{\max} > E_{\mathrm{d}})} \sigma\left(E\right) \frac{4\pi}{E_{\max}} dE. \tag{1.35}$$

Under the hypothesis that E_{d} is an isotropic function, Seitz and Koehler [30] derived the following formula for the displacement cross section:

$$\begin{aligned}\sigma_{\mathrm{d}} = 4\pi \left(\frac{Ze^2}{4\pi\epsilon_0 2m_0c^2}\right)^2 \frac{1-\beta^2}{\beta^4} \left\{\frac{E_{\max}}{E_{\mathrm{d}}} - 1 - \beta^2 \ln\left(\frac{E_{\max}}{E_{\mathrm{d}}}\right)\right. \\ \left. + \pi\frac{Ze^2}{\hbar c}\beta\left[2\left(\frac{E_{\max}}{E_{\mathrm{d}}}\right)^{1/2} - \ln\left(\frac{E_{\max}}{E_{\mathrm{d}}}\right) - 2\right]\right\}.\end{aligned} \tag{1.36}$$

Equation 1.36 has been used to evaluate the total knock-on cross section in different materials and also carbon nanostructures. Efforts have been done to improve the theoretical cross section by assuming that the target atoms are moving. First Meyer et. al. [24] showed that under 80 keV electron irradiation, the defect free graphene lattice remained undisturbed and that the knock-on damage begins a few keV above this energy. To theoretically predict the displacement cross section they approximated the phonon distribution of the material using the Debye model. They extracted the distribution of atom velocities from this model and calculated the maximum transferred energy $E_{\max}\left(v, E_{\mathrm{e}}\right)$ (Equation 1.29) as a function of electron energy E_{e} and atom velocity v. Hence, if we assume that the lattice is static and that the target atom is at rest (Equation 1.30), we fail to theoretically describe any displacements below 110 keV in carbon-based materials as seen in Figure 1.8.

A further improvement to the non-static model was done by Susi et. al. [25] since the Debye model does not properly describe two-dimensional systems. They extracted the out-of-plane velocities v_{z} from the phonon density of states (DOS). To estimate the DOS they calculated the phonon dispersion relation with DFT using the "frozen phonon method". This was done by individually displacing the two graphene unit cell atoms by 0.06 Å in all cartesian directions and

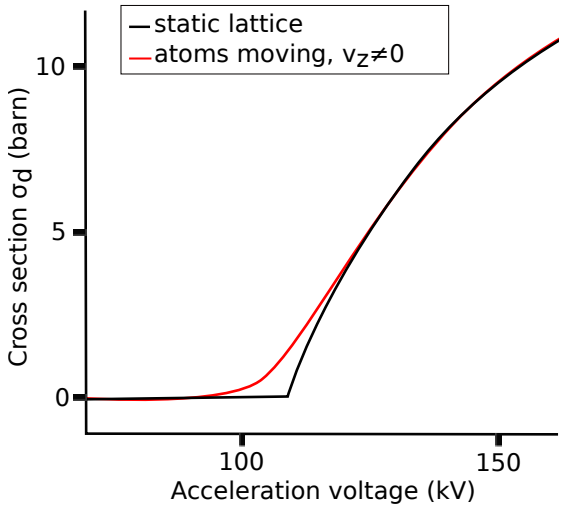

Figure 1.8: Displacement cross section for the static lattice approximation

calculating the forces on all atoms in a 7x7 supercell. The Gaussian velocity distribution $P_{v_z}(v)$ was calculated from the out-of-plane acoustical (ZA) and optical modes (ZO) of the DOS. The probability distribution of the target atoms in the z direction follows the normal distribution with a standard deviation equal to the temperature-dependent mean square velocity $\overline{v_z^2}(T)$:

$$P_{v_z}(v) = \frac{1}{\sqrt{2\pi\overline{v_z^2}(T)}} \exp\left(\frac{-v_z^2}{2\overline{v_z^2}(T)}\right) \tag{1.37}$$

The total cross section was then calculated by numerically integrating Equation 1.36 over all velocities v_z where the maximum transferred energy $E_{\max}(v, E_e)$ exceeds the displacement threshold E_d

$$\sigma(E, v) = \int_{E_{\max}(v,E_e)\geq E_d} P_{v_z}(v)\, \sigma_d\left(E_{\max}(v, E_e)\right) \mathrm{d}v. \tag{1.38}$$

They compared the model with experimental data acquired with a STEM where the displacement was measured for several acceleration voltages (see Figure 1.9). The ^{12}C and ^{13}C curves in Figure 1.9 represent the best fit to the experimental data points ^{12}C STEM and ^{13}C STEM, respectively, measured for graphene. Despite the improvement to the theoretical value, there remains a possible discrepancy between the experimental fit and the theoretical curve calculated with displacement threshold energies E_d and velocities v_z from DFT simulations.

In this thesis the model is further expanded by including the temperature effect not just on the

momentum transfer, but also on the displacement threshold and compared to the data measured in [25]. In order to predict the role of the temperature, carbon displacement simulations were carried out for graphene at various temperatures using molecular dynamics.

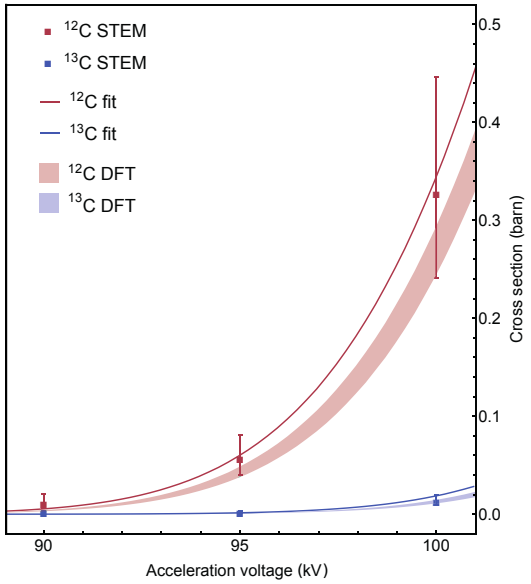

Figure 1.9: Experimental (STEM) vs. theoretical displacement cross-section (DFT). Adapted from Ref. [25]

1.5 Molecular dynamics

"*At the very heart of any molecular dynamics scheme is the question of how to describe - that is in practice how to approximate - the interatomic interactions*", Dominik Marx [31].

Molecular dynamics (MD) simulations are carried out in the hope of understanding the dynamic behavior of molecules in terms of their structure and the microscopic interaction between their atoms. MD acts as a link between experiments and theory, the macroscopic world and the microscopic length and time scales, providing us the means to learn some things, which cannot be found in other ways. To make direct comparisons with experimental data, a good model of molecular interactions is vital. The accuracy of the predictions they provide is subject to the limitations imposed by computational power.

There are two major types of simulation methods for many-body systems, stochastic and deterministic simulations, which are covered by the Monte Carlo (MC) [32] and the MD, respectively. Monte Carlo simulations test the configuration space by trial moves of particles. Within the so-called Metropolis algorithm [33], the energy change from step n to $n+1$ is used as a

trigger to accept or reject the new configuration [34],[35]. Low energy paths are always accepted, while those with higher energy are accepted within a probability range governed by Boltzmann statistics [34],[35]. By averaging over all the Monte Carlo steps, the properties of the system can be calculated.

We can distinguish between three types of MD in terms of amount of empirical input and accuracy:

- Classical,
- Semi-ab-initio and
- Ab-initio molecular dynamics

In the coming sections, the differences between those types of simulations will be discussed. For now we will concentrate on what they all have in common. All of the above simulations consist of the step-by-step numerical solution of classical equations of motion [36] [37], which are in the Newton formalism

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i \quad and \quad \mathbf{f}_i = -\frac{\partial}{\partial \mathbf{r}_i} U\left(\mathbf{r}^N\right), \tag{1.39}$$

where $\mathbf{f}_i$ is the force acting on atom i, which is derived from the potential energy $U\left(\mathbf{r}^N\right)$, where $\mathbf{r}^N = (\mathbf{r}_1,\ \mathbf{r}_2, ...\mathbf{r}_N)$ represents the complete set of $3N$ atomic coordinates.

For classical MD and semi-ab-initio calculations, the part of the potential energy U for the interactions between atoms in the same molecule and those in other molecules is usually split into one-body, two-body, three-body terms and has the form:

$$U(\mathbf{r}^N) = \sum_i u\left(\mathbf{r}_i\right) + \sum_i \sum_{j>i} v\left(\mathbf{r}_i, \mathbf{r}_j\right) + \sum_i \sum_{j,k>i} v\left(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k\right) + ... \tag{1.40}$$

The term $u\left(\mathbf{r}\right)$ represents an externally applied potential field. It is common to focus on the pair potential $v\left(\mathbf{r}_i, \mathbf{r}_j\right)$ or the atomic bond order potential described in section 2.1. The most commonly used form of pair potential, is the Lennard-Jones two-body potential

$$v^{LJ} = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]. \tag{1.41}$$

It is used to describe the interaction between a pair of neutral atoms or molecules. One parameter, ϵ, is the depth of the potential well, which describes the region surrounding the lowest potential energy point. σ describes the distance where the particle-particle interaction force is null, while r is the distance between the particles. The $\propto r^{-12}$ term stands for the repulsion at short

ranges between the particles, caused by the overlapping of electron orbitals [38]. The $\propto r^{-6}$ term accounts for the the long range attraction. This potential was used in the earliest studies and gave a very good approximation on the equilibrium state of argon [39]. Having defined the potential energy function $U\left(\mathbf{r}^N\right)$, the atomic forces can be calculated as

$$\mathbf{f}_i = -\frac{\partial}{\partial \mathbf{r}_i} U\left(\mathbf{r}^N\right). \tag{1.42}$$

For simplicity, we can describe a structure composed of atoms with potential energy $U\left(\mathbf{r}^N\right)$ and coordinates $\mathbf{r}^N = (\mathbf{r}_1,\ \mathbf{r}_2, ...\mathbf{r}_N)$ also in terms of kinetic energy $K\left(\mathbf{p}^N\right) = \sum_{i=1}^{N} \mid \mathbf{p}_i \mid^2 /2m_i$ with atomic momenta $\mathbf{p}^N = (\mathbf{p}_1,\ \mathbf{p}_2, ...\mathbf{p}_N)$. The total energy or the hamiltonian can be written as the sum of the kinetic and potential energy $H = K + U$. Now if we write the classical equations of motion as

$$\dot{\mathbf{r}_i} = \dot{\mathbf{p}_i}/m_i \quad \text{and} \quad \dot{\mathbf{p}_i} = \mathbf{f}_i, \tag{1.43}$$

we get a system of coupled ordinary differential equations. These equations need to be integrated step-by-step numerically by the MD algorithm. The integration time step δt must not be too large, since it may produce inaccurate results. A good example of an algorithm which allows the use of longer timesteps without putting the accuracy of the simulations at risk is the 'velocity Verlet' algorithm [40], which can be written as:

$$\mathbf{p}_i\left(t + \frac{1}{2}\delta t\right) = \mathbf{p}_i(t) + \frac{1}{2}\delta t \mathbf{f}_i(t) \tag{1.44}$$

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i + \delta t \mathbf{p}_i\left(t + \frac{1}{2}\delta t\right)/m_i \tag{1.45}$$

$$\mathbf{p}_i(t + \delta t) = \mathbf{p}_i\left(t + \frac{1}{2}\delta t\right) + \frac{1}{2}\delta t \mathbf{f}_i(t + \delta t). \tag{1.46}$$

After the step of Equation 1.45, the force is evaluated to give $\mathbf{f}_i(t + \delta t)$ for Equation 1.46. This construction advances the coordinates and momenta over a timestep δt. The main attributes of the Verlet algorithm are that it is time reversible; it requires just one force evaluation per step and it is easy to program. Concerning MD, the main ingredients needed to start are:

- a model to describe the interaction between atoms, molecules etc.
- a integrator which shifts particle positions and velocities from time t to $t + \delta t$,
- a microcanonical ensemble (NVE).

These three essential quantities define a MD simulation. In this study, classical molecular dynamics and semi-ab-initio simulations have been carried out using the 'velocity Verlet' algorithm. In the next sections, the main aspects of classical molecular dynamics and ab-initio calculations will be discussed including the methods used in this master thesis.

1.5.1 Classical molecular dynamics

MD is a well-established, powerful tool used to investigate many-body condensed matter systems and biomolecular congregations. It uses predefined potentials and force fields based on empirical data or on independent electronic structure calculations. In classical molecular dynamics, the interatomic potentials are determined in advance. Particles will interact according to the predefined potentials.

Since classical MD uses simple predefined potentials, they are not computationally expensive and can be used to simulate a large number of atoms. The atoms interaction is mediated by analytical potentials, solving Newton's equation of motion for each particle numerically and therefore yielding an approximation of the system's time dependent behavior. The potential of choice in this study is the second generation reactive empirical bond order potential energy expression for hydrocarbons (REBO2) by Brenner et. al. [41]. We also tested analytical potentials by Erhart and Albe [42], Tersoff [43], and Brenner [44]. However, it was the REBO2 potential (see section 2.1) which described best the displacement threshold of carbon atoms from the graphene lattice when compared to experimental data.

1.5.2 Ab-initio molecular dynamics

"Ab-initio" methods, in latin "from the beginning" are purely theoretical ways to describe interactions between atoms. They are based on various approximations, which are necessary in order to solve Schrödinger's equation for many-body systems.

In a real system, there are a series of electrostatic interactions between particles, such as electron-nuclei attraction, electron-electron repulsion and nuclei-nuclei repulsion. The time-independent Schrödinger equation does fur such a system depend on the positions of all electrons r_i and of all nuclei R_i

$$H\Psi\left(r_i, R_i\right) = E\Psi\left(r_i, R_i\right), \tag{1.47}$$

where E is the total energy of the system and Ψ is the wavefunction representing all the particles

of the system. This equation cannot generally be solved. One can assume that the nuclei and the electrons can be treated separately. The validity of this assumption is based on the mass difference between the nuclei and the electrons, one proton being $\approx$ 1837 times heavier than one electron. This is called the Born-Oppenheimer approximation

$$\Psi\left(r_i, R_i\right) \approx x\left(R_i\right) \Psi\left(r_i\right). \tag{1.48}$$

The nuclei may be treated separately $x\left(R_i\right)$ and as standing particles, given the fact that they are that much heavier electrons. However, this simplification $\Psi\left(r_i\right)$ can still not be solved analytically because of the correlation and exchange effects (Pauli exclusion principle and Coulomb interaction) between the electrons. The solution is to treat each electron separately as moving in a mean field potential $V\left(r\right)$. So now we have N one-electron wave functions interacting with the same mean field potential $V\left(r\right)$

$$-\frac{\hbar^2}{2m}\nabla^2\Psi_i\left(r\right) + V\left(r\right)\Psi_i\left(r\right) = \epsilon_i\Psi_i\left(r\right), \tag{1.49}$$

where $\Psi_i\left(r\right)$ are the one-electron wave functions, ∇ the Nabla operator, $\hbar$ the reduced Planck constant, m the mass of the electron and ϵ_i the energy eigenvalues. The electrostatic interaction of one electron with all other electrons can be described by the mean field potential. The N-electron wave functions can be expressed in terms of electron density $\rho\left(r\right)$ as

$$\rho\left(r\right) = \sum_i^N \left|\phi\left(r\right)\right|^2. \tag{1.50}$$

According to Kohn and Sham [45], the one-electron Schrödinger equation of a fictional system of non-interacting electrons can generate the same electron density as any other system of interacting particles. Density functional theory (DFT) formulated by Hohenberg [46], Kohn and Sham is based on the Born-Oppenheimer and the single-electron approximation, where the Hamiltonian is

$$H = -\frac{\hbar^2}{2m}\sum_i \nabla_i^2 + \sum_i V_{\text{ext}}\left(r_i\right) + \frac{1}{2}\sum_{i\neq j}\frac{e^2}{\left|r_i - r_j\right|}. \tag{1.51}$$

The fundamental proposition of DFT is that the ground state and excited state properties of a system with many electrons, interacting through an external potential V_{ext}, can be determined by the electron density $\rho\left(r\right)$. The Kohn-Sham DFT theory describes a system in which the energy is expressed as a functional of the ground state density $E\left[\rho\right]$.

Chapter 2

Methods

2.1 A second-generation reactive empirical bond order potential (REBO2)

The reactive empirical bond order potential [44] is a classical potential energy expression for carbon and hydrocarbon molecules that allows for bond making and breaking with appropiate changes in atomic hybridization. REBO2 [41] yields an improved version of describing bond energies, lengths, and force constants for carbon-carbon bonds when compared to its predecessor. The basis of the potential described here is based on the general analytical form derived by Abell from the chemical pseudopotential theory [47]. This form is based on a parametrized bond order function, which is used to introduce many-body effects and chemical bonding into a pair potential. Abell showed that the chemical binding energy E_{b} can be simply written as a sum over atomic sites *i*

$$E_b = \sum_i \sum_{j(>i)} \left[V^R\left(r_{ij}\right) - b_{ij} V^A\left(r_{ij}\right) \right]. \tag{2.1}$$

The sum in Equation 2.1 is over the nearest neighbors *j* of atom *i*, the functions $V^R\left(r_{ij}\right)$ and $V^A\left(r_{ij}\right)$ are pair-additive interactions that represent interatomic repulsions (core-core) and attraction from valence electrons, respectively. The term r_{ij} is the distance between pairs of nearest-neighbor atoms i and j, and b_{ij} is the bond order [41], the number of chemical bonds between atoms i and j. The empirical bond order function used here is written as

$$b_{ij} = \frac{1}{2}\left[b_{ij}^{\sigma-\pi} + b_{jj}^{\sigma-\pi} \right] + b_{ij}^{\pi}. \tag{2.2}$$

Values for functions $b_{ij}^{\sigma-\pi}$ and $b_{jj}^{\sigma-\pi}$ depend on the positions and bond angles for atoms i and j. The function b_{ij}^{π} is written as a sum of two terms:

$$b_{ij}^{\pi} = \Pi_{ij}^{RC} + b_{ij}^{DH}. \tag{2.3}$$

The value of Π_{ij} depends on whether a bond between atoms i and j has a radical character or not [48]. The term b_{ij}^{DH} depends on the dihedral angle (angle between two intersecting planes of atoms) for carbon-carbon double bonds. Equation 2.2 combined with Equation 2.1 are used to define the covalent binding energy of any collection of carbon atoms or hydrogen.

In the second-generation potential (REBO2), the forms:

$$V^R(r) = f^c(r)\left(1 + Q/r\right) Ae^{-\alpha r} \tag{2.4}$$

and

$$V^A(r) = f^c \sum_{n=1,3} B_n e^{-\beta_n r} \tag{2.5}$$

are used for the pair terms [41]. The distance between the atoms is r, and $f^c(r)$ limits the range of the covalent interaction by assuming a value of one for nearest neighbours and zero for all the other distances to the atoms. The repulsive pair interaction is given by the screened Coulomb function Q and the attractive term by the bond order B_n.

2.2 Density functional tight binding (DFTB)

Classical interatomic potentials based on empirical data provided by experiments or ab-initio calculations can be very fast and can give an accurate description of materials where the physical properties are well understood. However, they neglect quantum mechanical effects and often fail to describe geometries which are not included in their construction. On the other hand, ab-initio calculations based on DFT represent a reliable point of reference against the experiments, but too slow to describe large systems (some hundreds of atoms).

The density functional tight-binding (DFTB) method is based on the density functional theory formulated by Kohn and Sham (KS-DFT) [49]. The DFTB method is less empirical than classical MD since the parametrization is based on a number of DFT calculations and the electron correlation is included from the beginning. It consists of a series of models that are derived from a Taylor series expansion of the KS-DFT total energy $E[\rho]$ [49]. A reference density ρ^0 is assumed and perturbed with some fluctuations $\rho(r) = \rho^0(r) + \delta\rho(r)$ instead of finding the

electron density $\rho(r)$ which minimizes the energy. The total energy is expanded in a Taylor series up to the third order [49] and can be written as

$$E\left[\rho^0+\delta\rho\right]=E^0\left[\rho_0\right]+E^1\left[\rho_0,\delta\rho\right]+E^2\left[\rho_0,(\delta\rho)^2\right]+E^3\left[\rho_0,(\delta\rho)^3\right]. \tag{2.6}$$

Along the years, different models have been built depending on the terms of this expansion. The first two terms of Equation 2.6, $E^0\left[\rho_0\right]$ and $E^1\left[\rho_0,\delta\rho\right]$ contribute to the non-self-consistent DFTB1 method [50, 51]. For systems where the charge transfer between atoms is small, or for atoms of similar electronegativity, DFTB1 is very well suited because the higher order terms are neglectable. The addition of the terms E^2 and E^3 contributes to the self-consistent charge (SCC) DFTB2 [52] and DFTB3 [53] methods respectively.

In the DFTB approach (see Ref. [54]) every one-electron wavefunction $\psi(r)$ is expressed as a linear combination of atomic orbitals ϕ_μ and their coefficients c_μ

$$\psi(r)=\sum_\mu c_\mu\phi_\mu(r). \tag{2.7}$$

The main goal is to solve the eigenproblem

$$\sum_\nu H^\sigma_{\mu\nu}c^\sigma_\nu=\epsilon^\sigma_\mu\sum_\nu, S_{\mu\nu}c^\sigma_\nu \tag{2.8}$$

where c^σ_ν represents the molecular orbital coefficient, ϵ^σ_μ is the energy eigenvalue, and

$$H^\sigma_{\mu\nu}=\langle\phi_\mu|\tilde{H}^\sigma|\phi_\nu\rangle \quad \text{and} \quad S_{\mu\nu}=\langle\phi_\mu|\phi_\nu\rangle \tag{2.9}$$

are the Hamiltonian and the overlap matrices, respectively. The spin state is indicated by the index σ. The Hamiltonian matrix element $H^\sigma_{\mu\nu}$ for the spin-polarized self-consistent charge (SCC), with orbitals ϕ_μ and ϕ_ν located on atoms A and B, respectively, can be written as

$$\begin{aligned}H^\sigma_{\mu\nu}=&\langle\phi_\mu|\tilde{H}_0|\phi_\nu\rangle+\frac{1}{2}S_{\mu\nu}\sum_C\sum_{l''\in C}\left(\gamma_{Al,Cl''}+\gamma_{Bl',Cl''}\right)\Delta q_{Cl''}\\&\pm\frac{1}{2}S_{\mu\nu}\left(\sum_{l''\in A}W_{All''}m_{Al''}+\sum_{l''\in B}W_{Bl'l''}m_{Bl''}\right).\end{aligned} \tag{2.10}$$

The first term of the matrix corresponds to the non-SCC DFTB1 method. The second term is the SCC contribution, where the sum runs over all the shells l'' of all atoms C in the system [54].

The shift vector, $\gamma_{Al,Cl''}\Delta q_{Cl''}$, which contains the SCC potential at site Al, can be constructed. γ consists of a long-range Coulombic term and a short-range term S

$$\gamma_{Al,Cl''} = \frac{1}{R_{AC}} - S\left(U_{A,l}, U_{C,l''}, R_{AC}\right). \tag{2.11}$$

R_{AC} gives the distance between atoms A and C and $U_{A,l}$ and $U_{C,l''}$ are the Hubbard parameters for the atoms and shells [54]. The Hubbard model [55] is used to describe the hopping of electrons between atoms in the tight-binding model and includes the on-site repulsion, which comes from the Coulombic repulsion between electrons on the same orbital. The charge difference $\Delta q_{Cl''}$ represents the difference between the sum of the Mulliken charges on shell l of atom C and the total charge on that shell in an isolated atom [54]. Mulliken charges are derived from the Mulliken population analysis [56] and provide the means to estimate partial atomic charges. The third term of Equation 2.10 contains the spin contributions.

DFTB requires two sorts of parameters. First, there are the electronic parameters regarding the wave function and density confinement, which enter the first term [49] Equation 2.10, and the Hubbard parameters and its derivatives, which are computed from DFT calculations and enter the rest of the equation.

2.3 Atomic simulation environment

The atomic simulation environment (ASE) [57] is a software package written in the Python programming language with the purpose of setting up and conducting atomistic simulations. Calculations of forces, energy and other quantities can be performed. The NumPy library [58] and the syntax of Python make it possible to perform complex simulation tasks. ASE can be used as an a front-end for molecular dynamics to set up, perform, visualize, and analyze simulations.

2.3.1 Unit cell construction

The first thing to do in a atomistic simulation is to set up a cell of atoms. Atoms of different chemical species are included in the *Atoms* object. They are given Cartesian coordinates and have properties such as velocities, masses, or magnetic moments. Appendix A is an example of a Python code to build up the unit cell of graphene. Using the *Atoms* object we can create a system of two carbon atoms, set the distance between them and define the unit cell vectors. Periodic boundary conditions apply to the created unit cell to mimic the behavior of an infinite

system. This involves the copying of the system in the periodic directions, so that the system is surrounded by an exact copy of itself. The structure is built and can be saved as a .POSCAR file [59]. Using the built-in graphic user interface (GUI) of ASE one can visualize the atom cell Figure 2.1. ASE GUI can also be used to add desired atoms to the system and manually move them to different locations.

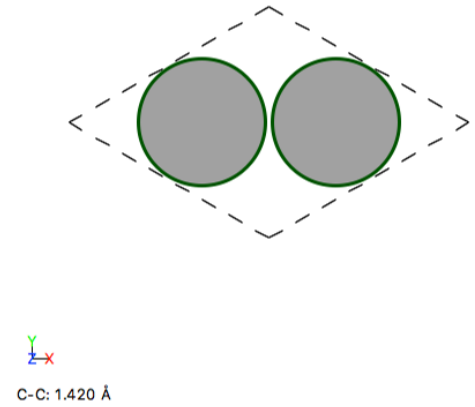

Figure 2.1: Orthorhombic graphene unit cell, as visualized with ASE GUI.

2.3.2 Unit cell relaxation

We perform a unit cell relaxation to make sure the potential energy of the system is as low as possible. The bond length between the atoms in the unit cell must correspond to the one given by the potential energy expression we use, in order to prevent strain effects. Appendix B calculates the potential energy as a function of the volume of the unit cell. The structure optimization algorithm starts from an initial guess for the atomic positions and tries to move the atoms to minimize the potential energy using an iterative procedure until certain conditions are fulfilled. The optimizer FIRE (fast inertial relaxation engine [60]) was used. It adds an artificial force term to guide the atoms towards the steepest descent of the potential energy. If the atoms move against the forces at some point, the forces are set to zero and the dynamic parameters are reset. A *Calculator* needs to be attached to the *Atoms* object in order to deliver the potential energy; in this case REBO2, this is an interatomic potential (see section 2.1). The properties that can be obtained from the *Atoms* object depend on the type of calculator. Since we are trying to find the optimal cell volume, the *Trajectory* object helps us store the *Atoms* information for each step so in the end we can compare between them and choose the unit cell with the lowest potential energy. We can use ASE GUI to select the frame where the potential energy has a minimum

(see Figure 2.2) and save the optimized structure for further use. Now that we found the optimal unit cell size in terms of the lowest potential energy, we can multiply the unit cell and build a large cell of atoms and start thermalizing the structure.

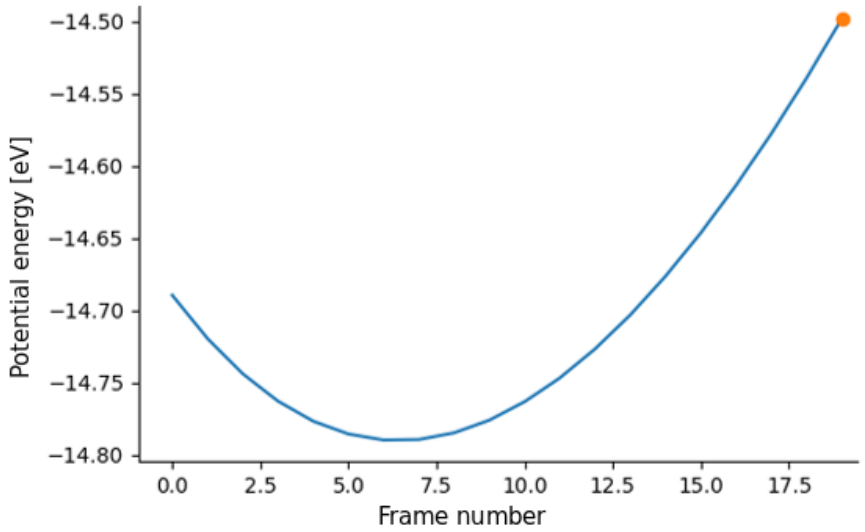

Figure 2.2: Frames having different potential energies. Each frame corresponds to a different cell volume.

2.3.3 Thermalization

In order to simulate events occurring at different temperatures, the structure needs to be thermalized. Therefore we need a molecular dynamics object that will move the atoms according to their forces and integrate the second law of Newton numerically (see section 1.5). VelocityVerlet is a suitable algorithm to describe dynamics in a NVE ensemble, where the Newton's second law is preserved and therefore also the number of atoms and volume of the system. For the velocity distribution of the atoms we use the Maxwell-Boltzmann distribution. Appendix C is an example of a code used to thermalize graphene at 300K. The reason we chose the temperature to be 600 K has to do with the fact that half of the thermal energy will go into the potential energy of the atoms and the other half into their kinetic energy; in order to simulate 300 K we had to initialize at 600 K. This time we also set up a DFTB *Calculator* to describe the interactions between the atoms in the lattice for that particular temperature. After a number of timesteps the temperature stabilizes (see Figure 2.3) and we can save the thermalized state (see Figure 2.4) as a trajectory file. This trajectory file will contain information on the positions, velocities and forces of all the atoms in the structure.

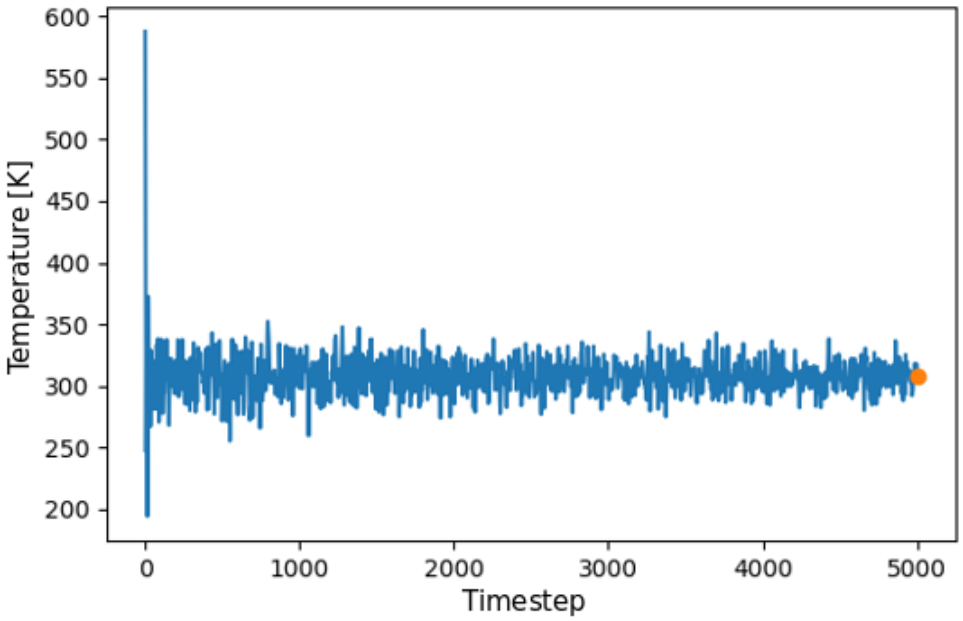

Figure 2.3: Temperature of the system against the number of timesteps ($\delta t = 1$ fs).

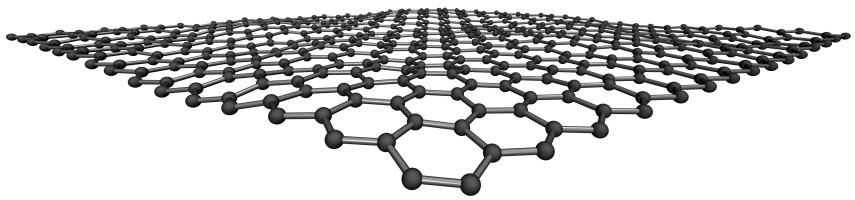

Figure 2.4: Thermalized graphene structure of 450 atoms at 300 K.

2.3.4 Displacement simulations

The thermalized graphene structure can now be used to simulate STEM experiments, that is, electrons scattering elastically from the carbon atoms in the lattice and leading to their displacement. We will assume that the electron beam is perpendicular to the graphene plane, and will transfer the carbon atoms the same amount of momentum they would receive from the electrons experimentally.

The code in Appendix E first loads a thermalized graphene structure. It then defines a list of atoms that we want to displace. A loop will start over the desired atoms, finding the displacement threshold using a DFTB calculator for each one in a given range of energy, 18.7 to 25 eV in steps of 0.1 eV. This runs until the displacement criterium is fulfilled: whether the atom is 5 Å above the lattice or its velocity component starts pointing in the negative z-direction. If the atom is not displaced, the trajectory is reloaded to the initial thermalized state and the same procedure is repeated with a higher energy value until the atom gets displaced. Figure 2.5 depicts the displacement simulation at two different timesteps, t_1 and t_2, towards the beginning of the displacement and towards the end of the displacement respectively. These simulations were performed for both the interatomic potential REBO2 and the DFTB method.

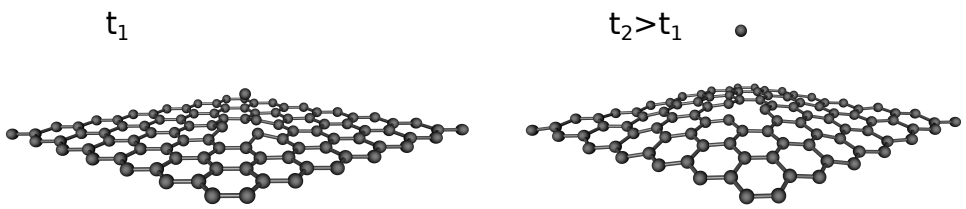

Figure 2.5: Displacing a carbon atom. Visualization of the simulation at two different timesteps.

Chapter 3

Results

3.1 Displacement simulations

3.1.1 REBO2

We used ASE and ran the script in Appendix D with Python using the methodology described in the previous section to displace carbon atoms. We took this approach to understand how the displacement threshold of the carbon atoms is affected at different temperatures. The simulations ran for a 800 carbon atom cell thermalized at 100, 200 and 300 K respectively. We decided upon a cell of 800 atoms because it is the largest system we can treat effectively and it is large enough to not suffer from boundary effects. Simulations on large atom cells would become computationally too expensive. The first round of simulations at 300 K (Figure 3.1), already proved that the spread in displacement thresholds E_{d} is quite large, going up $+6$ eV and only -2 eV below the 0 K displacement threshold (white bar), calculated to be 25.1 eV. We went to lower temperatures to check how the distribution of displacement energies changes relative to the 300 K case and whether or not the spread is similarly shifted towards higher E_{d} values (see Figure 3.2 and Figure 3.3). Neither the 200 K nor the 100 K case gave any dramatic change in the direction of the energy spread, which looks similarly shifted towards higher E_{d} values.

It is of importance to understand why the distribution of E_{d}'s is shifted to higher values and why an atom is easier or harder to eject, to find out the energy range one can operate an electron microscope with the purpose of displacing atoms. We plotted several atom parameters against the displacement value such as: in-plane velocity, out-of-plane velocity, z-position relative to the neighbors, forces in-plane and out-of-plane. None of these resulted in any trend when compared

to the E_{d}. To check whether the position of the atoms in the lattice relative to each other play a role or not, we looked at the graphene sheet at 0 K, chose an atom and changed its position relative to one neighbor. We moved the atom in small increments towards and away from it and ran displacement simulations to check how the E_{d} is affected (Figure 3.4). Regardless of the displacement direction, the displacement threshold always increased, which explains why the distribution was skewed towards higher E_{d}'s.

The interatomic potential REBO2 has given us a displacement threshold value of $E_{\mathrm{d}} = 25.1$ eV for 0 K (represented by the white line in Figure 3.1), which is quite far from DFT values [25] of around 22 eV. When running displacement simulations on the thermalized structures, the spread of the E_{d} values was large and unphysically shifted towards higher energy values. One would expect a symmetrical distribution of the E_{d} values, since the contributions to the positions and velocities of the atoms are equally distributed. Due to the obvious problems with REBO2 results, we decided to check the results with a more accurate modeling method.

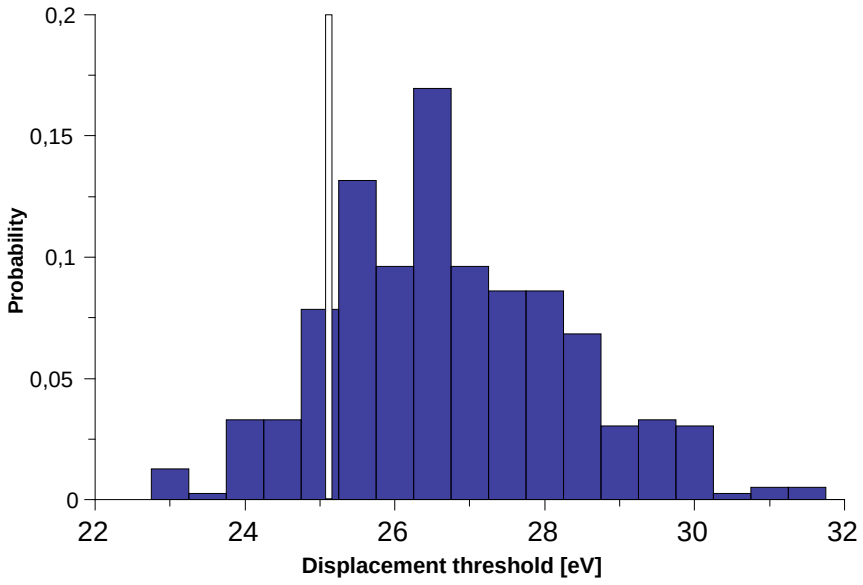

Figure 3.1: Histogram showing the spread of E_{d} values at 300 K with REBO2. The white line represents the E_{d} for 0 K.

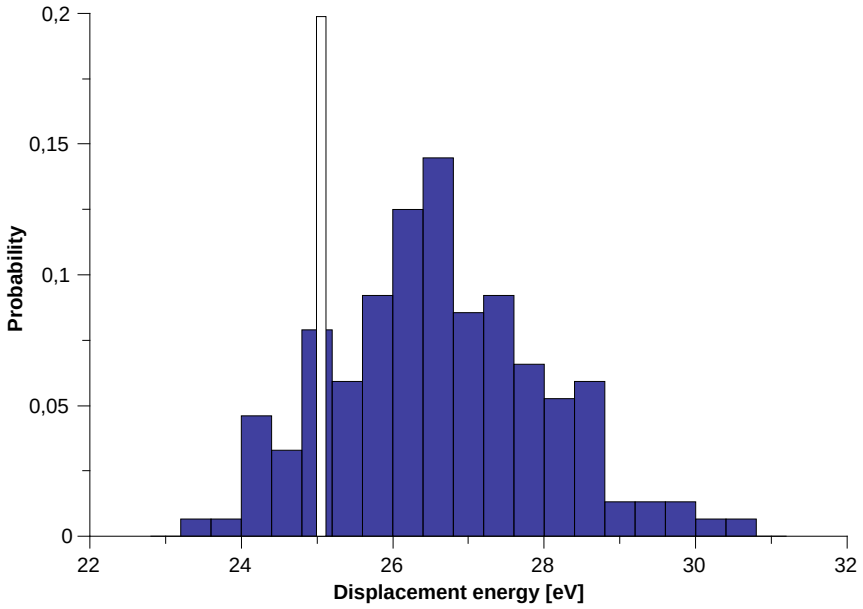

Figure 3.2: Histogram showing the spread of E_{d} values at 200 K with REBO2. The white line represents the E_{d} for 0 K.

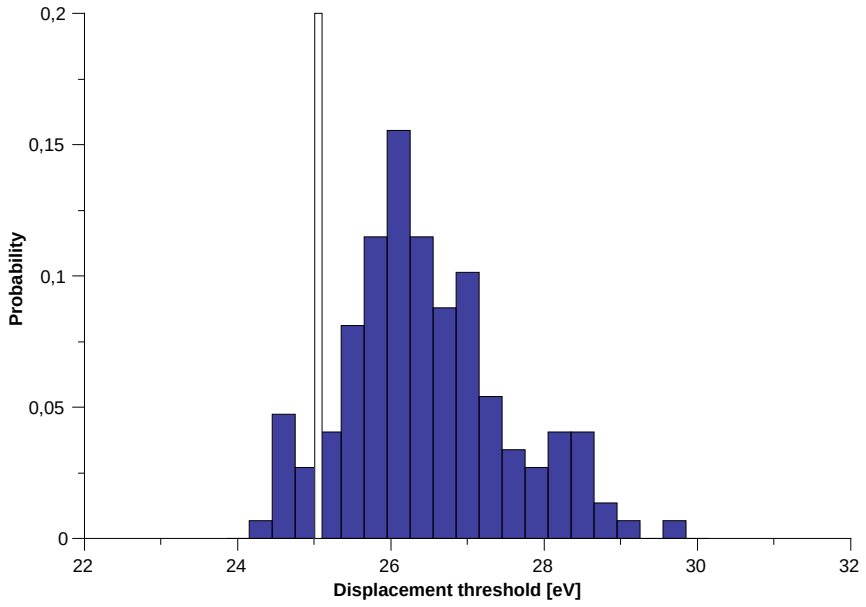

Figure 3.3: Histogram showing the spread of E_{d} values at 100 K with REBO2. The white line represents the E_{d} for 0 K.

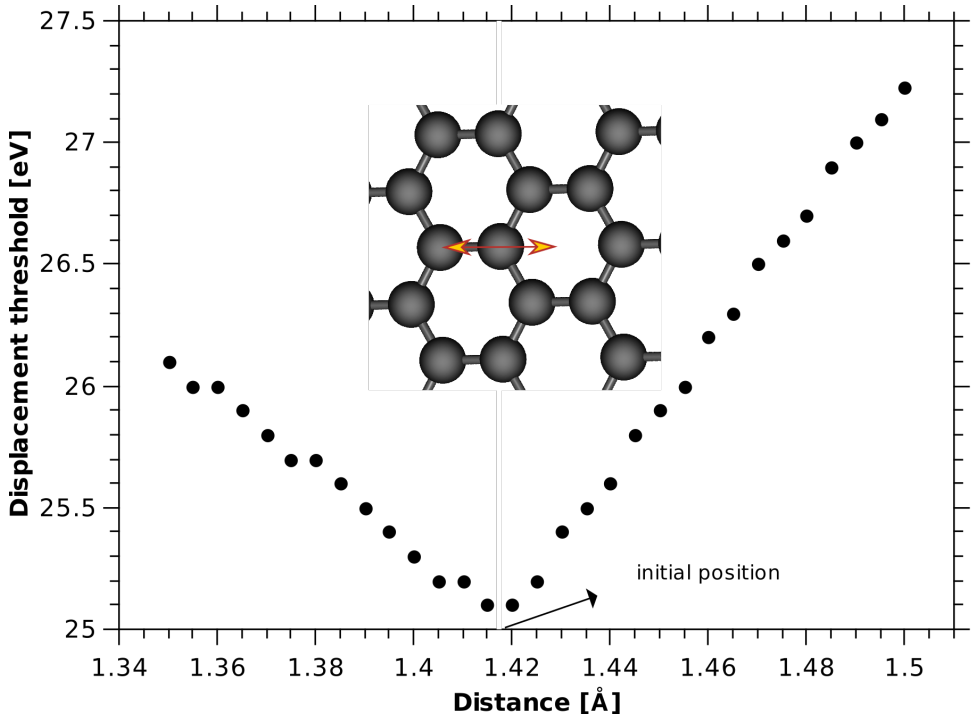

Figure 3.4: Carbon atom moved from its initial position towards a neighbor or towards the middle of the hexagon

3.1.2 DFTB

We ran another set of displacement simulations using the DFTB method. Again, we used ASE and ran the simulations using the Python code in Appendix E. Since the DFTB approach is computationally more expensive than REBO2, we thermalized a graphene cell with only 450 atoms at 75, 150, 225, 300, 350 and 450 K to find the displacement threshold.

We tested the whole cell for 150, 300 and 450 K, 400 atoms at 350 K, and only 119 displaced atoms for 75 and 225 K, respectively. Just by looking at the histograms (see Figure 3.5) we observe that the spread in the displacement threshold E_{d} is narrow at low temperatures with values between 19 and 21 eV, and gets wider to a range of 18 to 22 eV as we go towards higher temperatures. The displacement threshold for graphene at 0 K was calculated at 19.9 eV with the DFTB method, and the spread of E_{d} is symmetric, unlike in the case of the classical potential REBO2. We fitted a Gaussian function $P\left(E_{\mathrm{d}}\right)$ to the histograms in Figure 3.5 (black curves) to describe the probability distribution of the displacement thresholds which has the following form:

$$P\left(E_{\mathrm{d}}\right)=\frac{\sqrt{2}}{\sqrt{\pi}w}e^{-2\left(\frac{E_{\mathrm{d}}-x_{c}}{w}\right)^{2}}, \tag{3.1}$$

where w is the width of the distribution, x_c is the center of the distribution and E_{d} the displacement threshold. In Figure 3.6 one can see that the distribution widths vary between 0.82 and 1.53 eV depending on the temperature.

We tried again to figure out how the displacement threshold is affected by the position of the atoms in the lattice, and did again the experiment at 0 K where we moved one carbon atom towards its next nearest neighbor and away from it. While testing different positions, we noticed that at some point the E_{d} starts to decrease or increase depending on the direction of the displacement from the initial position. However, the magnitude of the change in E_{d} was very small ($\sim$ 0.2 eV).

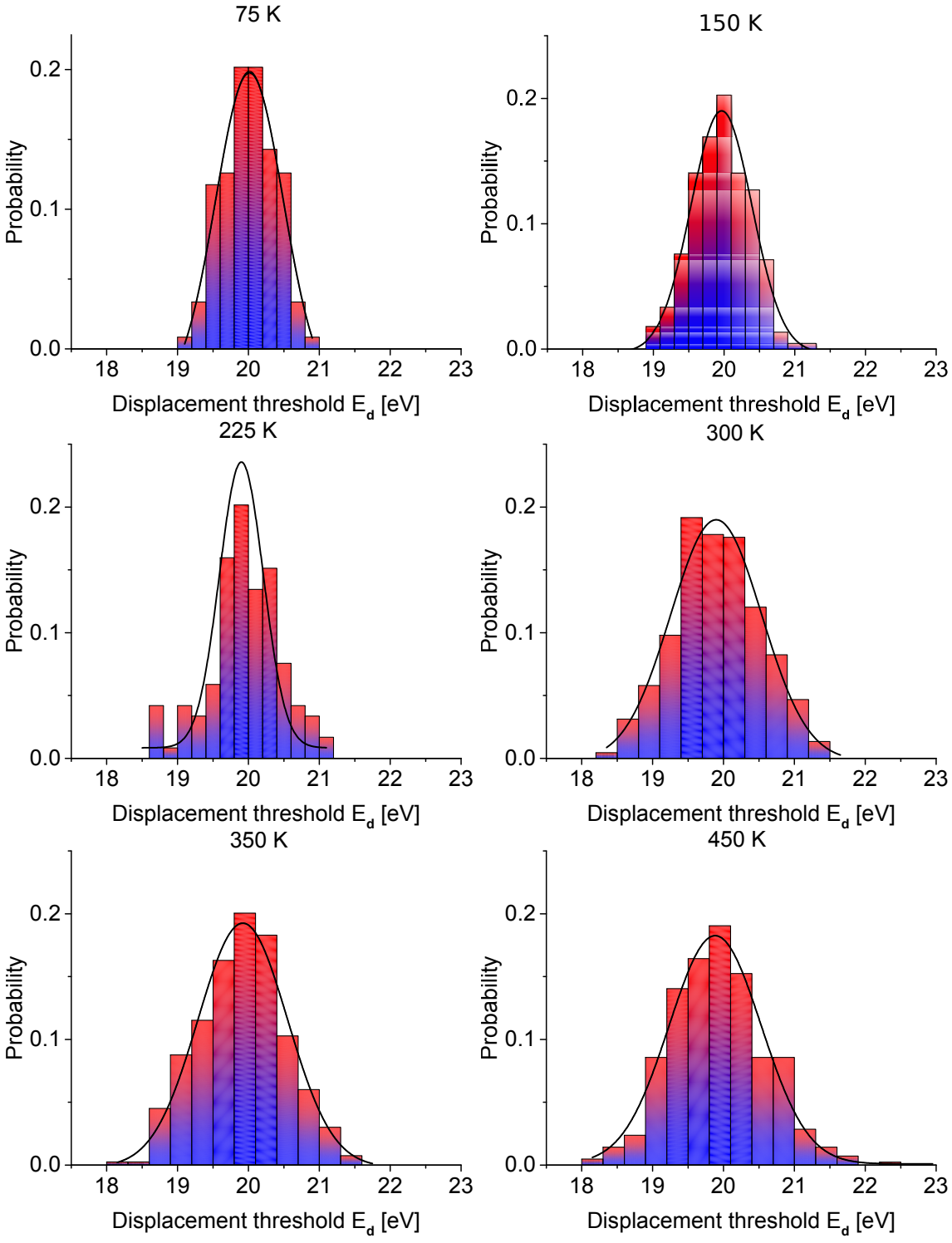

Figure 3.5: DFTB histograms for the spread of the displacement threshold for the carbon atoms.

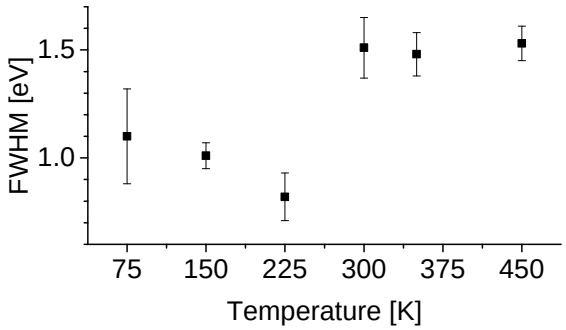

Figure 3.6: Width of the distribution extracted from the Gaussian fits to the histograms in Figure 3.5.

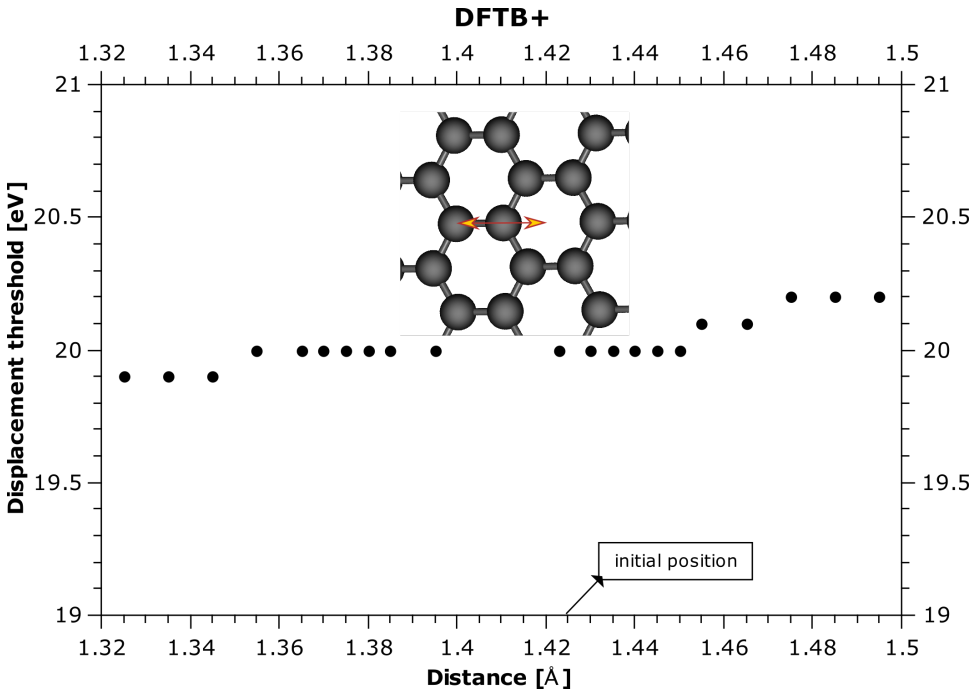

Figure 3.7: Carbon atom moved from its initial position towards the nearest-neighbor and towards the middle of the cell.

We pursued extracting information on the atoms velocities, positions, and forces and compare them to their displacement threshold. For example, we expected to see a lower E_{d} for atoms with a higher velocity component v_z, but that was not the case. Plotting the displacement threshold against the z-position of the atoms and doing a linear fit also did not give a clear trend (see Figure 3.8). Even though at 300 and 450 K, it looked like it might play a role as suggested by the the very low p value, which is a parameter that tests the null hypotheses, the effect couldn't be confirmed for 150 K. Therefore we took the 0 K lattice and carried out

displacement simulations on a single atom with different intial z-positions. The results are shown in table Table 3.1.

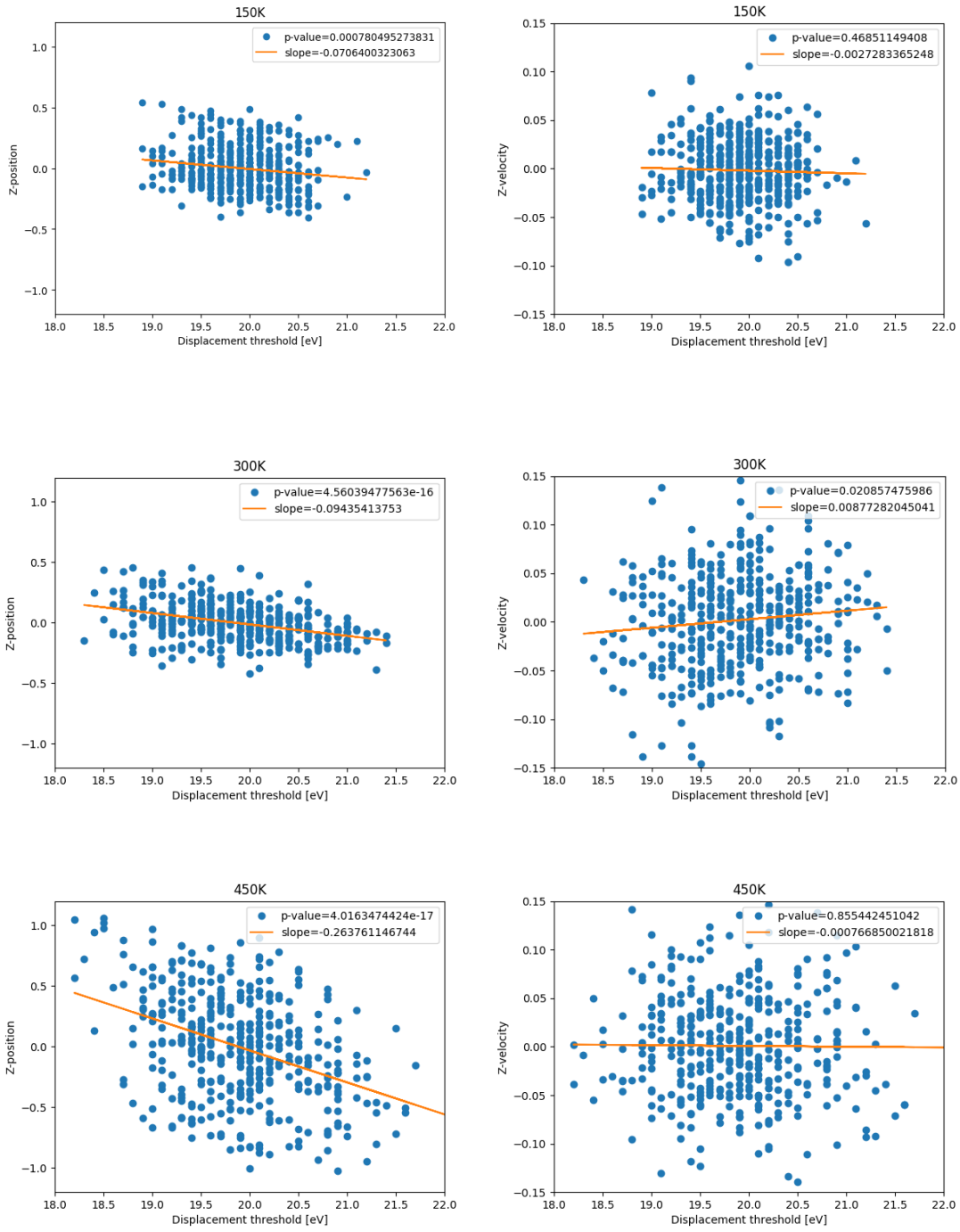

Figure 3.8: Distribution of displacement threshold E_{d} values at different temperatures as a function of their z-position (left) and out-of-plane velocity component v_z (right). The orange lines represent the linear function fitted to the values. The p-value indicates whether or not the results are significant.

Despite the fact that the atom had a higher or lower z-position with respect to its neighbors (see Figure 3.9), the atom only got ejected easier. Still there was no explanation to why we need to transfer more energy to the atom in some cases. Since changing the initial conditions of the atom alone failed to give us an explanation on the higher E_{d} values we went on looking at the neighboring atoms. Next we displaced the carbon atom together with its three neighbors and got the result shown in Table 3.2:

Table 3.1: z-displacement effect on the displacement threshold

z-displ. [Å]	-0.35	-0.3	-0.25	-0.2	-0.15	-0.1	0	0.1	0.15	0.2	0.25	0.3	0.35
E_{d} [eV]	19.0	19.5	19.5	19.6	19.7	19.8	0	19.8	19.7	19.5	19.3	19.1	18.8

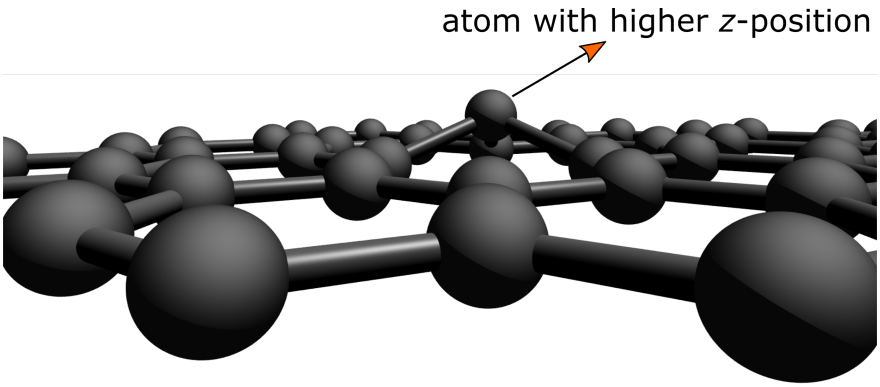

Figure 3.9: Schematic of atom lying above the rest

Table 3.2: z-displacement of the atom and its 3 neighbors

z-displacement [Å]	E_{d} [eV]
-0.2	21.7
-0.1	20.8
0	19.9
0.1	19.0
0.2	18.5

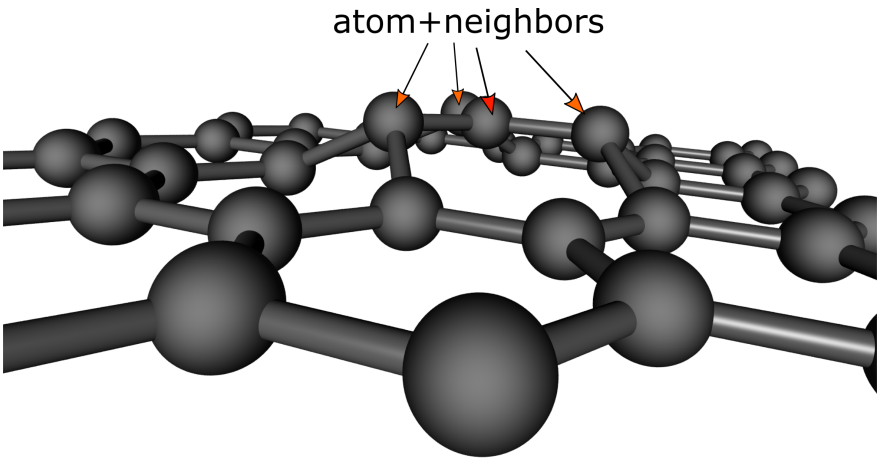

Figure 3.10: Schematic of atom and neighbors positioned above the rest

These simulations finally revealed how the displacement threshold E_{d} is affected by the geometry of the atoms in the cell. If the targeted atoms have a higher z -position relative to their neighbors, the force which pulls the atom back to the lattice is easier to repel. Therefore, the atoms need less energy to escape. To confirm the hypothesis we ran another set of simulations where the three neighbors of the ejected atom had initial positive and negative momentum.

The results show that the energy needed to displace a carbon atom is also dependent on the momentum of its neighbors. As depicted in Figure 3.11, if the ejected atom is not held back by its neighbors, it can escape easier. One has to consider that even though the momenta are pointing in the opposite direction of the ejection, the restoring forces are acting in the direction of the ejection and help the atom get knocked out with less energy.

Table 3.3: Initial momentum to the neighbors of the atom to be ejected (see Figure 3.11).

$\vec{p}$ [kg·m/s]	E_{d} [eV]
-0.15	18.3
-0.1	18.9
-0.5	19.4
0.5	20.4
0.1	20.8
0.15	21.3

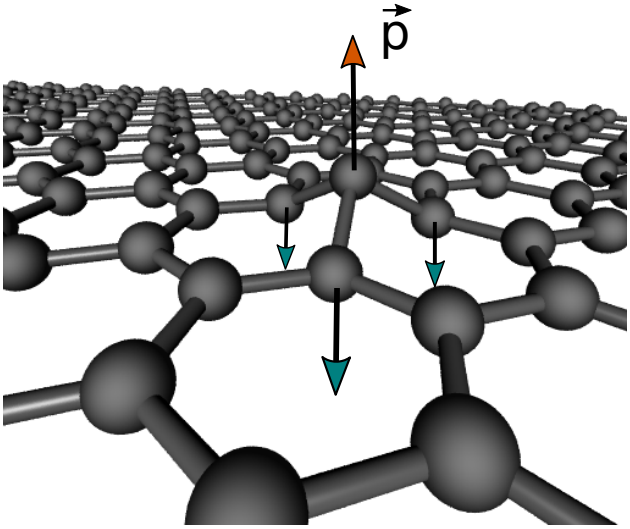

Figure 3.11: Schematic of the atom about to be ejected and its neighbors with negative momentum.

3.2 Improving the theoretical displacement cross section

As was presented in Section 1.4.1, a series of improvements have been done to the model of predicting the displacement cross section when operating TEMs at voltages below 110 keV. The static lattice approximation was replaced by the the model where the target atoms were moving [24, 25]. Equation 1.38 predicted the displacement cross section most accurately by integrating over a range of out-of-plane velocities v_z, for which the maximum transferred energy $E_{\max}$ (Equation 1.29) was exceeding the displacement threshold E_{d} of the target atom. However, one can see from the figure Figure 1.9 that there is still a possible discrepancy between the theoretical curve and the experimental fit. Until now, all formulations have assumed that E_{d} is a constant. Our simulations (Figure 3.5) clearly show that this is not an accurate approximation. The spread in the displacement threshold values E_{d} is a consequence of the fact that atoms are not at their equilibrium positions at any finite temperature.

For low temperatures (75, 150, 225 K) the distribution widths are narrow and vary between 0.82 and 1.1 eV (see Figure 3.6). The data points for 75 K and 225 K are less accurate since we only tested 119 atoms, compared to the full lattice at 150 K. At higher temperatures (300, 350 and 450 K) the values are between 1.48 and 1.53 eV. This may have to do with the fact that the vibration modes reach a saturation level around room temperature, or the DFTB method fails to describe accurately the behavior of the atoms at high temperatures.

To improve the theoretical displacement cross section, we need to also include the distribution of displacement thresholds E_{d} when calculating the total cross section with Equation 1.38. We included the distribution of displacement thresholds (see Equation 3.1) in the formula for computing the total displacement cross section

$$\sigma\left(E, v, E_{\mathrm{d}}\left(T\right)\right) = \int_{E_{\max}(v,E)\geq E_{\mathrm{d}}} P\left(E_{\mathrm{d}}\left(T\right)\right) P_{v_z}\left(v\right) \sigma_d\left(E_{\max}\left(E, v\right)\right) \mathrm{d}v \mathrm{d}E_{\mathrm{d}}. \tag{3.2}$$

To evaluate Equation 3.2 we used Sage [61], an open-source mathematical software system based on Python and built upon packages like NumPy, SciPy, matplotlib and many more. The script in Appendix F calculates the velocity distribution P_{v_z} from the phonon density of states taken from [25], and then numerically integrates Equation 3.2 over all velocities v_z where the maximum transferred energy exceeded the displacement threshold ($E_{\max} \geq E_{\mathrm{d}}$) and multiplies it with the Gaussian displacement threshold distribution, to calculate the displacement cross section.

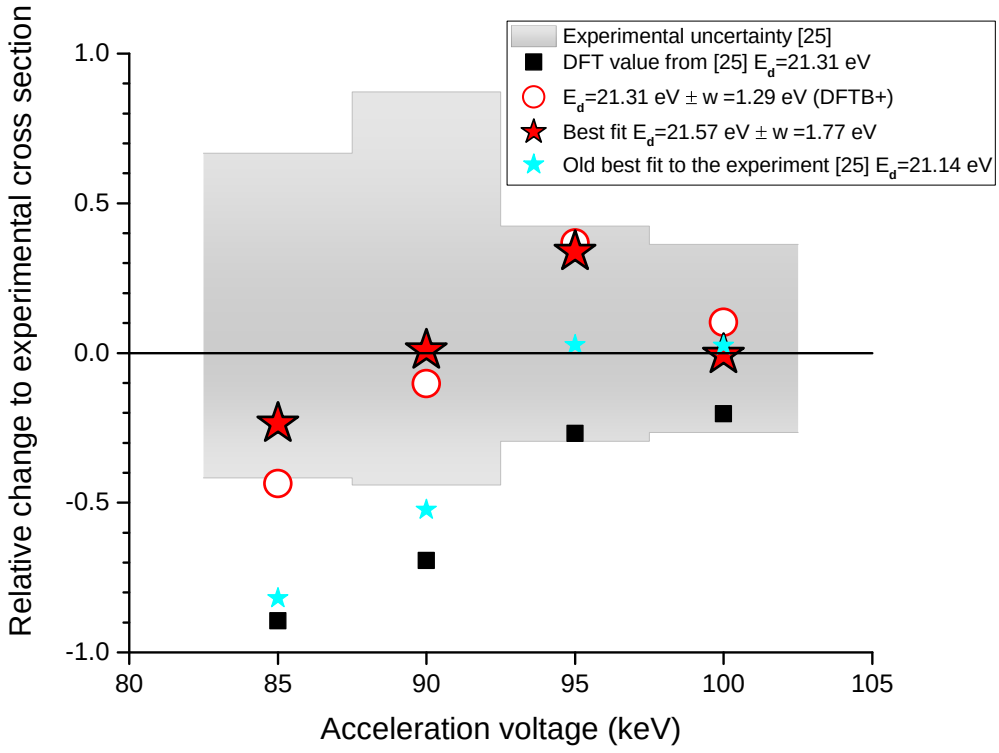

Figure 3.12: Comparison of theoretical cross section values and the STEM (shaded area) and DFT values from [25].

We computed the new displacement cross-section data and compared it with the experimental data acquired with the STEM in [25] (Figure 3.12). The relative difference was calculated as

$$\Delta = \frac{\sigma_{\mathrm{new}} - \sigma_{\mathrm{exp}}}{\sigma_{\mathrm{exp}}}. \tag{3.3}$$

The black squares represent the displacement cross sections calculated with Equation 1.38 with E_{d}=21.31 eV value from DFT. The displacement cross section calculated with Equation 3.2 for the same E_{d} value with the spread of w = 1.29 eV (from DFTB+ 300 K), is much closer to the experimental data (red circles). We also calculated the best two parameter-fit to the cross section values by changing the displacement threshold E_{d} and the spread w, and compare them to experimental data via Equation 3.3.

Chapter 4

Conclusion

In this thesis, I have shown that the displacement threshold energy E_{d} should not be assumed to have a constant value for systems at finite temperature. Instead, the threshold is affected by the fact that the atoms are displaced from their equilibrium positions, which leads to a normal distribution of values around the 0 K value. In addition, I have algebraically derived the formula to calculate the maximum transferred energy in a relativistic scattering process between an electron and a moving nucleus, and verified the validity of typically used approximations.

From the simulations with an interatomic REBO2 potential at different temperatures, we noticed that there is a large spread in E_{d} values. We tried to figure out how the displacement threshold is affected, but couldn't reach a conclusion. The spread of displacement threshold values was noticeable for the analytical potential simulations, but the location of the distribution in terms of E_{d} was not physical. Therefore, we moved on to DFTB calculations for a more accurate description of the displacements.

The DFTB simulations gave us a symmetric spread of energies in contrast to the REBO2 method. We discovered that the displacement threshold of the target atoms is strongly influenced by its neighboring atoms. If the neighboring atoms have a initial momentum opposite to the displacement direction of the target atom, the restoring force acting on the atom will be smaller, and so it will be easier to knock out. On the contrary, if they have momentum in the same direction as the target atom, the restoring force will be larger, and the atom will require more energy to be knocked out.

It is typically assumed that the displacement threshold has a fixed value when the total cross section is evaluated. However, our simulations show that this is an approximation. We improved the model for evaluating the theoretical displacement cross section by including the

effect the temperature has on the displacement thresholds. If we compare the old theoretical values calculated for a displacement threshold of $E_{\mathrm{d}} = 21.31$ eV to the same DFT value with an added spread of $w = 1.29$ eV, there is a noticeable improvement in the match to the experimental data. A two-parameter fit yields $E_{\mathrm{d}} = 21.57$ eV, $w = 1.77$ eV, which further improves the agreement. Additional improvements, such as ab-initio molecular dynamics simulations and more precise measurements should allow us to improve the model. An accurate description of the displacements of carbons atoms from graphene would be a step forward in understanding how to also controllably manipulate heteroatoms such as silicon or nitrogen.

Appendices

A Unit cell construction

```
import numpy.math.sqrt as sqrt
from ase import Atoms, io

# add 2 Carbon atoms
a = Atoms('C2')

# nearest neighbour carbon-carbon distance
a0=1.42

#set the unit cell vectors and scaled positions
a.set_cell([[a0*3/2,a0*sqrt(3)/2,0],[a0*3/2,-a0*sqrt(3)/2,0],
[0,0,10]])
a.set_scaled_positions([[1./3,1./3,0],[2./3,2./3,0]])
io.write('graphene_unit_cell.POSCAR',a)
```

B Unit cell relaxation

```
import numpy as np
import sys
from ase import Atoms, io

# import calculator
from atomistica import Rebo2
# object that stores temporal evolution of a system
from ase.io import Trajectory
# structure optimizer
from ase.optimize import FIRE

# load unit cell
name = sys.argv[1].replace('.POSCAR','')
atoms = io.read(name+'.POSCAR')

# define the calculator
calc = Rebo2()

# attach the calculator to the atoms object
atoms.set_calculator(calc)
cell = atoms.get_cell()

# define trajectory name and properties
traj = Trajectory(name+'_cellrlx.traj','w', properties=['energy','forces'])

# loop over different cell sizes to find the minimum energy unit cell
for x in np.linspace(0.9375, 1.0125, 20):
    atoms.set_cell([cell[0] * x, cell[1] * x, cell[2]], scale_atoms=True)
    opt = FIRE(atoms)
    opt.run(fmax=0.01)
    traj.write(atoms)
```

C Thermalization

```
import os, sys
from ase import Atoms
from ase.calculators.dftb import Dftb
from ase.io import Trajectory
from ase import io

# defines distribution of velocities at certain temperatures
from ase.md.velocitydistribution import MaxwellBoltzmannDistribution

# algorithm used to calculate dynamical properties
from ase.md.verlet import VelocityVerlet
from ase import units
import numpy as np
fname = 'dftb_graphene_sc15'

# load the atoms
system = io.read('dftb_relaxed_450atoms.POSCAR')

# set the calculator
calc = Dftb(label='carbon',
            atoms=system,
            kpts=(1,1,1),
            Hamiltonian_MaxAngularMomentum_='',
            Hamiltonian_MaxAngularMomentum_C='"p"')
system.set_calculator(calc)

# choose timestep
ts = 1

# set the temperature
temp = 600
```

```
name=fname+"_MB_300Kelvin"
traj = Trajectory(name+".traj", 'w', system, properties={"energy",
"velocities", "forces"})

# apply distribution to the system
MaxwellBoltzmannDistribution(system, temp * units.kB)
# set VelocityVerlet as the integrator
dyn = VelocityVerlet(system, ts * units.fs)

# save trajectory for each timestep
dyn.attach(traj.write, interval=1)

# amount of timesteps to run
dyn.run(5000)
traj.close()
```

D REBO2 displacement simulations

```
from ase import Atoms
from atomistica import Rebo2
from ase.io import Trajectory
from ase import io
from ase.md.velocitydistribution import MaxwellBoltzmannDistribution
from ase.md.verlet import VelocityVerlet
from ase import units
import numpy as np
import sys

fname = sys.argv[1].replace('.traj','') #load file
atoms = io.read(fname+'.traj')
calc = Rebo2()
atoms.set_calculator(calc)
ts = 0.3
atoms2 = atoms.copy()

#randomize the atom number about to knock out
for i in np.random.randint(501,800, size=33):
    nat = i
    print nat

    name=fname+"_stat_at"+str(nat)
    flog=open(name+'.txt','a')
    flog.write('\n')
    atoms = atoms2.copy()
    calc = Rebo2()
    atoms.set_calculator(calc)
    dyn = VelocityVerlet(atoms, ts * units.fs, logfile='md.log')

    for e in np.arange(24.5,29.0,0.1):
```

```
        traj = Trajectory(name+"_E"+str(e)+".traj", 'w', atoms,
        properties={"energy","velocities", "forces"})
        dyn.attach(traj.write, interval=5)
        atoms[nat].momentum[2]=(2*e*atoms[nat].mass)**0.5
        dyn.run(600)

        if atoms[nat].position[2] > 6.0:
            print i, "Ejected at:", e
            flog.write(',')
            flog.write(str(e))

        atoms = atoms2.copy()
        calc = Rebo2()
        atoms.set_calculator(calc)
        dyn = VelocityVerlet(atoms, ts * units.fs)
    flog.close()
```

E DFTB+ displacement simulations

```
from ase import Atoms, io
from ase.calculators.dftb import Dftb
from ase.io import Trajectory
from ase.md.verlet import VelocityVerlet
from ase import units
import numpy as np
import sys
from ase.io.dftb import read_dftb_velocities, write_dftb_velocities

# load trajectory
fname = sys.argv[1].replace('.traj','')
system = io.read(fname+'.traj')

# choose timestep
ts = 0.3

atoms2 = system.copy()

# create the list of atoms to eject
a = np.arange(1,40,1)
b = np.arange(180,220,1)
c = np.arange(330,370,1)
indices = np.concatenate([a,b,c])

# loop over the atoms
for nat in indices:
    print nat
    name=fname+"_stat_at"

#open a .txt file to save the atom number and the displacement threshold
    flog=open(name+'.txt','a')
```

```
    system = io.read(fname+'.traj')
    flog.write(str(nat))
    system = atoms2.copy()
    calc = Dftb(label='carbon',
            atoms=system,
            kpts=(1,1,1),
            Hamiltonian_MaxAngularMomentum_='',
            Hamiltonian_MaxAngularMomentum_C='"p"',
            )
    system.set_calculator(calc)
    dyn = VelocityVerlet(system, ts * units.fs, logfile='md.log')

# loop over different maximum transferred energies to the atom
    for e in np.arange(18.7,25,0.1):
        traj = Trajectory(name+"_E"+str(e)+"_"+str(nat)+".traj", 'w', system,
        properties={"energy", "velocities", "forces"})

# saving the positions of all atoms to the trajectory every five time steps
        dyn.attach(traj.write, interval=5)

# applying momentum in z-direction to the carbon atom
        system[nat].momentum[2]=(2*e*system[nat].mass)**0.5
        stop = False

# conditions to test whether the atom is displaced or not
        for iterations in range(12):
            dyn.run(50)
            if system[nat].position[2] > 5.0:
                print nat, "Ejected at:", e
                flog.write(',')
                flog.write(str(e))
                stop = True
```

```
                break
            if system.get_velocities()[nat,2] < 0:
                print nat, "Stalled at:", e
                break
        if stop:
            break

# add this line so that the old trajectory won't be overwritten with new data
        system = atoms2.copy()
        system.set_calculator(calc)
        dyn = VelocityVerlet(system, ts * units.fs)
    flog.write('\n')
```

F Numerical integration of the displacement cross section

```
#DEFINE CONSTANTS, GET THE DENSITY OF STATES


M=12.011*1.660539040e-27
x=var('x')
v=var('v')
T=var('T')
E_d=21.14
eps_0 = 8.85418e-12
e = 1.60217662e-19
c = 299792458.0             # always use float values
m_0=9.10938356e-31
m_1=510998.0
Z = 6.0
hbar= 1.054571800e-34
T = 293.15
w=var('w')
w_z=1.0
k=1.38064852e-23


import numpy as np
import sys


# read the zdos data and normalize
data = np.loadtxt('path/zdos.csv',delimiter=',')
u = data[:,0]
y = data[:,1]
f = u * e / hbar       #convert of the frequency axis from eV to 1/s


df=f[1]
int = sum(y)*df        #define integral
```

```
# normalize the data so that the integral over the zdos yields 2
ynorm = y * 2 / int

datanorm = data       #save the normalized data
datanorm[:,0] = f
datanorm[:,1] = ynorm
np.save('path/zdos_normalized.npy', datanorm)
x = datanorm[:,0]
y = datanorm[:,1]

from sage.calculus.interpolation import spline
datalist = zip(x,y)

# equation (9) from Ref. [30], the part to be integrated
u = var('u')
f = (1/2+1/(exp(hbar*u/(k*T))-1))*u
#define the integrand
y2 = y.copy()
for i in srange(len(x)):
    val = x[i]
    y2[i] = f(u=(val+0.005))*y[i]

datalist2 = zip(x,y2)
#interpolate the integrand
integrand=spline(datalist2)

# CALCULATING THE DISPLACEMENT CROSS SECTION
cs = [0.0,0.0,0.0,0.0]
voltages = [85000.0,90000.0,95000.0,100000.0]

# experimental data from STEM
data = [0.0012,0.0109,0.0566,0.328]
```

```
# calculate velocity from the density of states, Eq. (9) from Ref.[30]
v_z = (hbar/(2*M))*integrand.definite_integral(0,3e14)

# velocity distribution Equation (1.20)
P=1/sqrt(2*pi*v_z)*exp(-v^2/(2*v_z))
# defined after Eq. (21) from Ref.[30]
v_max= 8*sqrt(v_z)
err = 0.0
i = 0
E = var('E')
v = var('v')
E_e = var('E_e')

#define max. transferred energy Equation (1.13)
E_n=M*v^2/2
r=1/c*sqrt(E_e*(E_e+2*m_0*c**2))+M*v
t=1/c*sqrt((E_e+E_n)*(E_e+2*m_0*c^2+E_n))
E_max= (r+t)^2/(2*M)/e
# set initial value for the total displacement cross section, Eq. (3.2)
sigma_tot = 0.0

# width for 300K ~1.29 eV
wd = 1.29

# loop over different acceleration voltages
for U in srange(85000.0,105000.0,5000.0):
    sigma= 0.0
    sigma_tot = 0.0
    beta= 0.0
    e_val=21.3125
    beta = sqrt(1-1/((U/m_1)+1)^2)
```

```
    E_e=e*U

# center of the distribution
    xc  = e_val

# width of the distribution
    w   = wd
    dx = 0.1
    x = np.arange(19.0,24.0,dx)
# gaussian distribution of displacement thresholds Eq. (3.1)
    g = sqrt(2/pi)/w*exp(-2*((E-xc)/w)^2)

# loop to integrate the total cross section, Eq. (3.2)
    for e_val in x:
              sigma  = 4*pi*(Z*e^2/(4*pi*eps_0*2*m_0*c^2))^2*((1-beta^2)/beta^4)*
              (E_max/e_val-1-beta^2*ln(E_max/e_val)+pi*Z*e^2/
              (hbar*c)*beta*(2*sqrt(E_max/e_val)-ln(E_max/e_val)-2))

              sigma_tot += g(E=e_val)*numerical_integral(1e28*P(v=v)*
              sigma(v=v,E_e=e*U)*heaviside(E_max(v=v,E_e=e*U)-e_val),
              -v_max,v_max,rule=1,max_points=1000)[0]
    sigma_tot *= dx
    cs[i] = sigma_tot
    print(U,N(cs[i]))
    i = i+1

# calculates the deviation from the experiment (sigma_tot -sigma_experiment)/sigma_e
for i in srange(4):
  err += N(abs(cs[i]-data[i])/data[i])
  print(err)
print(err)
```

Bibliography

[1] A. Chuvilin, J. C. Meyer, G. Algara-Siller, and U. Kaiser, "From graphene constrictions to single carbon chains," *New Journal of Physics*, vol. 11, no. 8, p. 083019, 2009.

[2] C. Jin, H. Lan, L. Peng, K. Suenaga, and S. Iijima, "Deriving carbon atomic chains from graphene," *Phys. Rev. Lett.*, vol. 102, p. 205501, May 2009.

[3] A. Chuvilin, U. Kaiser, E. Bichoutskaia, N. A. Besley, and A. N. Khlobystov, "Direct transformation of graphene to fullerene," vol. 2, no. 6, pp. 450–453.

[4] J. Kotakoski, A. V. Krasheninnikov, U. Kaiser, and J. C. Meyer, "From point defects in graphene to two-dimensional amorphous carbon," *Phys. Rev. Lett.*, vol. 106, p. 105505, Mar 2011.

[5] O. Krivanek, G. Corbin, N. Dellby, B. Elston, R. Keyse, M. Murfitt, C. Own, Z. Szilagyi, and J. Woodruff, "An electron microscope for the aberration-corrected era," *Ultramicroscopy*, vol. 108, no. 3, pp. 179 – 195, 2008. Proceedings of the Sixteenth International Microscopy Congress.

[6] T. Susi, J. C. Meyer, and J. Kotakoski, "Manipulating low-dimensional materials down to the level of single atoms with electron irradiation," *Ultramicroscopy*, vol. 180, no. Supplement C, pp. 163 – 172, 2017. Ondrej Krivanek: A research life in EELS and aberration corrected STEM.

[7] J. Kotakoski, D. Santos-Cottin, and A. V. Krasheninnikov, "Stability of graphene edges under electron beam: equilibrium energetics versus dynamic effects," *ACS nano*, vol. 6, no. 1, pp. 671–676, 2011.

[8] R. Peierls, "Quelques propriétés typiques des corps solides," *Annales de l'institut Henri Poincaré*, vol. 5, no. 3, pp. 177–222, 1935.

[9] L. Landau, "Zur theorie der phasenumwandlungen ii," *Phys. Z. Sowjetunion*, vol. 11, no. 545, pp. 26–35, 1937.

[10] K. S. Novoselov, A. K. Geim, S. V. Morozov, D. Jiang, Y. Zhang, S. V. Dubonos, I. V. Grigorieva, and A. A. Firsov, "Electric Field Effect in Atomically Thin Carbon Films," *Science*, vol. 306, pp. 666–669, Oct. 2004.

[11] A. H. Castro Neto, F. Guinea, N. M. R. Peres, K. S. Novoselov, and A. K. Geim, "The electronic properties of graphene," *Rev. Mod. Phys.*, vol. 81, pp. 109–162, Jan 2009.

[12] P. R. Wallace, "The band theory of graphite," *Physical Review*, vol. 71, no. 9, p. 622, 1947.

[13] C. Lee, X. Wei, J. W. Kysar, and J. Hone, "Measurement of the elastic properties and intrinsic strength of monolayer graphene," *science*, vol. 321, no. 5887, pp. 385–388, 2008.

[14] A. A. Balandin, S. Ghosh, W. Bao, I. Calizo, D. Teweldebrhan, F. Miao, and C. N. Lau, "Superior thermal conductivity of single-layer graphene," *Nano letters*, vol. 8, no. 3, pp. 902–907, 2008.

[15] W. Bao, F. Miao, Z. Chen, H. Zhang, W. Jang, C. Dames, and C. N. Lau, "Controlled ripple texturing of suspended graphene and ultrathin graphite membranes," *Nature nanotechnology*, vol. 4, no. 9, pp. 562–566, 2009.

[16] K. S. Novoselov, A. K. Geim, S. Morozov, D. Jiang, M. Katsnelson, I. Grigorieva, S. Dubonos, Firsov, and AA, "Two-dimensional gas of massless dirac fermions in graphene," *nature*, vol. 438, no. 7065, pp. 197–200, 2005.

[17] N. M. R. Peres, F. Guinea, and A. H. Castro Neto, "Electronic properties of disordered two-dimensional carbon," *Phys. Rev. B*, vol. 73, p. 125411, Mar 2006.

[18] M. Knoll and E. Ruska, "Das elektronenmikroskop," *Zeitschrift für Physik A Hadrons and Nuclei*, vol. 78, no. 5, pp. 318–339, 1932.

[19] M. Haider, S. Uhlemann, E. Schwan, H. Rose, B. Kabius, and K. Urban, "Electron microscopy image enhanced," *Nature*, vol. 392, pp. 768–769, 1998.

[20] P. D. Nellist, M. F. Chisholm, N. Dellby, O. Krivanek, M. Murfitt, Z. Szilagyi, A. R. Lupini, A. Borisevich, W. Sides, and S. J. Pennycook, "Direct sub-angstrom imaging of a crystal lattice," *Science*, vol. 305, no. 5691, pp. 1741–1741, 2004.

[21] P. Hawkes, "Aberration correction past and present," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1903, pp. 3637–3664, 2009.

[22] J. Kotakoski, J. Meyer, S. Kurasch, D. Santos-Cottin, U. Kaiser, and A. Krasheninnikov, "Stone-wales-type transformations in carbon nanostructures driven by electron irradiation," *Physical Review B*, vol. 83, no. 24, p. 245420, 2011.

[23] A. Krasheninnikov and K. Nordlund, "Ion and electron irradiation-induced effects in nanostructured materials," *Journal of applied physics*, vol. 107, no. 7, p. 3, 2010.

[24] J. C. Meyer, F. Eder, S. Kurasch, V. Skakalova, J. Kotakoski, H. J. Park, S. Roth, A. Chuvilin, S. Eyhusen, G. Benner, *et al.*, "Accurate measurement of electron beam induced displacement cross sections for single-layer graphene," *Physical review letters*, vol. 108, no. 19, p. 196102, 2012.

[25] T. Susi, C. Hofer, G. Argentero, G. T. Leuthner, T. J. Pennycook, C. Mangler, J. C. Meyer, and J. Kotakoski, "Isotope analysis in the transmission electron microscope," *Nature communications*, vol. 7, p. 13040, 2016.

[26] F. Banhart, "Irradiation effects in carbon nanostructures," *Reports on progress in physics*, vol. 62, no. 8, p. 1181, 1999.

[27] N. F. Mott, "The scattering of electrons by atoms," *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, vol. 127, no. 806, pp. 658–665, 1930.

[28] A. Zobelli, A. Gloter, C. Ewels, G. Seifert, and C. Colliex, "Electron knock-on cross section of carbon and boron nitride nanotubes," *Physical Review B*, vol. 75, no. 24, p. 245402, 2007.

[29] W. A. McKinley Jr and H. Feshbach, "The coulomb scattering of relativistic electrons by nuclei," *Physical Review*, vol. 74, no. 12, p. 1759, 1948.

[30] F. Seitz and J. Koehler, "Displacement of atoms during irradiation," *Solid State Physics-Advances In Research And Applications*, vol. 2, pp. 305–448, 1956.

[31] D. Marx and J. Hutter, *Ab initio molecular dynamics: basic theory and advanced methods*. Cambridge University Press, 2009.

[32] C. P. Robert, *Monte carlo methods*. Wiley Online Library, 2004.

[33] H. Haario, E. Saksman, J. Tamminen, *et al.*, "An adaptive metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, 2001.

[34] J. Grotendorst, D. Marx, and A. Muramatsu, "Quantum simulations of complex many-body systems: from theory to algorithms," *NIC series*, vol. 10, pp. 211–254, 2002.

[35] G. Sutmann, *Classical molecular dynamics and parallel computing*. FZJ-ZAM, 2002.

[36] I. Newton, *Sir Isaac Newton's mathematical principles of natural philosophy and his system of the world*, vol. 1. Univ of California Press, 1962.

[37] M. P. Allen *et al.*, "Introduction to molecular dynamics simulation," *Computational soft matter: from synthetic polymers to proteins*, vol. 23, pp. 1–28, 2004.

[38] P. Dollfus *et al.*, *Simulation of Transport in Nanodevices*. John Wiley & Sons, 2016.

[39] L. Verlet, "Computer" experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules," *Physical review*, vol. 159, no. 1, p. 98, 1967.

[40] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, "A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters," *The Journal of Chemical Physics*, vol. 76, no. 1, pp. 637–649, 1982.

[41] D. W. Brenner, O. A. Shenderova, J. A. Harrison, S. J. Stuart, B. Ni, and S. B. Sinnott, "A second-generation reactive empirical bond order (rebo) potential energy expression for hydrocarbons," *Journal of Physics: Condensed Matter*, vol. 14, no. 4, p. 783, 2002.

[42] P. Erhart and K. Albe, "Analytical potential for atomistic simulations of silicon, carbon, and silicon carbide," *Physical Review B*, vol. 71, no. 3, p. 035211, 2005.

[43] J. Tersoff, "Modeling solid-state chemistry: Interatomic potentials for multicomponent systems," *Physical Review B*, vol. 39, no. 8, p. 5566, 1989.

[44] D. W. Brenner, "Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films," *Physical Review B*, vol. 42, no. 15, p. 9458, 1990.

[45] W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects," *Physical review*, vol. 140, no. 4A, p. A1133, 1965.

[46] P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," *Physical review*, vol. 136, no. 3B, p. B864, 1964.

[47] G. Abell, "Empirical chemical pseudopotential theory of molecular and metallic bonding," *Physical Review B*, vol. 31, no. 10, p. 6184, 1985.

[48] Y. Hu, S. Shen, L. Liu, C. S. Jayanthi, S.-Y. Wu, and S. B. Sinnott, "Thin-film nucleation through molecular cluster beam deposition: Comparison of tight-binding and many-body empirical potential molecular dynamics simulations," *The Journal of chemical physics*, vol. 116, no. 15, pp. 6738–6744, 2002.

[49] M. Elstner and G. Seifert, "Density functional tight binding," *Phil. Trans. R. Soc. A*, vol. 372, no. 2011, p. 20120483, 2014.

[50] D. Porezag, T. Frauenheim, T. Köhler, G. Seifert, and R. Kaschner, "Construction of tight-binding-like potentials on the basis of density-functional theory: Application to carbon," *Physical Review B*, vol. 51, no. 19, p. 12947, 1995.

[51] G. Seifert, D. Porezag, and T. Frauenheim, "Calculations of molecules, clusters, and solids with a simplified lcao-dft-lda scheme," *International journal of quantum chemistry*, vol. 58, no. 2, pp. 185–192, 1996.

[52] M. Elstner, D. Porezag, G. Jungnickel, J. Elsner, M. Haugk, T. Frauenheim, S. Suhai, and G. Seifert, "Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties," *Physical Review B*, vol. 58, no. 11, p. 7260, 1998.

[53] M. Gaus, Q. Cui, and M. Elstner, "Dftb3: extension of the self-consistent-charge density-functional tight-binding method (scc-dftb)," *J. Chem. Theory Comput*, vol. 7, no. 4, pp. 931–948, 2011.

[54] B. Aradi, B. Hourahine, and T. Frauenheim, "Dftb+, a sparse matrix-based implementation of the dftb method," *The Journal of Physical Chemistry A*, vol. 111, no. 26, pp. 5678–5684, 2007.

[55] J. Hubbard, "Electron correlations in narrow energy bands," in *Proc. R. Soc. Lond. A*, vol. 276, pp. 238–257, The Royal Society, 1963.

[56] R. S. Mulliken, "Electronic population analysis on lcao–mo molecular wave functions. i," *The Journal of Chemical Physics*, vol. 23, no. 10, pp. 1833–1840, 1955.

[57] A. Larsen, J. Mortensen, J. Blomqvist, I. Castelli, R. Christensen, M. Dulak, J. Friis, M. Groves, B. Hammer, C. Hargus, *et al.*, "The atomic simulation environment—a python library for working with atoms," *Journal of Physics: Condensed Matter*, 2017.

[58] S. v. d. Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: a structure for efficient numerical computation," *Computing in Science & Engineering*, vol. 13, no. 2, pp. 22–30, 2011.

[59] G. Kresse and O. Lebacq, "Vasp manual," 2013.

[60] E. Bitzek, P. Koskinen, F. Gähler, M. Moseler, and P. Gumbsch, "Structural relaxation made simple," *Physical review letters*, vol. 97, no. 17, p. 170201, 2006.

[61] T. Developers, "Sagemath," 2016.