Efe Astarlioglu

Simulation of carbon nanostructures with a vacancy present

Metropolia

Author Title Number of Pages Date	Efe Astarlioglu Simulation of carbon nanostructures with a vacancy present 35 pages + 0 appendices 28 May 2018
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Professional Major	Smart Systems Engineering
Instructors	Sami Sainio, Thesis Supervisor

Carbon is a very special element given it can take part in millions of different chemical compounds due to its 4 valence electrons and its compact size to fit in larger molecules. Further, carbon forms allotropes. Most familiar allotropes of carbon are diamond and graphite. Graphene and carbon nanotube will be the two allotropes of carbon which will be the subject in this project.

Both graphene and carbon nanotubes have unique mechanical, thermal, optic and electrical properties. Therefore, they are either in use, or can potentially be used, in myriad different applications in industries or research. Consequently, it is important for future development to understand the behavior of these allotropes in diverse settings.

Main task of this project was to investigate the behavior and effect of vacancy defects (point defects) in graphene sheet and carbon nanotube of different sizes as a function of temperature. In order to study these behavior and effects, computer simulations on molecular dynamics were employed. This report will introduce the necessary chemistry and material science concepts as well as tools used in simulation of the structures.

As a result of the simulations, diffusion of vacancies in the structures was observed. The simulations ran with a resolution of 1 femtoseconds in which the structures were heated from 0K to upwards of 4500K.The diffusions took place at high temperatures for both graphene and carbon nanotube.

Keywords	carbon nanotube, graphene, simulation, molecular dynamics

Metropolia

Contents

List of Abbreviations

ASE   Atomic Simulation Environment

LAMMPS   Large-Scale Atomic/Molecular Massively Parallel Simulator

AMU   Atomic mass unit.

CNT   Carbon Nanotube

GUI   Graphical User Interface

1 Introduction

Carbon is one of the most abundant elements in the known universe and its ability to take part in diverse and complex chemical compounds makes it the ideal candidate to be the building block of the known life forms. In addition to being the building block of life, some recently discovered allotropes of carbon, such as graphene and carbon nanotubes, show unique features that could be beneficial to different industries or research. [1]. Possible applications range from using graphene composites for lightweight and durable aircraft wings to as an agent in biomedical for drug delivery [2].

With the development on both hardware and software aspect, computers can do more operations per unit time. The operation power made computers suitable to mimic the natural phenomena for researchers to investigate. The simulations run by the computers are now widely used around globe for different purposes such as education, research or design. Main reason why this topic was chosen is because both simulations and computers improve progressively and their importance for the society grow.

The question this thesis is trying to answer is how the structure of a graphene sheet or a carbon nanotube, which includes a vacancy, behave as it is heated. The topic for this thesis was presented by Sami Sainio who is working on his postdoctoral research in Professor Tomi Laurila's research team. The research team is investigating carbon nanostructures to utilize them in development of biosensors.

After the introduction, report will continue with the literature review which aims to provide the reader with the basic concepts of necessary chemistry, material sciences and physics concepts and will give a brief introduction on the graphene, carbon nanotubes and vacancies. Tools and implementation will cover how the Atomic Simulation Environment (ASE) and Large-Scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) were utilized to create atomic structures and running simulations. In results section, the analyzed results of the simulations will be presented and discussed. The report will end with a brief summary and the conclusions.

2 Literature Review

2.1 Atomic Structure

The concept that the matter is comprised of small and indivisible fundamental pieces have a long history which goes back to 400 B.C. The name "atom" was derived for this concept by Greek philosophers Leucippus and Democritus where '*a*' means not and '*to-mos*' mean divided. Although we now know atoms themselves can be divided, it is out of the scope of this report. [3,1505]

Atoms are described as the smallest pieces of an element that could take part in a chemical combination. Atoms are formed by nucleus, which is formed by protons and neutrons, and an electron cloud that surrounds it. Figure 1 depicts an atom.
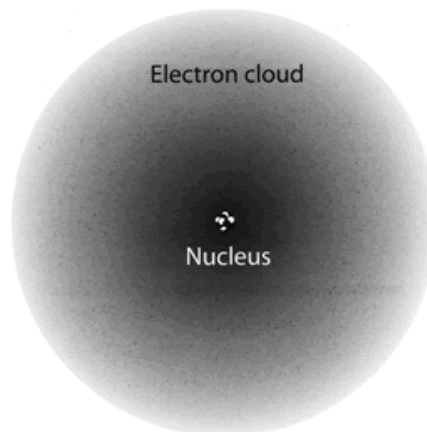

Figure 1.   The atom nucleus and the electron cloud. Reprinted from Energy Education [4].

Nucleus of an atom, although small in terms of volume, amounts to most of the mass of an atom. Since a single atom's mass is very small for humans to visualize, it is generally represented with atomic mass unit, also known as amu. Amu is defined as $\frac{1}{12}$ of a carbon-12 atom. Protons have a mass of 1.0073 amu while neutrons are slightly heavier with a mass of 1.0087 amu. Electrons are much lighter with a mass of 0.00055 amu. [5,36]

Another important aspect of an atom is the electric charge. Protons have an electric charge of +1, electrons have an electric charge of -1 and neutrons do not carry electric

charge. Because two opposite charges attract each other, electrons orbit the nucleus of the atom. An atom has the same amount of protons and electrons, therefore, is neutral as a whole system.

Atoms could become charged particles (ions) if they lose or gain electrons or protons. In conventional chemical reactions, nucleus of an atom is not involved. Instead, it is electrons that move. When an atom receives electrons, it becomes a negatively charged and that state is named anion. In the opposite case, when an atom releases its electrons it becomes positively charged and named cation.

2.2 Valence electrons and Carbon

Valence electrons refers to the electrons in an atom which are most likely to react with other atoms. This reaction can consist of sharing, releasing and receiving these electrons. Valence electrons of an atom are, generally, the ones that are farthest from the nucleus and have the highest energy. Valence electrons in an atom constantly try to find octet. Atoms will, therefore, try to take part in chemical compounds to have most the stable electron configuration they can achieve. [5,169]

Additionally, to the electronic structure of an atom, valence also depends on the immediate surroundings. Carbon, for example, have a total of four valence electrons. [1 pp. 31]

2.3 Chemical Bonding

Chemical bonds can be summarized as continuous attraction between two atoms, molecules or ions. This attraction (force) holds the chemical compounds together. Force between ions or atoms can be divided into two by how electrons behave in the system. If electrons are transferred between atoms and ions are developed, the resulting bonds are called ionic bonds. Should the electrons be shared between atoms, the resulting bonds are called covalent bonds.

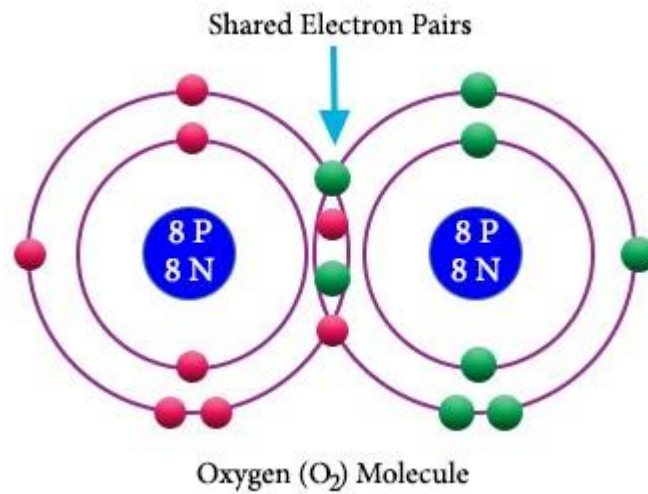Figure 2 demonstrates a covalent bond between two oxygen atoms.

Figure 2. Covalent bonding between oxygen atoms. Reprinted from TutorVista. 2018. [6]

As shown in the Figure 2, there is a double covalent bonding between two oxygen molecules which includes 4 electrons. This is the case because oxygen has six valance electrons. The reason behind oxygen having six valance electrons is the octet rule. Octet rule is the dictates that the atoms behave in a way so that they have eight electrons in their valence shells. The concept of valance will be discussed later in the report. Figure 3 illustrates an ionic bond.
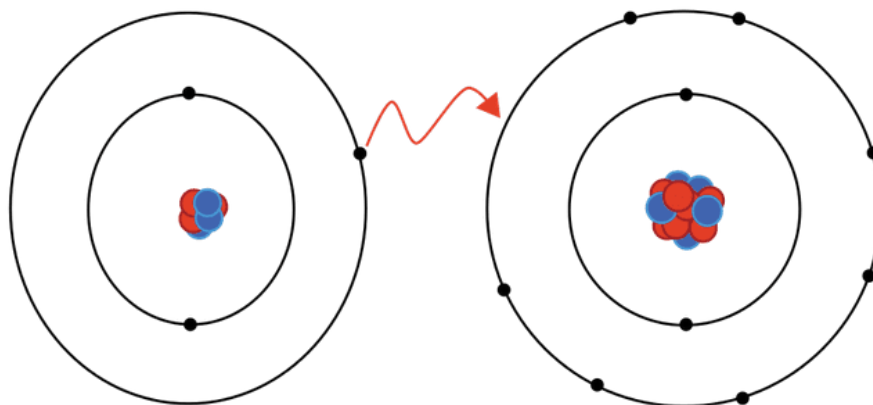

Figure 3. An ionic bond. Reprinted from ThoughtCo. 2018.[7]

It can be observed from the Figure 3 that an electron given by the atom on the left would cause both atoms to be ions and therefore they would be attracted to each other because of the electrostatic force.

Two different types of chemical bonds, metallic and covalent bonds, will be discussed in this report. These two types of bonds were chosen because both graphene and carbon nanotubes include metallic and covalent bonds.

2.3.1   Metallic Bonds

Metallic bonds form when metallic elements release their valence electrons. These released electrons constitute an electron ocean around the atoms. Because of the electron release, each atom becomes a cation. Since electrons are already negatively charged, the cations are jointly attracted to the electron ocean. This joint attraction generates a strong bond. [1,35]

Metals are good conductors of electricity even in low temperatures because the valence electrons can move freely. Same property also causes metals to be good conductors of heat.[1,35]

2.3.2   Covalent Bonds

Covalent bonding refers to the bonds which form between two or more atoms when they are sharing electrons. Electrons can be shared between atoms when they have the similar inclination to release or receive electrons. It is common for nonmetal elements to form covalent bonds. Attraction of the shared electrons to the nucleus of the atoms which form the molecule is very strong. Therefore, the bond that is created is also very strong. [5,188]

As it will be discussed in the chapter 2.4 and 2.5, graphene and carbon nanotubes are formed with covalent bonds that are of type of pi (π bonds) and sigma (σ bonds). Valence bond theory associate covalent bonds with the overlapping atomic orbitals. According to valence bond theory, there are two conditions that need to occur for a covalent bond to form. Firstly, orbital of an atom needs to overlap with another atom. Secondly, an electron combination should be occupying both orbitals. [5,230]

Sigma bonds are formed when two atomic orbitals overlap from their ends. Two *s* orbitals or two *p* orbitals could form a sigma bond. A sigma bond can also be formed between an *s* and a *p* orbital. [5,231] Figure 4 exhibits sigma bonding.
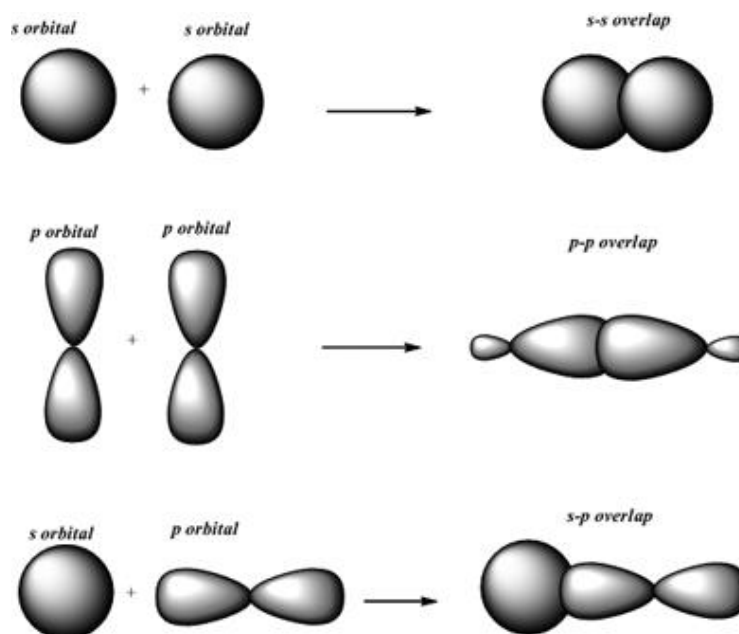

Figure 4.   Sigma bonding between different orbitals. Reprinted from Chegg Inc. 2018. [8]

Figure 4 also demonstrates if a hypothetical line between the nuclei was to be drawn, it would pass through the center of the region of overlapping, which is shown with a + sign in Figure 4.

Similarly to sigma bonds, pi bonds are formed when two atomic orbitals are overlapping. However, pi bonds are formed by overlapping of orbitals side to side, rather than end to end.

A pi bond can form only between two p orbitals. [5, 231] Figure 5 illustrates pi bonds.
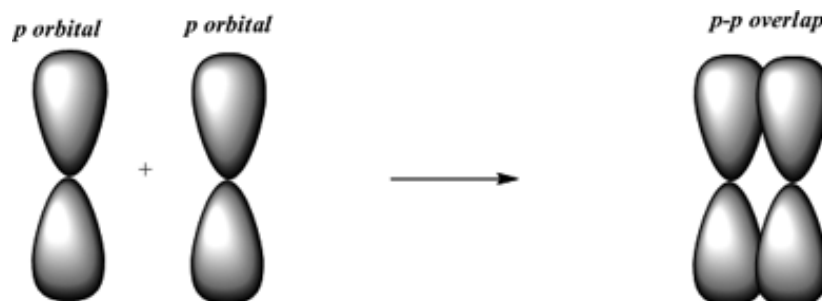

Figure 5.   Pi bonding. Reprinted from Chegg Inc. 2018. [8]

As it can be seen from the Figure 5, the region that is overlapped are situated above and below of the inter-nuclear axis. Additionally, the overlap in pi bonds are, generally, weaker compared to the sigma bonds. This is also the main reason why double bonding between atoms are commonly not twice as strong as a single bond between the same atoms. [5, 232]

The last bonding that should be mentioned is the van der Waals bonding. The root causes of the van der Waals forces, which causes the van der Waals bonding, is of quantum mechanical in its nature. A full analysis of them would therefore require discussion in quantum mechanics which is beyond the scope of this project. [1, 38]

Nevertheless, an abstract picture can be given. When two particles are charged with opposite charges and have a distance between them, a dipole moment is formed. Although atoms are neutral, if an external or internal electric field is applied on them, they will become polarized and a dipole moment occurs. [1,38]

Similarly, molecules can be polarized artificially. However, there are some molecules that are permanently polarized due to their internal parameters. Water, for example, has a permanent and intrinsic dipole moment. The molecules with a dipole moment attract each other which creates the van der Waals forces. [1,38]

2.4   Graphene

Graphene is an allotrope of carbon which is formed by a single layer of carbon atoms in a hexagonal arrangement and is two dimensional. Graphene is also the base structure for carbon nanotubes. Graphene is considered a crystal, because it is a solid and its microscopic structure is extremely ordered. Figure 6 illustrates a graphene sheet.
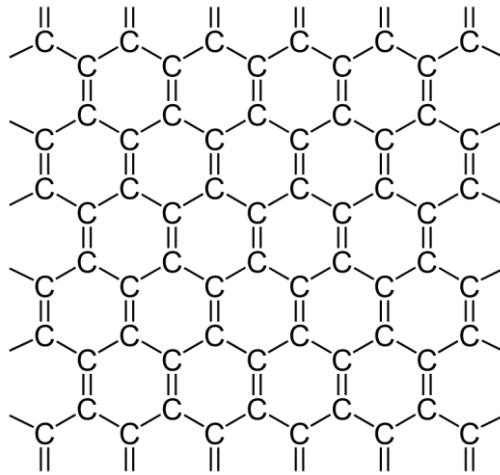

Figure 6.   Microscopic structure of graphene. Reprinted from CommerceGurus. 2015. [9]

It can be observed from the Figure 6 that each carbon atom is bonded to 3 other carbon atoms with 4 chemical bonds. Each carbon atom has a sigma bond with all of its neighbors and a pi bond with one of its neighbors. This is also illustrated in Figure 1. The distance between the atoms are 1.42 Å. [10]

Owing to its crystal structure, graphene has unusual band structure. Typically, the difference between conductors and insulators lies in the difference in their energy bands. In a material which is nonconductive, the energy bands are full or empty and they are separated by an energy gap. In a metal, one of the bands are only partly filled. Graphene has no gap but also no partially filled bands. This property can make graphene very important for the semi-conductor industry. [11]

Graphene is a great conductor of both heat and electricity and has very low resistivity at room temperature. In terms of mechanical properties, graphene has the highest tensile strength of any known material. [10]

2.5   Carbon nanotubes

Carbon nanotube (CNT) is another allotrope of carbon which has a nanostructure of a cylinder. They are generally assumed as graphene sheet that is rolled into a hollow cylinder shape. [12]

CNTs can form in different types and the chiral vector defines these formations. This vector, which is also depicted in Figure 7, is described with two parameters *n* and *m*. The angle this chiral vector has between the vector a_1, θ, also plays a role in the formation of the carbon nanotube.
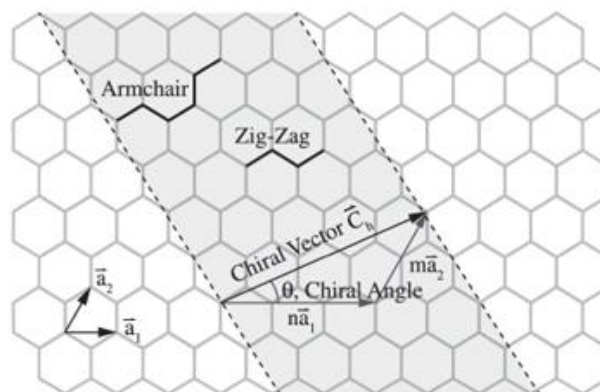

Figure 7.   Chiral vector. Reprinted from 11 Rafiee R, Pourazizi R. 2014. [12]

The angle θ in Figure 7 can range between 0 and 30 degrees. This determines which of the three carbon nanotube would form. If the angle is 0 degrees, the CNT is named zig zag and arm chair if the angle is 30 degrees. Should the angle is a value between 0 and 30 degrees, it forms a chiral CNT. Figure 8 shows the differences in the arrangements in different CNTs. [12]
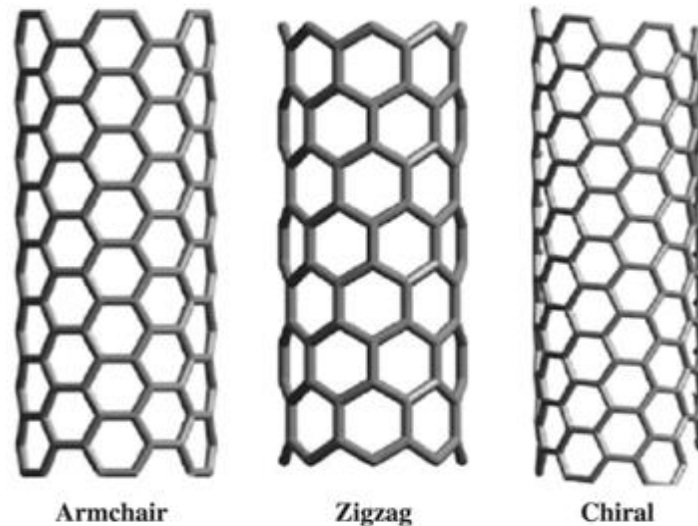
Figure 8.   Different carbon nanotubes. Reprinted from 11 Rafiee R, Pourazizi R. 2014. [12]

Depending on the atomic arrangement of carbon nanotubes, they can either show metallic or semi-conducting features. If carbon nanotube is of type armchair, it shows metallic features. Approximately, one third of zigzag or chiral type CNTs show metallic properties and rest show semi conducting properties. [13 pp. 76 - 77]

If a single walled nanotube is metallic, it is a good electrical conductor. All types of carbon nanotubes have high thermal conductivity and display superconductivity at low temperatures (below 20 Kelvin). [14]

2.6   Point Defects and Vacancies

Most important of the point defects for this project are vacancies, also known as crystallographic defects. In the presence of vacancies, an otherwise perfect crystal structure would be missing an atom at a lattice point. Even though vacancies disrupt only 1 lattice point, neighboring atoms could also be affected.

The relaxation of the structure around the vacancy can be seen in the Figure 9.
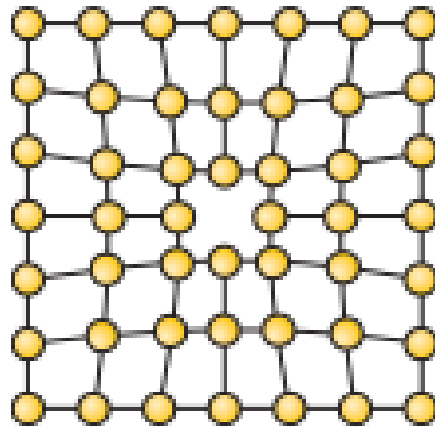
Figure 9.   Point defect in a crystal system Reprinted from The Science and Engineering of Materials. 2011. [1]

These imperfections in the crystal structure can be specifically introduced to a crystal structure or it could be a byproduct of atoms acquiring energy. High energy atoms could leave their lattice point and leave a vacancy behind. [1,115]

Vacancies in a crystalline structure also increases the entropy of the system, in turn, increasing the thermodynamic stability of the structure. Typically, around room temperatures, amount of vacancies in a crystalline per unit volume is low. As the temperature of the structure increases, "density" of the vacancies also increase in an exponential fashion. The equation below shows the behavior.

$$n_v = n \exp(\frac{-Q_v}{RT})$$

(1)

In this equation, n_v is number of vacancies per cm^3. n is the total number of atoms per unit volume. Q_v is energy necessary to create 1 mole of vacancies. R is the gas constant and T is the temperature. [1,115]

Additionally to the vacancies, there are four other point defects that report will briefly cover. All of these point defects could be naturally occurring or could be introduced to

the material for different purposes. If these defects are intentionally introduced to a material, they are named dopants. An example for this would be pure silicon being doped by phosphorus and boron to boost its properties. [1,115] Following figure demonstrates 3 other point defects.
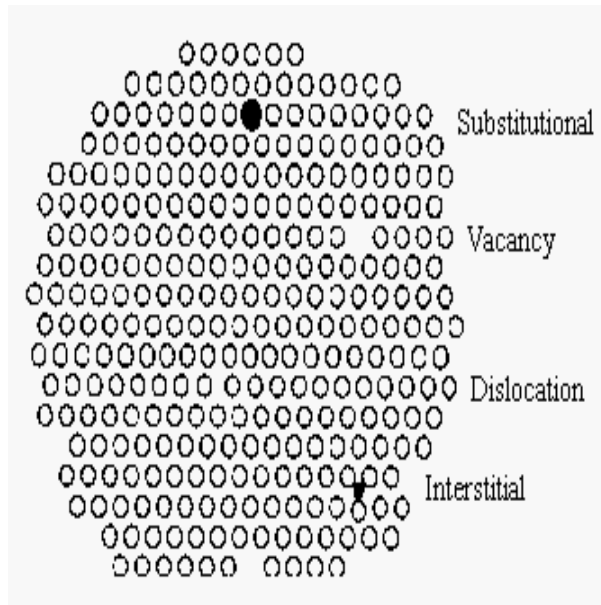

Figure 10. Point defects. Reprinted from Purdue University. 2004. [15]

An interstitial defect occurs when there is an extra atom present in a structure where there is no lattice point. Although the interstitial atoms are smaller than the atoms that constitutes the structure, they deform the surrounding structure. Substitutional defects take place when a different atom or ion, compared to the atom or ion that forms the crystal structure, is occupying a lattice point. The substitutional atoms can be larger or smaller than the regular atoms comprising the structure but in both cases, they disrupt the lattice spacing.

An additional point defect is the Frenkel defect. It is a combination of an interstitial defect and a vacancy. When an ion moves into an interstitial place from a lattice point, it produces a Frenkel defect. Even though the Frenkel defect is mostly observed in the ionic materials, it can be seen in metals or structures that have covalent bonds. The last point defect is only observed in ionic compounds. In Schottky defects, vacancies cause cations and anions to leave their lattice point for the whole structure to conserve its electrical neutrality. The figure below displays Schottky and Frenkel defects. [1,120]
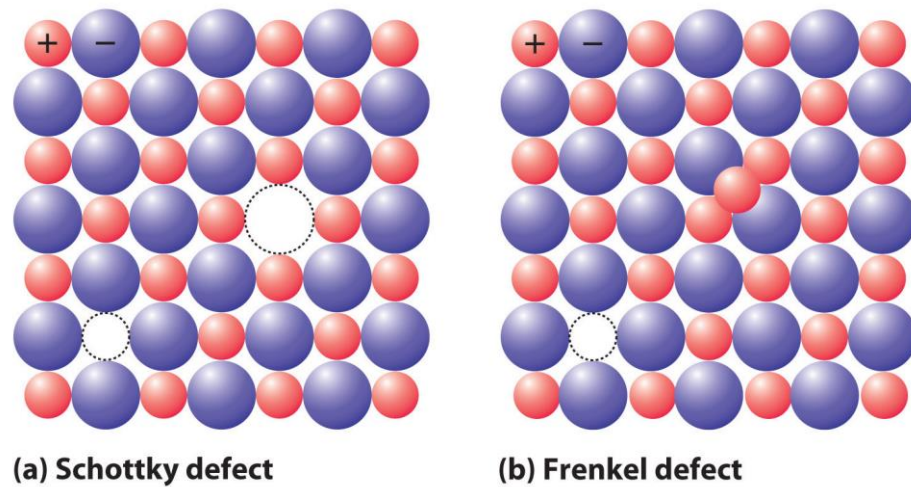
Figure 11. Schottky and Frenkel defect. Reprinted from MajorDifferences. 2018.[16]

2.7   Atomic Diffusion

Atomic diffusion is a process by which atoms or ions are transported in a solid material. This transportation of atoms can also be referred to as a flux. The magnitude of it is affected by two parameters. Temperature and the concentration gradient. As temperature increases, the capacity of atoms to diffuse increase. This takes effect because each of the atoms or ions possess more energy. The rate at which atoms diffuse as the temperature changes is given by the formula below. [1,159]

$$Rate = \ c_0 \exp(\frac{-Q}{RT})$$

(2)

In the formula, the c_0 and R are constants, T is temperature and Q is energy of activation. The activation energy is the amount of energy that is needed to move an Avogadro's number of atoms to move. Therefore, its unit is cal/mol.

When an atom abandons its lattice point and moves to a vacancy, its original spot is left as a vacancy. This makes vacancy move. This can also be interpreted as the vacancy

diffusion. Motion of the atom and motion of the vacancy will naturally be in opposite directions. It can be visualized in the following Figure 12 a. [1,162]

Motion of atom ⟶

⟵ Motion of vacancy

(a) Vacancy mechanism   (b) Interstitial mechanism

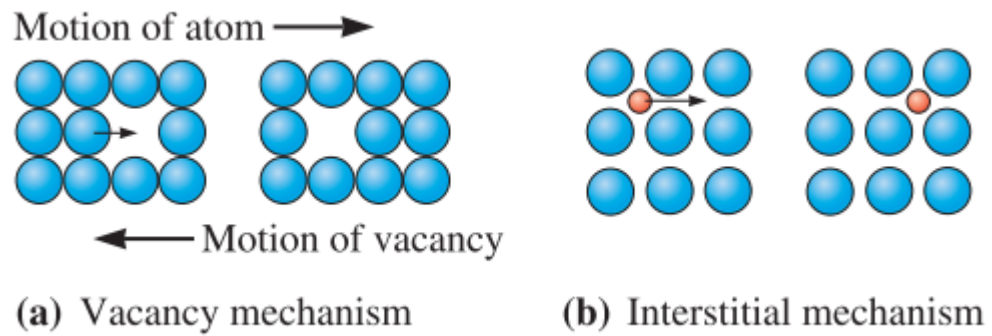Figure 12. Diffusion of vacancies and interstitial diffusion. Reprinted from Cengage Learning. 2011. [1, 162]

Additionally to the atoms in the lattice points, atoms that make up the interstitial defects can move in the structure. Vacancies are not a must for the interstitial diffusion. This kind of diffusion happens easier compared to vacancy diffusion and the diffusion of the atom is faster because of the size of the atom. [1,162].

3   Tools and Implementation

This section of the report will introduce the concept of molecular dynamics as well as the tools that were used and how they were utilized to run a simulation. In the case of material sciences, the aim of computer simulations is to gain understanding on the characteristics of how molecules behave in a given structure. These simulations can be a supplement to traditional experiments, or they can be an experiment themselves in order to test a prediction or theory. Computer simulations form a link between the macroscopic environment of traditional experiments and microscopic environment of the molecules. They reveal information on what makes structures behave the way they do internally.
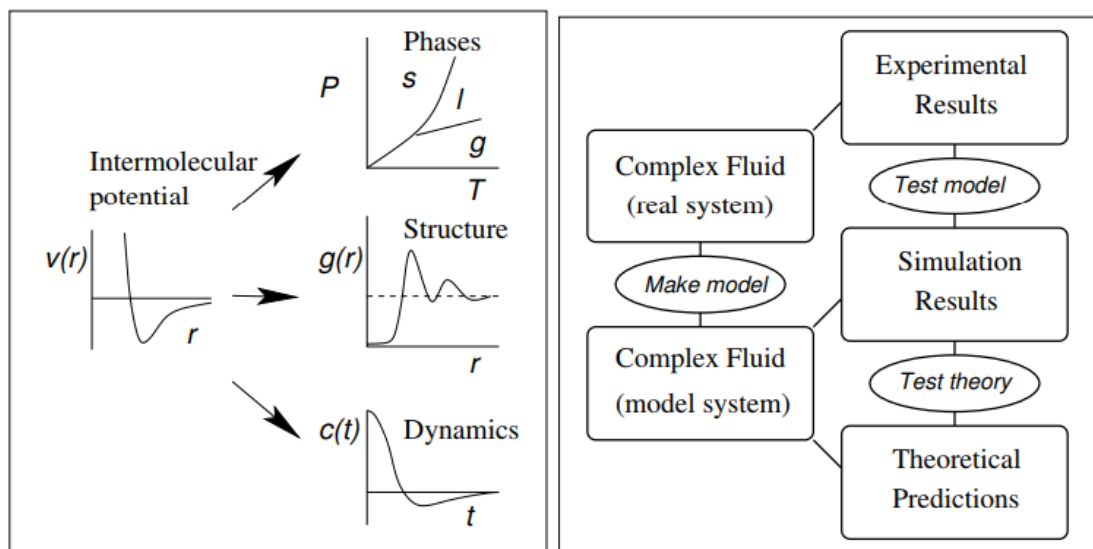

Figure 13. Linking between the theory and experiment. Reprinted from John von Neumann Institute for Computing. 2004.[17,2]

Figure 13 illustrates how computer simulations can be used and how they are the link between the theory and experiments. As shown in the figure above, researchers can create a model of a complex fluid system and test this model in order to prove a theoretical predictions or understand the intrinsic nature of the experimental results. Additionally, simulations can be utilized when a conventional experiment in the investigated issue is very difficult or impossible. For instance, a very low or high temperature or pressure environment could be purely simulated.

Depending on the purpose of a computer simulation, the accuracy it will be determined and the model and simulation will be constructed accordingly. In general, when the accuracy is compromised, less computing power and time will be needed and when the accuracy of a simulation is of essence, total computing power and time needed will be greater.

3.1 Molecular Dynamics

In this project, a special case of computer simulations, molecular dynamics, was used. Molecular dynamics aim to examine the physical movement of molecules and atoms in a given system. This investigation is carried out by solving the equations of motion (Newton's laws) for each atom per defined fixed amount of time. This fixed amount of time is also referred as time step. Molecular dynamics are being used in different branches of physics and chemistry, as well as material science. Figure 14 depicts the simplified flow of molecular dynamics.
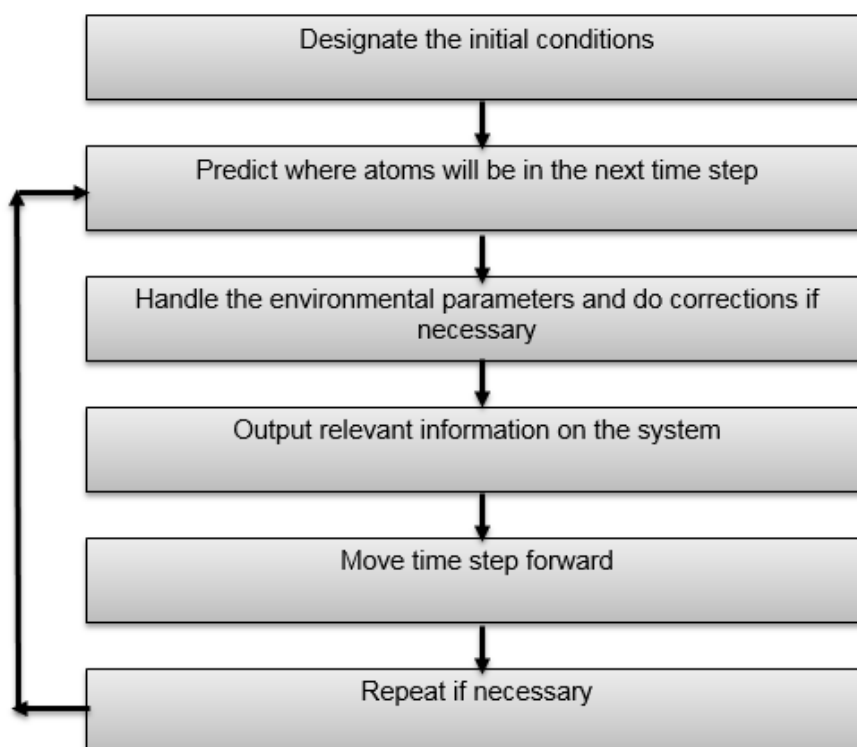

Figure 14. Simplified flowchart of a molecular dynamics algorithm

The macro state of a structure is defined by its micro states. Designation of the initial conditions consist assigning each atom or molecule with a coordinate, velocity and rotation, if necessary. To predict the new location and parameter of particles, an approach is needed to determine the net force acting on the object. From there on, acceleration of the particle can be found using classical equations of motion. Figure 15 describes the interatomic potential energy as a function of interatomic distance.
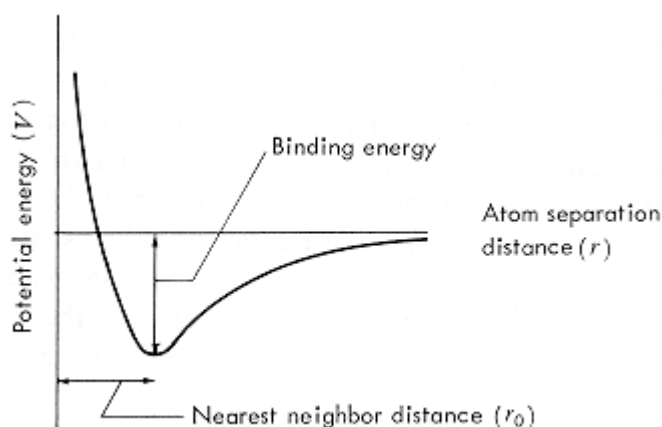

Figure 15. Interatomic potential energy with respect to interatomic distance. Reprinted from Prentice-Hall. 1973. [18]

The forces acting on particles are calculated by differentiating the total potential energy with respect to position of atoms. It should be noted that interactions between atoms are of quantum mechanical in nature and are defined by the Schrödinger's Equation. Because even a small structure contains a vast number of atoms, solving the Schrödinger's equation for each atom would require immense amount of computing budget. Therefore, all the interatomic potentials are approximations.

As investigation of a phenomenon might demand different environmental parameters, they must be accounted after the initial prediction. For instance, if the investigation is on a system where temperature should not change, thermostat algorithms are used to keep the temperature restrained.

3.2   Atomic Simulation Environment

For creating the molecular structure, the tool chosen is Atomic Simulation Environment (ASE). ASE is a library developed in Python to build, manipulate, simulate and analyze

the atomic structures. At the moment, most recent stable release of ASE is v.3.16.0 and it was used in this project. As of now, ASE can be used freely. [19]

Python, as a programming language, is popular in scientific research domain as it has an understandable syntax and is free and open source and have good libraries such as NumPy, SciPy and matplotlib. As ASE is developed in Python, it can make use of the language's powerful features when scripting, as well as its popularity amongst scientific communities. [19]

ASE is designed with an object-oriented approach and therefore becomes the front end where there is a high level of abstraction. This object orientation also boosts ASE's flexibility and customizability.

There are different ways of creating a molecular structure in ASE. However, all of these ways include, explicitly or implicitly, using the Atoms library. Atoms library is responsible of many critical functions of ASE. Atoms module not only creates the atoms which would then form molecules, it handles all the parameters that is related with the atoms as well as the unit cell.

The atoms and the unit cell information are held in arrays. Therefore, manipulation of single or multiple atoms is fairly easy using NumPy. Atoms object have relevant functions to get and set the values for these parameters in a safe manner. These functions can manipulate parameters of a single atom or all the present atoms. For instance, if position of an atom needs to be changed, the function *set_positions()* can be called where argument for this function is the index of the atom. Similarly, function *get_potential_energy()* can be called which would display the total potential energy of all atoms in the structure.

Using the atoms library, a unit cell with the desired properties could be created. This unit cell can then be copied and translated in space to form crystal structures. Another way to build structures would be to utilize the build option of ASE. This function calls the atoms object implicitly and creates structures.

ASE also has a visualization library with which not only structures can be seen but also provides a graphical user interface. The viewing window provides a 3-dimensional vision of the structures (rotation, zooming in and out) and the structures can be manipulated

through this graphical interface. The function responsible is *view()* and it takes the structures name as an argument. In the figure below bicyclobutane molecule can be seen in the graphical user interface.
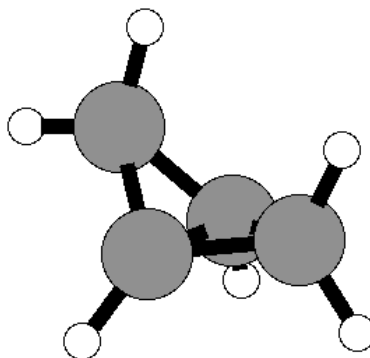

Figure 16. Bicyclobutane molecule shown by the view function of ASE.

Finally, ASE provides input/output functionalities with functions *read()* and *write()*. Write function, creates a file with a chosen format and writes the information of the structure to this file. Read function has the similar but reverse capabilities where it can input information from a file with different formats into ASE. It should be noted that although read and write functions can support a lot of different formats, not all formats are readable and writable simultaneously.

3.3 LAMMPS

Even though ASE is able to run molecular dynamics simulations, LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) is chosen to be used as the molecular dynamics program for this project. LAMMPS is a program specifically developed for classical molecular dynamic simulations by Sandia National Laboratories and Temple University. LAMMPS is developed in C++ and is open source. [20]

LAMMPS was initially released in 1995. In this project the most recent stable release was used, which was released in 11 August 2017. Despite LAMMPS is designed to run on parallel computers, using Message Passing Interface (MPI), it can efficiently run on

desktops or laptops. Another important aspect of LAMMPS is that it runs from an input script.

In terms of what LAMMPs cannot do is that it cannot operate via a GUI or build the molecular structures. The structures must be built by external tools, hence the use of ASE in this project. In addition, LAMMPS cannot visualize the molecular dynamics simulations or plot the data that user chooses to output. External tools will be used to accomplish these tasks. [20]

To run a simulation on LAMMPS, two main components are needed. An input script from which the LAMMPS read the parameters about the simulation and call the necessary modules, and an input structure which defines the structure and its initial conditions for the simulation.

3.4   LAMMPS Input Script

LAMMPS perform the simulation by reading the input script one line at a time. Once all the tasks that were read from the input script are finished, LAMMPS terminates. Nearly all commands in LAMMPS have a default value it will take if it is not modified explicitly by user. Although in the most of the cases the arrangement of the commands can be irrelevant, there are some exceptions [20]. Figure 17 depicts the first example.

```
timestep    2

run         50

run         100
```

```
run         50

timestep    2

run         100
```

Figure 17. Two different run commands

The first example, one on the left, the time step is defined to be 2 femtoseconds. This specific time step is then used on two simulations. The first is run for 50 time steps and then the second for 100 time steps. In the second script, the default time step is used for a simulation of 50 time steps, then the time step is changed to 2 femtoseconds and a

simulation of 100 time steps is run. Naturally, the output of these simulations would be different.

There are also other logical problems that might occur in the input script. For instance, unexpected behavior or errors could be caused by trying to change some parameters on atoms that are yet to be defined. On a similar note, if a command needs a value as an input from another command, the latter would have to precede the first one.

In the input script if a line is not blank, it is evaluated as a command. While LAMMPS commands are case sensitive, and completely lower-case, upper-case letters can be used for user's own variables. If a command becomes too long to fit on a single line, user can use the "&" character to break it into two lines and LAMMPS will interpret is a single command. Any line starting with the character "#" is interpreted as a comment.

Input script of a LAMMPS simulation can be broken down into 4 different sections. The first section would be the initialization. In the initialization, the parameters for the structure, atoms and whole of the simulations are defined. Second section is where the definition of atoms is made. In the case of this project, the atoms will be defined by reading an external .XYZ file.

After the definition of the structure and atoms, specific settings for the simulation need to be made. In this section of the input script, not only the force fields and other simulation related settings are established, also the output options are set. The last section will then run the simulation. If a molecular dynamics simulation is needed to be run with different settings, the last two sections can be repeated.

3.5   Related LAMMPS commands

The very first command that needs to be defined is the *units.* This command appoints the units that will be used in the calculations, output, log and files. In this project we used the units of the *metal.* This defines the units as following, although not limited to;

- distance = Å

- time = picoseconds

- energy = eV

- mass = grams/mole

- force = eV/Å

The commands *dimension* and *boundary* sets the dimensions of the simulation box in which the structure will be in and simulated. Simulation box can be either two or three dimensional. The lengths of the simulation box are read with *read_data* command.

Read_data command is how .XYZ file of the structure that was generated by ASE, is read into LAMMPS simulation. The format of this file is important as LAMMPS expects it to be a certain way. ASE does not output the .XYZ file in a format that LAMMPS is able to read. Therefore, a small bash script was used to parse and convert this file to the format LAMMPS expects it to be.

Similarly, to the input script, LAMMPS reads the input structure line by line and treats any line starting with "#" as a comment line. The input structure should start with the number of atoms in the structure. It should then give how many different atom types are in the structure and their masses and all other relevant information such as velocities, bonds, angles or dihedrals about the structure that will be simulated. Additionally, it should have the three vectors that would constitute the simulation box as mentioned earlier.

The style of the atoms, meaning, what features and behavior the atoms will show has to be defined. For example, simulating bio-molecules or a liquid system would call for different calculations. This description is done by the *atom_style* command. *atom_style* should, precede the *read_data* command.

Data about the thermodynamics of the system can be periodically put out during the simulation for post simulation processing intentions. Parameters that will be output can be determined through *thermos_style* command. Subsequently, using the command *thermo*, period of this output can be set in terms of the timestep used in the simulation.

For instance, if the timestep is 1 femtosecond, and thermodynamic data is output every 1000 timesteps, information will be output every picosecond.

Timestep of a simulation very critical. As the value of the timestep gets smaller, the resolution of the simulation will be larger. If two simulations with same amount of computing budget would have different timestep values, the simulation with the smaller timestep value could simulate the structure for a shorter amount of time but with more resolution. Additionally, if a simulation does not have enough resolution, critical events during the simulation might be completely missed or interpreted incorrectly. Timestep of simulation in LAMMPS can be set by the command *timestep*. The unit of the time will depend on what units were chosen.

Next parameters that need to be set in the simulation is to decide on the interatomic potential that will be used in the simulation. In order to do this, the command *pair_style* command is used. Even though the command name suggests that it can only set the formulas for pairwise interactions, it includes many-body potentials such as Tersoff potential, which was used in this project.

Coefficients for the potential formulas can be set manually or read from a file. *pair_coeff* is the command responsible with this task. *pair_style* command has to be precede *pair_coeff* command. The simulation box also must be defined before potentials and coefficients defined.

For visualizing purposes, the information on the atoms needs to be output periodically. This is achieved by *dump* command. With this command, myriad of different information about each atom can be stored in a file periodically.

The last two important commands are *fix* and *run*. Fix covers any action that will be imposed on the atoms in each timestep. The atoms these operations will be applied need to be grouped in different IDs, if there are more than 1 type of atoms. Then different updates can be done on them. These include from temperature to position changes to applying some constraints to boundary conditions. Lastly, the simulation can be run for given amount of timesteps by the command *run*.

3.6   Visualizing the Results

By getting the coordination information of atoms from the LAMMPS in the desired slices of time, a visualization of the results could be achieved. For this purpose, Open Visualization Tool (Ovito) will be used. Ovito is a free and open source software available for different platforms. [21]

Ovito was designed to help scientists with visualization and analysis of particle simulations by Alexander Stukowski in Darmstadt University of Technology. The program was developed in C++. [21]

Ovito can parse the output of the *dump* command of LAMMPs directly and render it using OpenGL. To use Ovito, the positions of the atoms must be output with *dump* command in LAMMPS. Ovito then uses this coordinate information to draw the molecule. Information in each of the timesteps would consist 1 frame. Each of the frames could be used for analysis or could be turn into an animation with a specific frame rate which can be set.

As the energy of the system increases, the vibration and the translation of the atoms gets more erratic. When this sort of chaotic behavior becomes the case, it was observed that analyzing the results with the Ovito was not possible as bonds could not be seen. Therefore, no patterns were available to for analysis. The figure below demonstrates how a frame in Ovito looks as well as demonstrating how chaotic the structure might look.
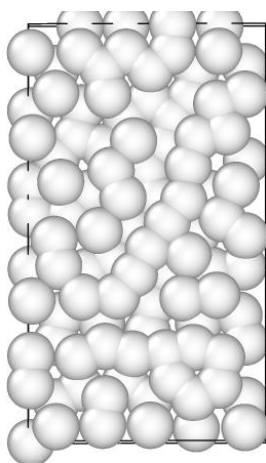

Figure 18. A sample frame from Ovito

As it can be seen, it would be impossible to analyze a frame like this. It looks completely random to make any educated guesses on what is happening in the structure. To be able to make better observations, visualization library of the ASE was used as it can display what is taking place in the structure internally. Figure 19 demonstrates this.
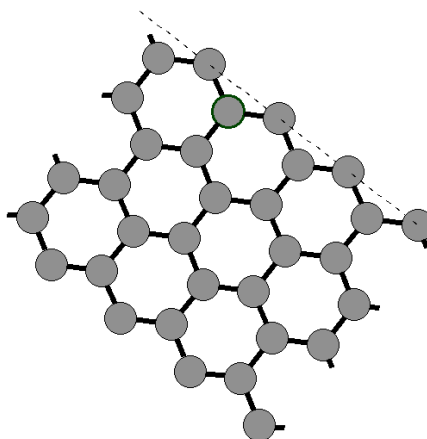
Figure 19. GNR with bonds shown.

A graphene nanoribbon is shown in the Figure above. The bonds can be seen in the Figure so when the structure start behave unusually, the bonds could be traced frame by frame if necessary for observation purposes. Although ASE's IO library is not able to produce an output that would be compatible with the input of LAMMPS, it can read a LAMMPS *dump* output.

4   Results

This section of the document will report the observations made on the simulations. Two of the simulations will be discussed in detail as clear observations were able to be made from hem. With the exception of the two simulations discussed below, simulations showed no recognizable patterns. The structures either showed no change at all or completely disintegrated.

One of the first observation of the simulations was the oscillation of individual atoms as energy was given into the system as heat. Since each atom in the structure started with zero velocity, it meant the system was at absolute zero. As energy was introduced into the system, individual atoms started to oscillate, as well as the structure as a whole. Oscillation of the structure as a whole was especially clear in the graphene nanoribbon simulations. Although the carbon nanotube showed an oscillating behavior in the form of getting compressed and relaxed, it was not as clear as the GNR.

In both of the simulation below, to keep the temperature of the system stable during the simulation, a Nose-Hoover thermostat was used. The thermostat algorithm is used with the *fix* command in LAMMPS since updates needs to be done on each atom for each timestep the simulation runs. This update consists of changing the velocities by linking them with a coefficient that is recalculated each iteration.

Nose-Hoover style thermostat in LAMMPS require a damping parameter which defines how fast the temperature is relaxed. The value is in time units. Since *units metal* command was used in this simulation, the time unit is picoseconds. Damping parameter used for this simulation of graphene was 0.3 which equates to 300 fs was given to relax the temperature. For the carbon nanotube, this parameter was set to 0.1 picoseconds, therefore 100 fs was given for the relaxation of the temperature. Due to the internal workings of the thermostat algorithm, an oscillation around the goal temperature was expected and it was observed.

4.1   Graphene Nanoribbon

Graphene nanotube in the simulation that will be displayed below is of type armchair and its size is six by six. This type of graphene nanoribbon without a vacancy contain 144

carbon atoms. Since a vacancy was introduced into the structure, it contains 143 atoms. In the figure below GNR can be seen before simulation from both front and side views.
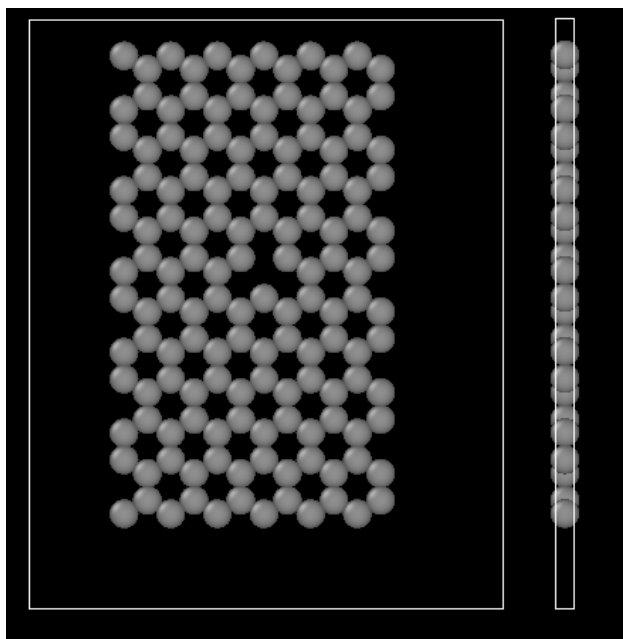

Figure 20. Front and side view of graphene.

Since graphene is a 2-dimensional material, the simulation was also run in only 2-dimensions. The Y-axis of the simulation was eliminated, and boundaries of the Y-axis set to be fixed. Timestep used for the simulation was 1 fs and the information on the velocities of the atoms were output every 20 timesteps, meaning every 20 fs. This makes each figure below to have 20 fs resolution in between them.
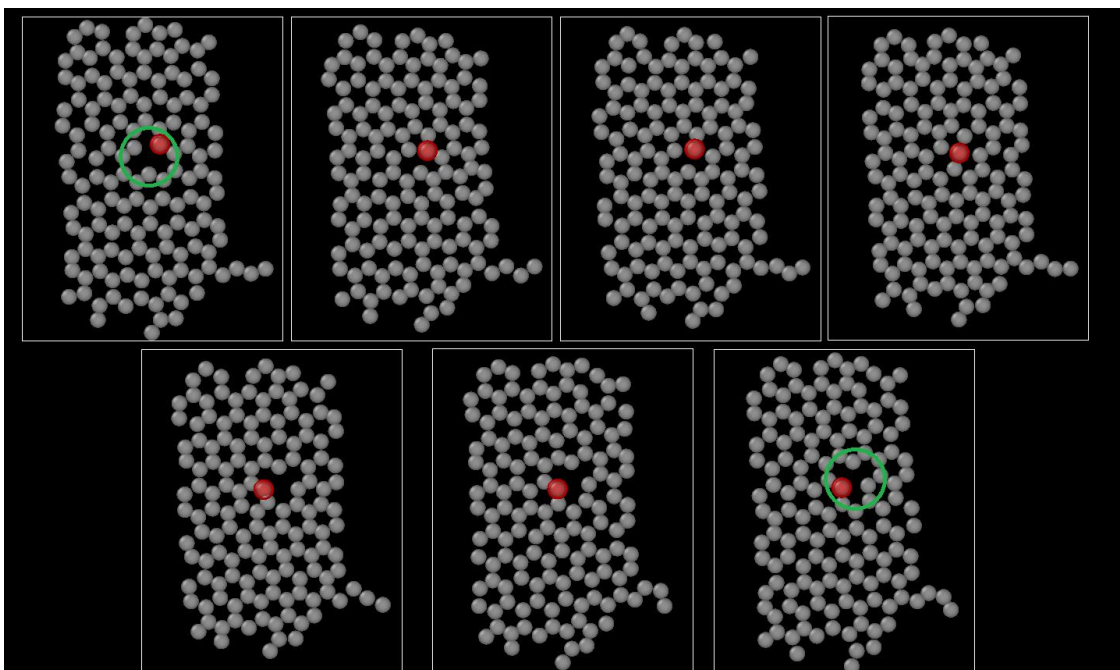
Figure 21. Diffusion in graphene nanoribbon

It can clearly be seen in the Figure 21 that the red highlighted atom with the index 84, diffuses to left and leaves its original lattice vacant. The diffusion was observed to happen at 1.63 fs after the start of the simulation which also means at 163th iteration. At that point, the temperature of the system was recorded at 3289.0064 K.

The graph below displays the change of temperature (K) in the system with respect to time in picoseconds. Mentioned fluctuation around the goal temperature can be observed. The system reaches its stable temperature about 3 picoseconds after the start of the simulation. A very high peak around 9000 K can be noticed.
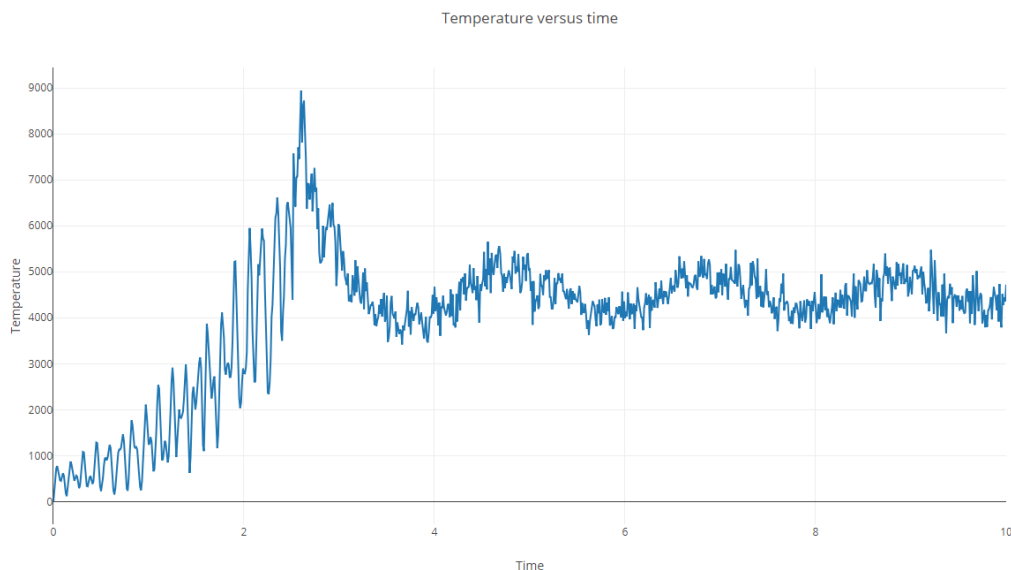
Figure 22. Temperature versus time graph of GNR simulation.

At this point, why the temperature reaches at that point is not clear for the author. The influences of the temperature of the system to reaching an unintended value is also unknown. Although explanation to answer the question of why was sought, the attempts have been unsuccessful.

Another unexplained observation is a string which consist 4 carbon atoms form in the right bottom corner of the GNR. After visual inspection, it was identified that those atoms move from the top side of the structure to bottom due to periodic boundary conditions. However, it is unknown if that behavior would be seen in nature or what could have caused it in simulation.

4.2 Carbon Nanotube

Carbon nanotube that will be demonstrated below is of type armchair. Without the vacancy, it would consist of 96 Carbon atoms. However, with the vacancy created, it contains 95 atoms. In the figure below, you can see the carbon nanotube in the start of the simulation, at 0 degrees Kelvin. Both of the figures represent the same structure, from different angles. The image on the right, displays the structure from a point of view which is looking straight down from the z-axis. Therefore, the vacancy cannot be seen as there is no perspective. The structure was rotated around the Z-axis in the picture at the left, so the vacancy can be seen. Red circle highlights the point of the vacancy.
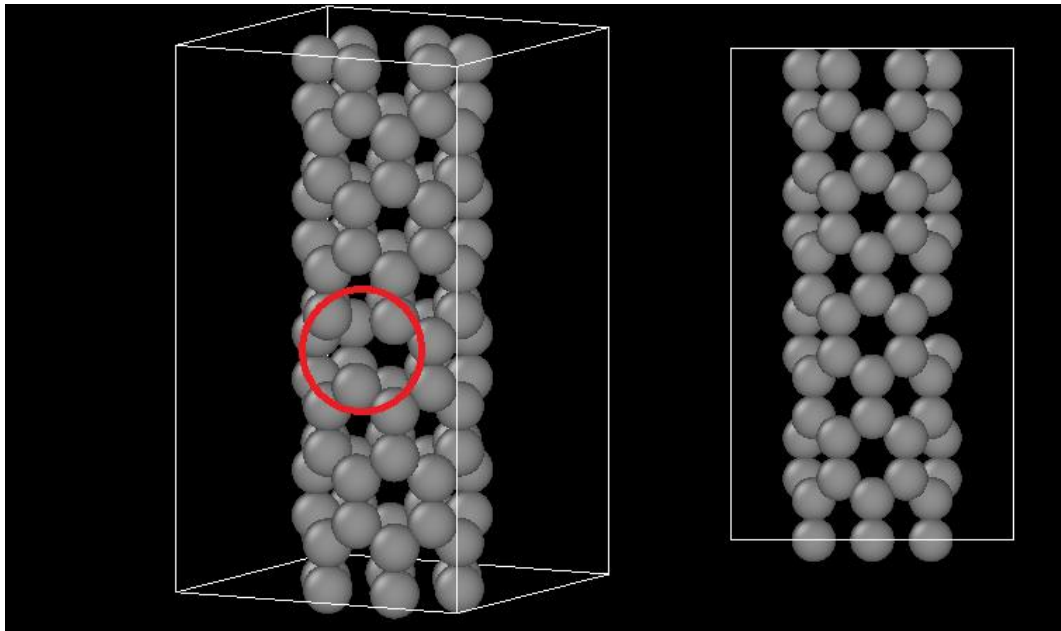
Figure 23. Carbon nanotube at the start.

The carbon nanotube in Figure 23 has *n* and *m* values of 6 and 0 respectively. The length parameter of carbon nanotube is 4. Each timestep of the simulation is 1 fs and the structure is heated up from 0 K to 3500 K. It took around 65 fs for the structure to reach 3500K approximately and the temperature of the system hovered around this temperature for the rest of the time in the simulation. A total of 10000 timesteps were simulated. Therefore, the total time of simulation was 10 fs. Interatomic potential used for the simulation was Tersoff potential.

First observation of a diffusion happened 4.32 ps into the simulation. At this point, the temperature of the system was 3859.2253 K. In the figure below, the diffusion is shown. The atom with the index 4 gains a velocity in the positive direction of the *z*-axis and jumps into the vacancy, leaving its original lattice point empty.

The figure below is based on the position and velocity information that was printed by the simulation. The time difference between these adjacent frames 40 fs.
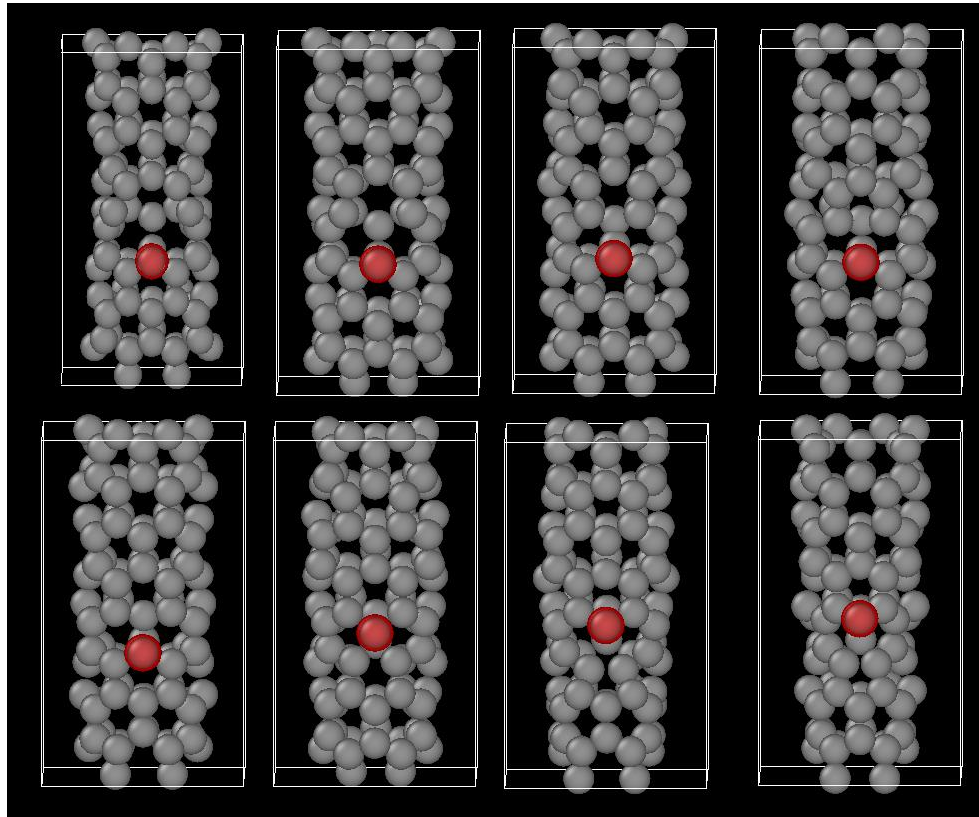

Figure 24. First diffusion

Even though the temperature of the system was kept relatively stable, 2 more diffusions were observed in the structure. First of these 2 diffusions took place at 5.72 ps into the simulations. At that point, temperature of the system was 3981.6811 K. In the figure below, vacancy which occurred in the first diffusion is highlighted by a green circle.

 The highlighted atom with index 42 moves in the negative x direction and positive z direction to fill the spot of the vacancy. The resolution of the output for the information on atoms were 4 fs. Since adjacent frames were not used like in the Figure 24 the time difference between frames are more than 4 fs.
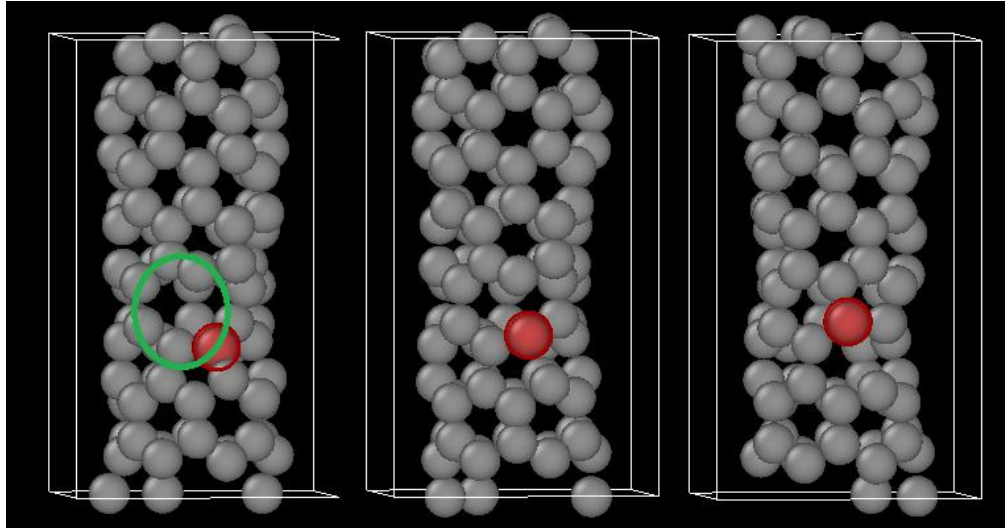
Figure 25. 2nd diffusion.

The last observation of a diffusion in system was 8.54 ps after the start of the simulation. The temperature at that point was 3983 K. The highlighted atom with the index 45, moved in negative z direction and negative x direction to fill the vacancy created by the 2nd diffusion. The diffusion is once again highlighted with a green circle in the figure below. As in Figure 25, the frames are not adjacent.
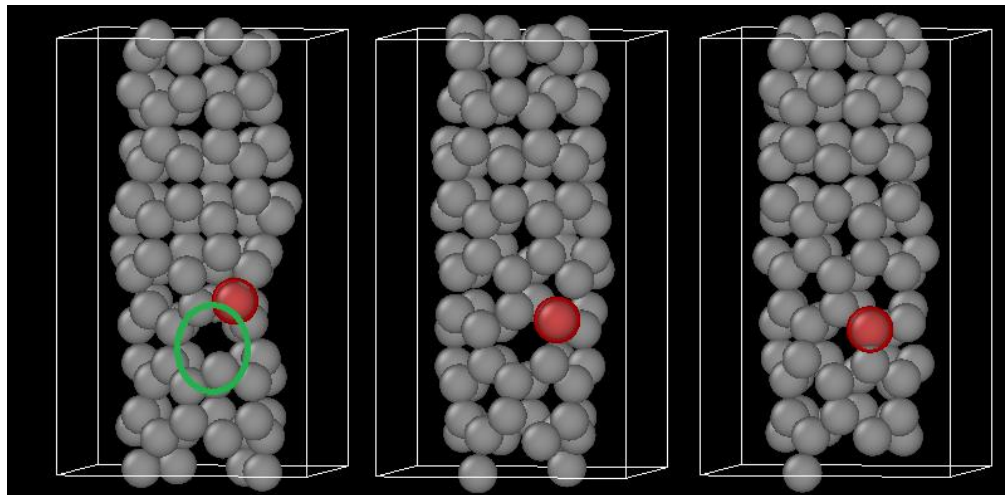

Figure 26. 3rd diffusion in the carbon nanotube

Although the system was intended to be at a stable temperature after reaching to 3500 K, it was noted by the thermodynamic output that the system fluctuated around this value after it reached it.

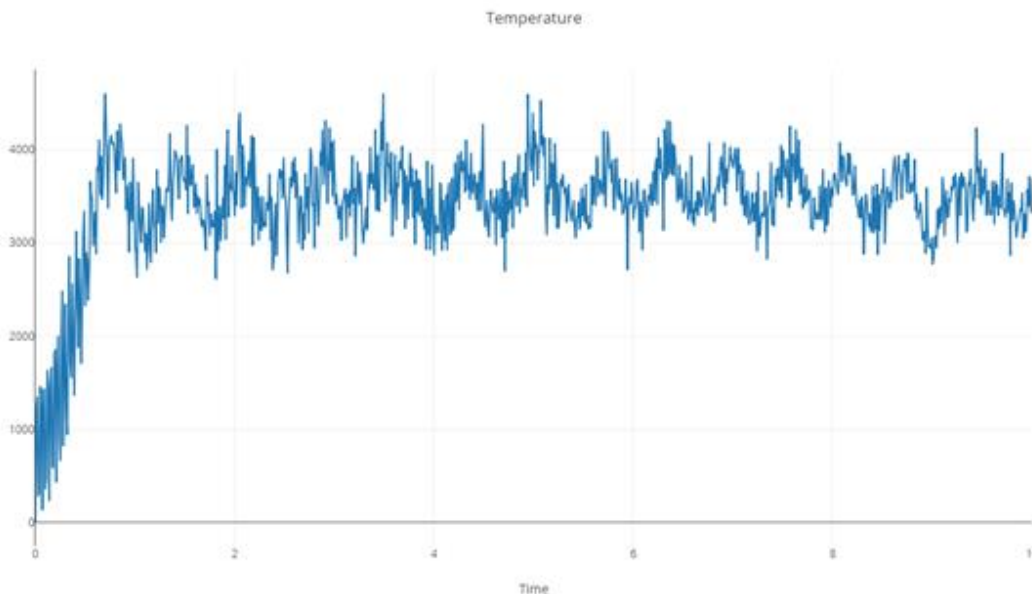The graph below shows the change in temperature with respect to time.

Figure 27. Temperature versus time graph of carbon nanotube.

The graph above starts from the timestep 0. Vertical axis shows the temperature in Kelvin and horizontal axis shows the time in picoseconds. It can be observed from the graph above that it takes approximately 1 picosecond for the system to reach 3500 K. Expected fluctuation around the goal temperature of 3500K can be easily observed.

Finally, an observation on the time it took for diffusion to take place after it starts varied between the diffusions. Still, it is hard to say exactly how much time it took for the diffusions since the resolution is 4 fs. Nevertheless, it was observed that the 2nd diffusion took the longest amount of time with 0.16 ps where 1st and 3rd diffusions took about approximately the same amount of time with 0.12 ps.

5   Reflections

During the thesis project, considerable amount of difficulties was encountered. The main one was the lack of knowledge on natural sciences related to the topic. Although the author took the thesis topic knowingly, the depth of knowledge that is needed to complete the project was more than author originally thought. At first, the project was thought to be an ICT project with a material science aspect by the author. However, it was realized that this type of project is more of a material science project with an ICT aspect.

Firstly, as a student of information and communication technology, author did not have the experience or knowledge to generate a structure with parameters such that it would meaningful to run a simulation on. Secondly, realizing the differences between the simulations and results that represent the nature of reality, and ones that does not also demands deep insights in physics and chemistry.

Another challenge during the thesis project was finding the relative information about molecular dynamics and understanding it. The information that can be found was either too superficial or too deep into the subject that required a lot of prerequisite knowledge on the subjects that makes up the foundation of the molecular dynamics.

Due to the challenging and diverse nature of this project, the author had to study different topics from different disciplines. This, while being complicated, increased the knowledge of the author. Furthermore, author gained experience on working on topics which heavily rely on the theory.

As voiced in the results section, there are observed behavior that the author could not explain. At this point, the reason why in some arrangement of the parameters did not cause vacancies to move or caused structures to disintegrate is also left unexplained. The fundamental assumption is that a spike in the temperature gives atoms enough energy for them to break their bonds completely and melt.

6 Conclusions

The purpose of this thesis was to inspect both the vacancy's and structure's behavior as a carbon nanomaterial is heated with a vacancy present. As expected from the literature review, moving of the vacancy was observed. A side purpose of this project was to show that an information and communication technology student would be able to understand the theory and tools of molecular dynamics enough to run and investigate the outcome, which the author believes to be achieved.

This thesis could be utilized as a tool by ICT students or professionals who would like to start working on molecular dynamics simulation. The reason is that the report introduces the necessary subjects from the ground level. Therefore, it could be a good starting point for individuals to expand their knowledge upon.

In terms of further improvements, unexplained phenomena in the simulations should be analyzed and clarified. Afterwards, structures that consist more atoms could be simulated for longer periods of time for the observation of flux of the vacancies. Furthermore, the number of vacancies in these larger structures could be increased, or their pattern in the structure changed, to investigate its effects by comparing them to each other.

References

1   Askeland D, Fulay P, Wright W. The Science and Engineering of Materials 6th edition. Stamford: Cengage Learning. 2011.

2   The University of Manchester. Graphene. 2018.URL: https://www.graphene.man-chester.ac.uk/ Accessed 03 June 2018.

3   Young H, Freedman R. University Physics with Modern Physics 14th Edition. Essex: Pearson Education Limited. 2016.

4   Fairfield F. The Electron Cloud. Letstalkaboutscience. 2012. URL: https://letstalk-aboutscience.wordpress.com/2012/02/16/the-electron-cloud/ Accessed 28 May 2018

5   Robinson W, Odom J, Holtzclaw H. General Chemistry 10th Edition. Boston: Houghton Mifflin Company. 1997.

6   Covalent Bond Examples. TutorVista.com. 2018 URL: https://chemistry.tu-torvista.com/inorganic-chemistry/covalent-bond-examples.html Accessed 27 May 2018.

7   Helmenstine A. Ionic Bond Definition. ThougtCo URL: https://www.thoughtco.com/definition-of-ionic-bond-604536 Accessed 27 May 2018.

8   Chegg Inc. Sigma and Pi Bonds. 2018. URL: https://www.chegg.com/homework-help/definitions/sigma-and-pi-bonds-6 Accessed 27 May 2018.

9   CommerceGurus. Graphene Structure. 2015. URL: http://graphenewhole-sale.com/graphene-structure/ Accessed 27 May 2018.

10   La Fuente J. Graphenea Inc. 2018. URL: https://www.graphenea.com/pages/gra-phene-properties#.WwsSYe6FOUl Accessed 27 May 2018.

11   Physics Reimagined. Graphene. 2018. URL: https://toutestquantique.fr/en/gra-phene/ Accessed 27 May 2018.

12   Rafiee R, Pourazizi R. Evaluating the influence of defects on the young's modulus of carbon nanotubes using stochastic modeling. 2014. URL: http://www.sci-elo.br/scielo.php?script=sci_arttext&pid=S1516-14392014000300031 Accessed 27 May 2018.

13   Booker R, Boysen E. Nanotechnology for Dummies. Hoboken: Wiley Publishing Inc. 2005.

14   Kaushik B, Majumder M. Carbon Nanotube Based VLSI Interconnects. Springer India, 2015.

15   Purdue University. More about Material Science. URL: http://chemed.chem.purdue.edu/genchem/topicreview/bp/materials/defects3.html Accessed 27 May 2018.

16   Major Differences. Difference Between Schottky Defect and Frenkel Defect. 2018. URL: https://www.majordifferences.com/2013/02/difference-between-schottky-defect-and.html#.WwC66u6FOUk Accessed 27 May 2018.

17   Allen P. Introduction to Molecular Dynamics Simulation. John von Neumann Institute for Computing. 2004.

18   Barrett C, Nix W, Tetelman A. The Principles of Engineering Materials. Prentice-Hall. 1973

19   ASE-developers. Atomic Simulation Environment. 2017.URL: https://wiki.fysik.dtu.dk/ase/ Accessed 27 May 2018.

20   Sandia Corporation. LAMMPS Documentation. 2018. URL: : http://lammps.sandia.gov/doc/Manual.html Accessed 27 May 2018

21   Stukowski A. Ovito User Manual. 2017. URL: https://ovito.org/manual/ Accessed 28 May 2018.