Brent Carey

Simulation and Analysis of Vacancies in Carbon Nanostructures

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Thesis

28 May 2018

Abstract

Author Title	Brent Carey Simulation and Analysis of Vacancies in Carbon Nanostructures
Number of Pages Date	40 pages + 1 appendices 28 May 2018
Degree	Bachelor of Engineering
Degree Program	Information Technology
Professional Major	Smart Systems
Instructors	Sami Sainio, Principal Lecturer

The discovery of the Buckminsterfullerenes in 1985 marked the beginning of the research field in carbon nanostructures. Since then, several more carbon allotropes existing in the nanoscale have been discovered, most notably the carbon nanotube and graphene. Due to the small scale of these materials, research is difficult, and production requires state of the art equipment to create even small samples. However, study has only increased with the accumulation of desirable properties these materials hold, boasting thermal and electrical conductivity, as well as immense strength. As the properties of carbon nanostructures continue to emerge, numerous research groups in academia have begun studying the synthesis and applications of these versatile materials.

This thesis aims to provide a supplementary insight to how the behavior of a vacancy in carbon nanostructures changes when energy is added to the system. While research of this kind is often done in a laboratory, this analysis is conducted in a digital environment, examining the relevance of simulations in comparison to ideal results.

The two primary software packages used for the simulations are Atomic Simulation Environment (ASE) and Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS). ASE provides tools to build, visualize, and export atomistic structures. These tools were used to generate and prepare the carbon nanotube and graphene nanoribbon for the forthcoming simulations. LAMMPS imports the exported ASE structures and executes the molecular dynamics simulations using the Tersoff potential for interatomic interactions.

Digital experiments regarding phenomena in natural science is enhancing how modern research is conducted. They provide both valuable insights to experiments and can model hypothetical scenarios that are difficult to physically execute. Simulations do have limitations such as system size and complexity of the model. The molecular dynamics simulations conducted in this project can be ran on a personal computer; however, some systems require considerable computation, needing the computing power a supercomputer. While there will always be need for the laboratory, simulations can be an economical and effective means for augmenting research.

Keywords	Simulation, Carbon Nanotube, Graphene, ASE, LAMMPS

Contents

Appendices

List of Abbreviations

CNT	Carbon nanotube.
GNR	Graphene nanoribbon.
C	Carbon.
ASE	Atomic Simulation Environment.
LAMMPS	Large-scale Atomic/Molecular Massively Parallel Simulator.
GUI	Graphical User Interface.

1 Introduction

Carbon nanostructures are not particularly new to the world of science, in fact the field of research is already more than 30 years old, marking the discovery of Buckminsterfullerenes in 1985 [1]. However, their impressive properties have recently rendered them an important topic in materials research. Possibly, the most coveted aspect of these nanomaterials is their diversity in potential applications. These applications cover a large spectrum, including electronics, optics, aerospace, medicine, and many other fields of technology.

The motivation for this thesis topic stemmed from the research being conducted at Aalto University's Department of Chemistry and Materials Science. Proferssor Tomi Laurila's research group is investigating the use of carbon nanomaterials in designing new forms of biosensors. While much of the research has been done in a laboratory environment, simulations have been used to support research and provide interesting insights. By incorporating simulation software and various algorithms it is possible to perform digital experiments that augment the study of natural phenomena.

The goal of this thesis is to analyze vacancy defects in carbon nanomaterials using simulations in a purely digital environment. Two allotropes of carbon are used in this research: the carbon nanotube (CNT) and the graphene nanoribbon (GNR). Once these structures are created and prepared, they are passed to the simulation environment where energy is added to the system. The simulation results are processed, analyzed, and then presented in this document. In addition to conducting the aforementioned experiment, an underlying project goal is to demonstrate the value of simulations and digitalization in science.

2 Theoretical Background

This chapter introduces the topics required for understanding the molecular dynamics simulations in the following sections. Firstly, a general concept of carbon and its chemical properties must be established. Then, it is possible to discuss the allotropes of carbon, or the different physical forms carbon can exist in. Lastly, this chapter covers vacancies; including how they are formed and their effect on the structures they are present in.

2.1 Introduction to Carbon

Carbon (C) is truly one of the most fascinating elements known to mankind. Its assortment of chemical properties makes it an extremely versatile element, allowing it to exist in vastly different forms. While carbon itself wasn’t recognized as an element until the late 18th century, its usage by humans reaches back into antiquity [1]. For example, wood, which is greatly composed of C, has been used for tools, buildings, and heating since prehistoric times [1,2]. And steel, a material crucial to transportation, infrastructure, and weapons throughout history, is fabricated by introducing a small amount of C to iron. C is one of the most abundant elements in the known universe and is considered one of the building blocks of life [2,1]. C is known to occur in countless compounds (organic and inorganic) and as you can see from Figure 1, following oxygen, C is the second most abundant element in the human body [3].

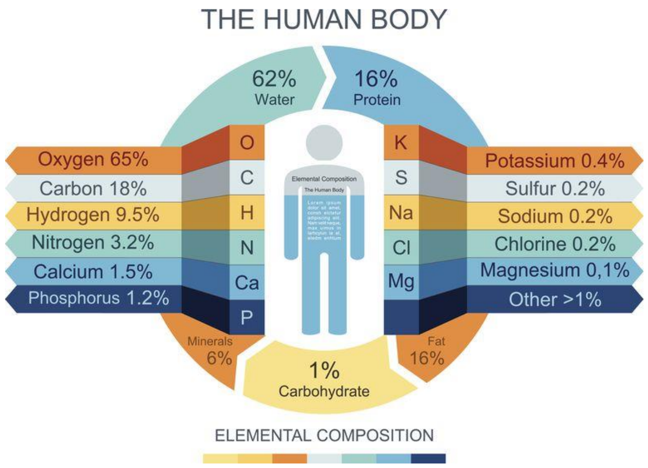


Figure 1: Elemental composition of the human body. Copied from Helmenstine (2017) [3]

2.1.1 Atomic Structure

Carbon’s interesting forms and their diverse material properties can be understood when analyzing C at a smaller scale. This scale is so small that it is possible to even see the building blocks that make up a single atom, or unit, of C. These building blocks are called neutrons, protons, and electrons. And with a specific combination of these particles, it is possible to represent any element. This “specific combination” is called the atomic structure of an atom. Figure 2 contains an illustration of a C atom’s atomic structure, where the blue “+” orbs are protons, the red orbs are neutrons, and the green “-” orbs are electrons. The subatomic particles “electrons” and “protons” have negative and positive charges respectively, while neutrons contain no net electrical charge.

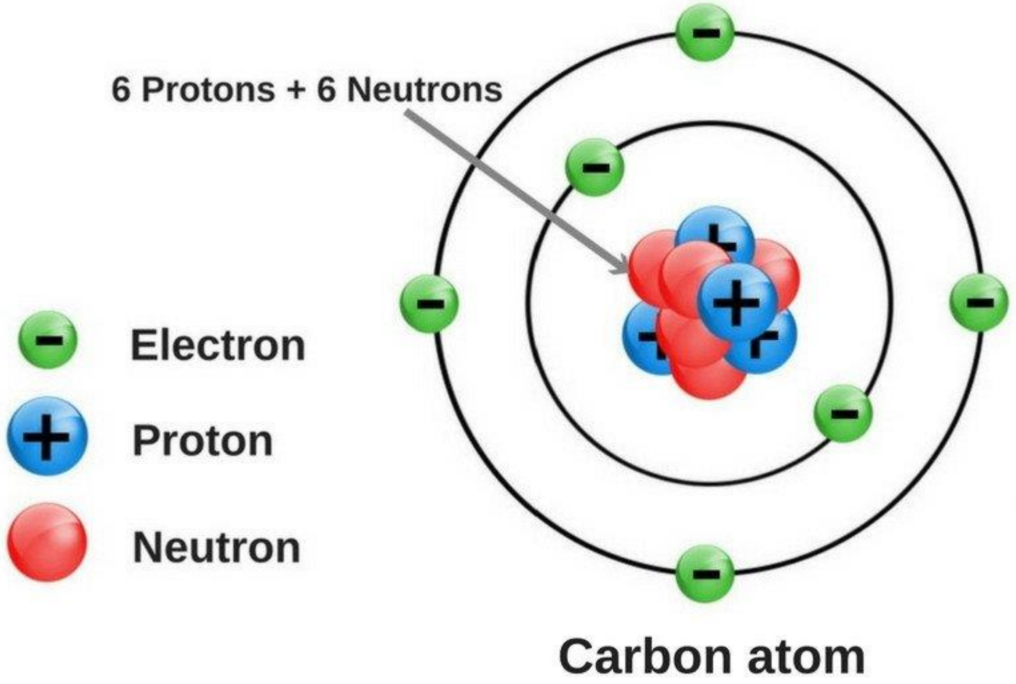


Figure 2: Atomic structure of C atom. Copied from Ashish (2017) [4]

The relatively dense center of the atom is called the nucleus and is composed of neutrons and protons (see Figure 2). The nucleus contains the bulk of an atom’s mass and has a net positive charge from the presence of protons. Negatively charged electrons are held in orbit around the nucleus due to electrostatic attraction. Figure 2 represents an electrically neutral C atom, meaning that there are 6 electrons, 6 protons, and 6 neutrons. The number of protons in an atom’s nucleus determines the element’s atomic number, which is also a unique label for that element. If an element is electrically neutral, then it has the same number of electrons and protons. [5,21-25.]

Electrons in an atom can be broken into two categories: core electrons and valence electrons. The category of electron can be distinguished based on where within the atom it resides. Electrons exist in circular areas around the nucleus called energy levels, or shells. The electrons that reside in the outermost energy level, or valence shell, are called valence electrons, and are the primary participants in bonding and chemical reactions. All the electrons that reside in the inner energy levels are called core electrons, which seldom participate in reactions. Figure 2 shows that C has 4 valence electrons in the outermost energy level and 2 core electrons in the inner energy level. While the scope of this research doesn’t require the in-depth knowledge of how electrons populate different energy levels, it is important to understand the valence shell and the electrons that exist there. [5,27-28.]

The amount of valence electrons in an atom determines how reactive an element is. Elements with a full outer energy level (no valence electrons), like the inert gases, are non-reactive. Elements tend to want a completely full or completely empty valence shell. Figure 3 plots the position of an element on the periodic table (or the amount of valence electrons) vs its electronegativity. Electronegativity refers to an atoms tendency to gain electrons. On the other hand, an atom that tends to give up electrons, or has a nearly empty valance shell, is called electropositive. For example, in Figure 3, Fluorine has 7 electrons in its outer energy level, needing to gain only one electron to achieve a full valence shell. Fluorine is very reactive or electronegative because it readily accepts electrons. A C atom, has 4 electrons in its outer energy level and therefore would need to gain (or share) 4 more electrons to achieve a full valence shell. [5,28.] While C isn’t extremely electronegative, having a value of about 2.5 in Figure 3, the 4 “empty” electron spots play an important role in its bonding and subsequent properties at a larger scale.

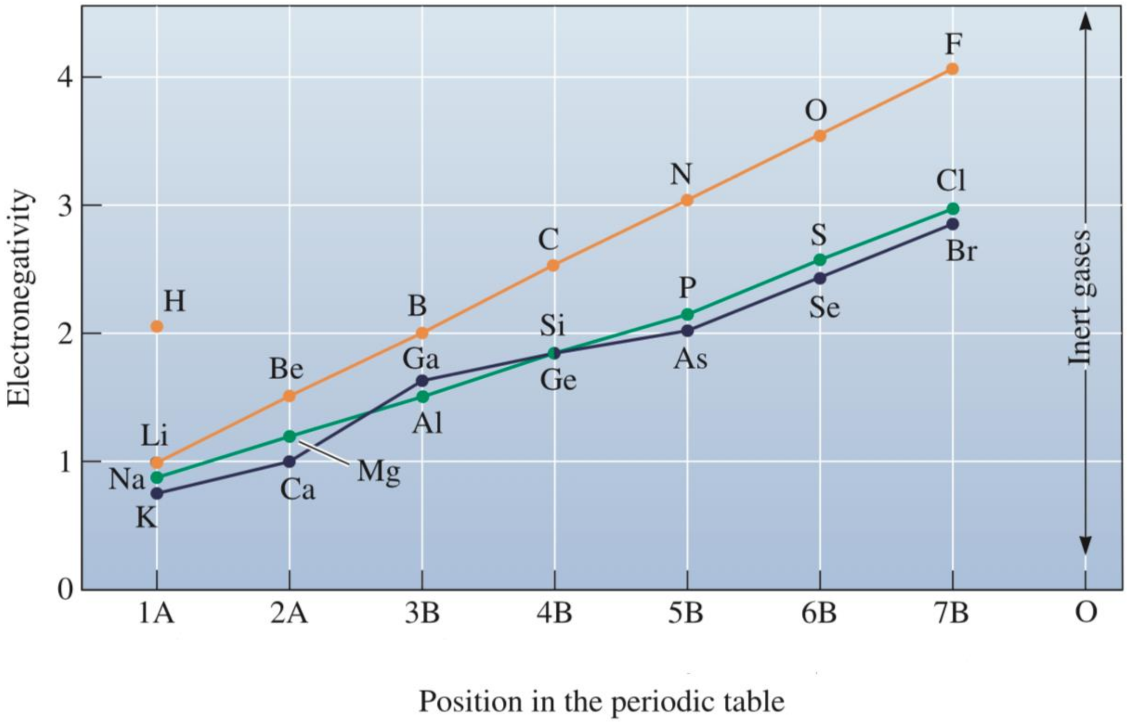


Figure 3: Electronegativities relative to position on periodic table. Modified from Askeland and Wright (2016) [5]

2.1.2 Bonding

Atomic structure greatly influences how an atom interacts with other atoms. These interactions are called bonds, of which there are 4 different types. 3 of these bond types are called primary bonds, and include metallic bonds, ionic bonds, and covalent bonds. Primary bonds are considered to be strong, and involve the transfer or sharing of valence electrons between atoms. Secondary bonds are relatively weaker and include the 4th type of bond, the van der Waals bond. [5,31.] Bonds are the mechanism in how different molecules and compounds are formed, and therefore play a key role in the synthesis of materials regardless of scale.

While metallic, ionic, and van der Waals bonds are important and deserve consideration, covalent bonds are most applicable to this research. Covalent bonds are described as a bond formed by sharing valence electrons with one or more other atoms. For example, in Figure 4(a) C has 4 valence electrons but wants 8 to achieve a full outer energy level. A possible solution is for C to share its valence electrons with 4 nearby C atoms, forming

4 covalent bonds. As depicted in Figure 4(b), each covalent bond introduces a new electron into C's valence shell, allowing it to fill its outer shell. Before a covalent bond can be formed, the participating C atoms must first be aligned properly. This is called a directional relationship, where there must be a specific arrangement of atoms that forms material determined angles between them. In the case of C, its arrangement creates a tetrahedron, which has 109.5° angles between each covalent bond (see Figure 4(c)). [5,32.]

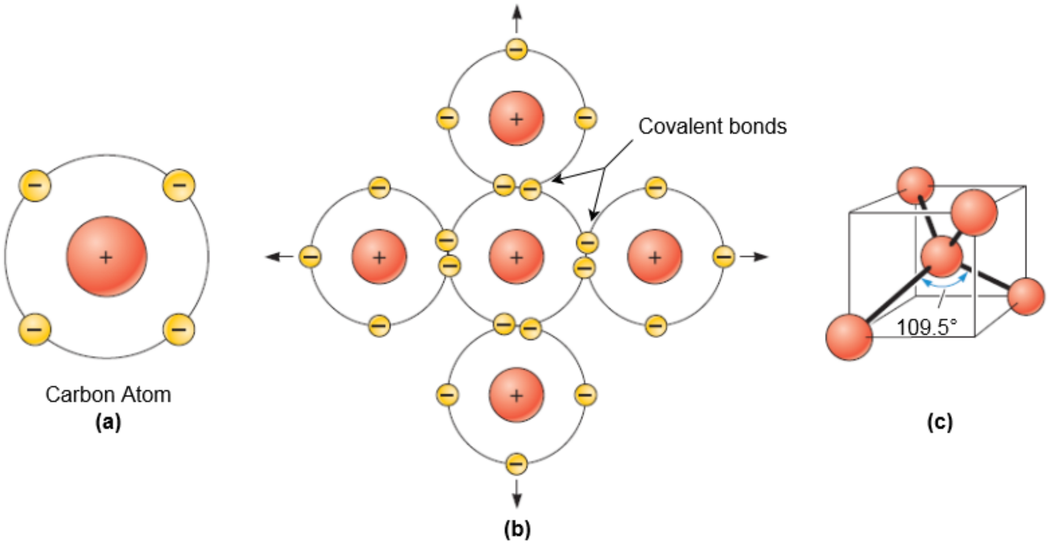


Figure 4: (a) Representation of C atom and its valence electrons. (b) C atom has formed covalent bonds with 4 other C atoms, giving it a full outer energy level. (c) C atoms arranged in a tetrahedron. Modified from Askeland and Wright (2016) [5]

Because covalent bonds are strong, materials comprised of them are also characteristically strong and hard. Covalently bonded materials also tend to have high melting points making them very applicable to high-temperature scenarios. While materials consisting of covalent bonds do share some properties, they can also be extremely different from each other. For example C, which forms covalent bonds, can exist in multiple different forms that exhibit very diverse material properties. Some of these forms and their properties are vital to this research and will be discussed in detail in the next section.

2.2 Allotropes of Carbon

Allotropy is used in materials science and chemistry to describe the phenomena of when pure materials have 2 or more atomic arrangements at different environmental conditions [5,44]. C has several different allotropes, or forms, that are synthesized at a variety of temperatures and pressures. Allotropes like diamonds and graphite are very familiar to us and are used daily in jewelry and pencil lead. On the other hand, allotropes like graphene, and nanotubes, have been discovered relatively recently, and their promising material properties have yet to penetrate everyday life. All C allotropes are composed of the same thing, pure C. However, these different forms exhibit widely different material properties. The atomic arrangement is the primary factor that determines these diverse materials.

The diamond is perhaps one of the most commonly recognized allotropes of C. As one of the hardest materials known to man, it’s highly valued, and used in a large variety of applications. Figure 4(c) showed the tetrahedron arrangement of C of which the diamond crystal is composed of. In Figure 5 you can recognize each tetrahedron, and imagine its structure repeating in every direction, creating the diamond we recognize. In a diamond, each C atom is covalently bonded to 4 other C atoms, producing a directional relationship as discussed in the previous section. Because the diamond is a crystal, the structure contains a long-range order, meaning that it regularly repeats the same atomic arrangement in all directions over a specific length scale. [5,40].

The fact that every valence electron, in every C atom, in a diamond participates in a covalent bond (ideally), greatly influences its macroscopic properties. Diamonds have an extremely high melting point and are excellent thermal conductors due to the strong bonds and stiffness. Diamonds are also excellent electrical insulators since all electrons outer electrons are being held rigidly in place by the bonds.

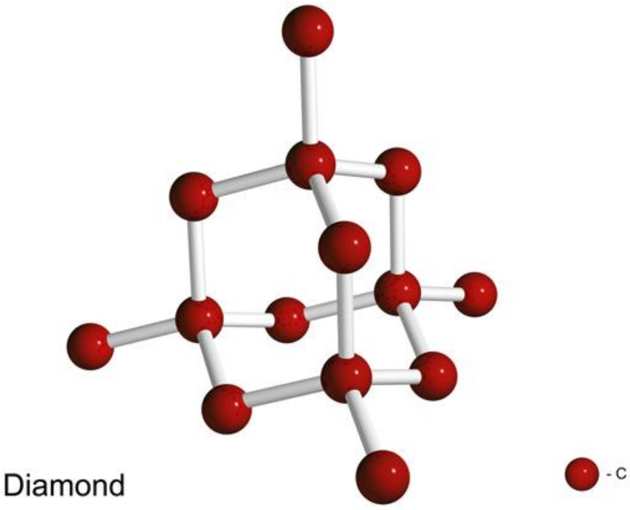


Figure 5: Atomic arrangement of diamond. Copied from Plait [6]

Graphite, diamond's less glamorous relative, is also made out of pure C, but as we know from writing with a pencil, the properties are very different. Figure 6 shows the atomic arrangement of graphite. Unlike the diamond, the C atoms are arranged in layers and are covalently bonded to 3 other C atoms instead of 4. This forms the hexagonal pattern relevant in Figure 6. The layers of graphite are held together by weaker van der Waals bonds and have a much greater distance than the close covalent bond. The combination of weak bond strength and long bond distance makes it easy for layers of graphite to sheer apart (think about writing with a graphite pencil). Because there are still many covalent bonds, graphite has a high melting point like diamond. However, dissimilar to diamond, graphite conducts electricity since atom's 4th valence electrons are not trapped in covalent bonds. [5,41-42].

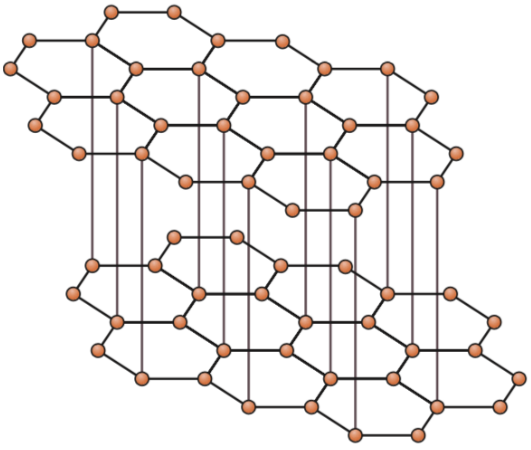


Figure 6: Atomic arrangement of graphite. Copied from Askeland and Wright (2016) [5,42]

2.2.1 Graphene

Graphene is a relatively new addition to C’s formidable allotropes and is one of the two structures investigated in this research. Since its first synthesis, nearly a decade and a half ago, graphene and the possible technologies it could produce has peaked interest in scientific communities all over the world. Science enthusiast and author Chris Woodford even opened a graphene related article with: “If the 20th century was the age of plastics, the 21st century seems set to become the age of graphene…” [7]. While the excitement around this possibly revolutionary material continues to rise, a method for synthesizing it on an industrial scale is still absent. Much more research needs to be appropriated towards achieving a deeper understanding of the material and finding a cost-effective means for its production.

2.2.1.1 Characteristics and Properties

Graphene is described as a single layer of graphite, meaning that it is a one-atom-thick sheet of C arranged in a hexagonal lattice as seen in Figure 7. It was first discovered at the University of Manchester in 2004 by professors Kostya Novoselov and Andre Geim. The two scientists somewhat stumbled upon the solution one Friday afternoon when philandering with a block of a graphite and a roll of Scotch tape. When they applied two pieces of tape to each side of a small sample of graphite and peeled them apart, a layer of graphite would be pulled from the bulk. By repeatedly using this method the graphite was reduced to a single atom thick layer, graphene. Since Novoselov and Geim’s discovery, laboratories all over the world still use this “Scotch tape method”, also known as the micromechanical cleavage technique, to acquire isolated samples of graphene. [8].

Being the first 2D material discovered, scientists were astonished by the huge variety of graphene’s material properties. Prominent graphene researcher Dr. Mikael Fogelstrom refers to graphene as “a material of superlatives” because it’s the best at almost every adjective used to describe it [9]. Since graphene is only one atom thick, it is the thinnest material ever, meaning that it’s only composed of surface. In addition, being a 2D material, it is also flexible and extremely light. Graphene is the strongest material known, being approximately 200 times stronger than steel. It also the best thermal conductor known to man and boasts an impressive amount of electrical conductivity. Lastly, graphene is nearly transparent, letting 98% of light through. [9].

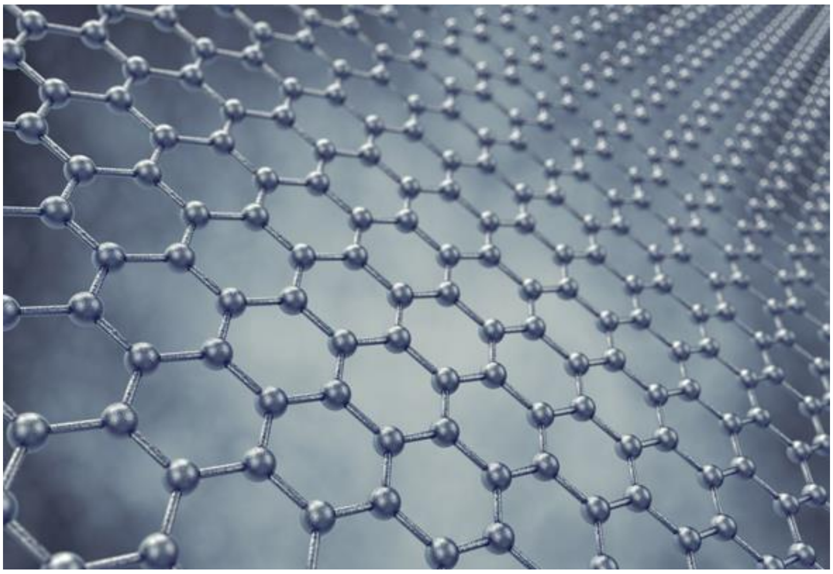


Figure 7: Section of graphene. Copied from Plait [6]

2.2.1.2 Applications

With all graphene's impressive properties, the possible applications are only limited by human ingenuity. It has the capability to not only replace many currently used materials but can also be used to synthesize completely new solutions. Once a suitable means of large-scale production is discovered, graphene research can be applied to nearly all fields of technology. For example, graphene could be a cheap and optimal water purification solution due to its impermeability. When a material is "impermeable", it means that small substances cannot pass through its structure, acting as a barrier for liquids and gas. By introducing small pores to a graphene membrane, just big enough for water molecules to pass through, the graphene "net" could block impurities, functioning as a filter [7]. Graphene could also be applied to composite materials, especially those used in the aviation industry. Materials that are strong and light are crucial to aerospace design and graphene is the best of both categories. It could even replace the steel used in an airplane's frame, making it lighter and more fuel efficient.

2.2.2 Nanotubes

The carbon nanotube (CNT) is another allotrope of C and like graphene exists on a very small scale. While it has been known and studied in the scientific community for a longer period than graphene, it is still a relatively new addition to the C allotrope family. CNTs have a variety of exceptional material properties, many of which are common with the other C allotropes. The CNT is the second nanostructure analyzed in this research.

2.2.2.1 Characteristics and Properties

CNTs are hollow, cylindrical structures consisting of pure C that typically exist on the nanoscale and microscale. The structures can range from 1 to 25 nanometers in diameter and can be several micrometers in length [5,44]. A CNT's structure is derived from a sheet of graphene that has been rolled up in a specific direction. A nanotube can be either single-walled, meaning that the tube surface is only 1 atom thick, or multi-walled, where the wall is several atoms thick. Multi-walled nanotubes can be thought to consist of multiple nanotubes nested within each other, with the largest being the outer most surface. This project will concentrate on single-walled nanotubes, so from this point on when discussing CNTs, it is referring to single-walled tubes unless otherwise specified.

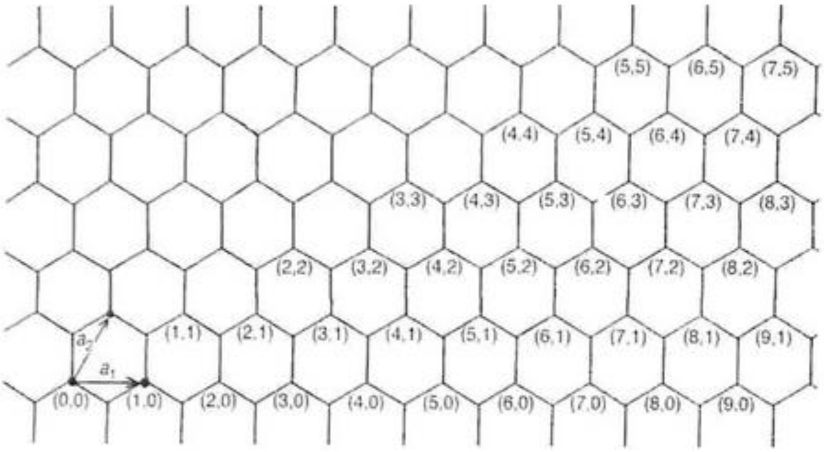


Figure 8: Chirality map overlapping a sheet of graphene. Copied from Harris [10].

CNTs and graphene share many common properties including incredible strength to weight ratio, thermal conductivity, and electrical conductivity. The similarities in material properties may come as no surprise; after all, CNTs are derived from graphene. As mentioned before, CNTs are essentially rolled-up graphene, of which there are several different configurations based on how it is rolled. A CNT can be defined as 1D unit cell, specified by a vector, (n,m), that determines how the graphene is rolled-up. Figure 8 depicts a sheet of graphene that has been overlapped by a lattice coordinate system called a chirality map. This map can be used as a tool to visualize the different structural configurations of a CNT. In order to create a CNT based on the indices (n,m), the sheet is rolled by superimposing (0,0) on the labeled hexagon (n,m). The values of (n,m) dictate the type and properties of the nanotube produced. [10].

There are three primary types of CNTs: “armchair”, “zig-zag”, and chiral. The armchair configuration in Figure 9(a), where n = m, and nanotubes where n - m equals a multiple of 3, produce nanotubes that have metallic electrical properties. While some zig-zag nanotubes (m = 0) and chiral nanotubes have semiconducting electrical properties (see Figure 9(b) and 9(c) respectively for structural illustration of these different types). [11].

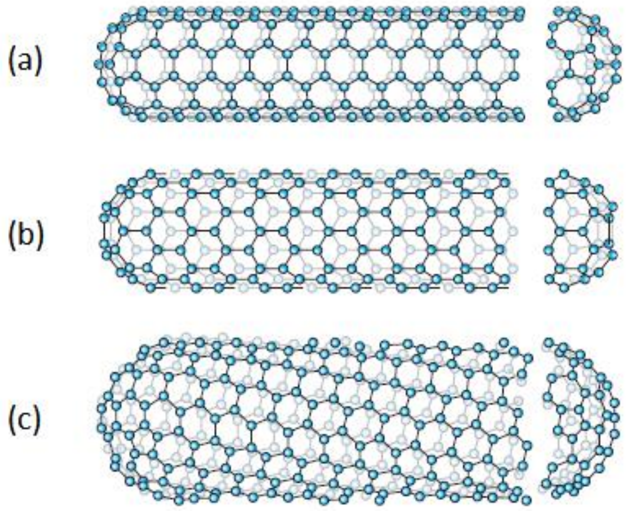


Figure 9: (a) Armchair nanotube. (b) Zig-zag nanotube. (c) Chiral nanotube. Modified from Askeland and Wright (2016) [5]

2.2.3 Applications

Like graphene, due to the lack of a large-scale production solution many of the applications for CNTs are hypothetical. However, in the laboratory environment, several research groups have been able to synthesize macroscopic materials completely built of

CNTs. The most common material being carbon nanotube fibers. These fibers have a similar appearance to thread but retain many of the properties of CNTs, such as increased strength, and thermal/electrical conductivity. There are several different methods for producing these fibers, most of which are still in development and need further optimization. A common technique is call dry spinning which can be roughly thought of as spinning wool into yarn. The source of nanotubes can come from a CNT forest, which metaphorically resembles a bamboo forest on a microscopic level. As nanotubes are pulled from the forest, they get tangled and stick together through van der Waals interactions. This phenomena causes the nanotubes to be aligned in the same direction with the axis of the fiber, and allows the fibers to be spun into a tread. [12,28-31].

In 2013, Rice University’s carbon nanostructure research group began producing these carbon nanofibers and testing their applications. One of their experiments included simultaneously suspending a lightbulb and supplying current to it with their synthesized nanofibers. Due to technological constrains, the experiment doesn’t encompass CNT’s full potential, yet it still demonstrates the multifunctionality of the material. [13].

2.3 Defects

The majority of the topics discussed in this document have described ideal structures and environments that rarely exist in the natural world. In reality most materials contain defects and impurities that can have a large impact on their properties. Depending on the impurity, defects can be very detrimental to the structures they are present in if they are not deliberately placed. However, defects can also have positive outcomes on the material, depending mainly on the producer’s needs. Pure iron, for example, is somewhat soft, but when trace amounts of carbon impurities are added to its structure the material’s strength is considerably increased [5,104]. There are three fundamental types of defects: line defects, surface defects, and point defects [5,103]. Because this research examines vacancies, which are categorized as a point defect, this section will not cover line or surface defects in any detail.

2.3.1 Point Defects

Point defects are defined as "… localized disruptions in otherwise perfect atomic or ionic arrangements in a crystal structure" [5,104]. While the name specifies a "point" in a structure, the influence of these defects affect many of the surrounding atoms in the lattice. These types of imperfections occur when an atom's movement increases, often due to the addition of energy to the system, or during material processing. There are several different types of point defects, each of which affect the surrounding region differently. This project will briefly discuss interstitial defects, substitutional defects, and most importantly vacancies.

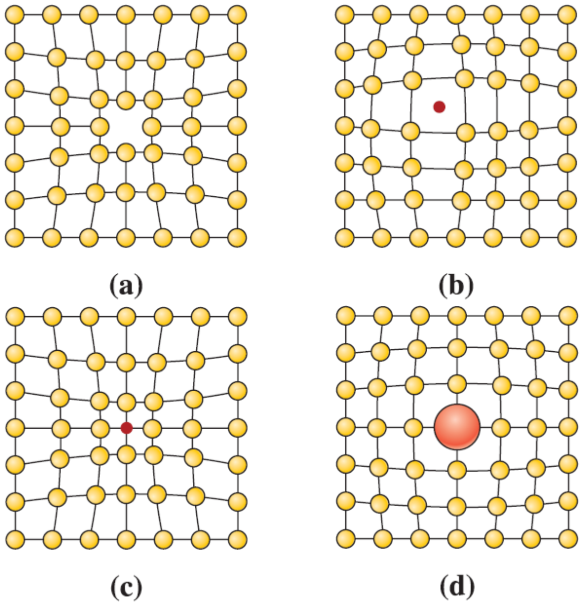


Figure 10: (a) Vacancy. (b) Interstitial atom. (c) Small substitutional atom. (d) Large substitutional atom. Modified from Askeland and Wright (2016) [5]

Figure 10(b) shows an interstitial atom in a cubic crystal lattice. Even though the crystal lattices discussed thus far have been hexagonal and the atoms have been C, the principle remains applicable despite details such as these. An interstitial defect occurs when an atom occupies a normally empty space in the lattice structure. As shown in Figure 10(b), the interstitial atom's presence causes the surrounding atoms (and even atoms near to those) to be pushed and compressed to "make room" for the intruding atom. Interstitial atoms can either be the same or different in comparison to the local atoms. The larger the atom is, the greater its affect will be. However, the area neighboring an interstitial defect will be distorted regardless of the atom's size.

Figures 10(c) and 10(d) show substitutional defects. Like the name implies, this defect occurs when an atom of a different type replaces a local atom at its lattice site. Like interstitial defects, regardless of the substitutional atom's size it will still distort the neighboring crystal region. When a substitutional atom is larger than the local atoms, it reduces the interatomic spacing of surrounding atoms as depicted in Figure 10(d). On the other hand, when the substitutional atom is smaller than the local atoms, the interatomic spacing of nearby atoms increases, pulling the neighbors towards the site like in Figure 10(c).

2.3.2 Vacancy

The vacancy is the last point defect defined and is the most important to this project. In the simulations discussed in the follow chapters, two carbon nanostructures containing vacancies will be heated up and analyzed. For this reason, it is essential to understand what vacancies are, and how they can affect the structures that contain them.

A vacancy is created when there is a hole in a normally occupied lattice site, as shown in Figure 10(a). When a vacancy occurs, it increases the randomness and therefore the thermodynamic stability of the material [5,105]. In Figure 10(a), you can see how the vacant atomic site makes a strain on the structure's lattice, causing the neighboring atoms to push inward to somewhat compensate for the hole. As a material is heated up (or energy is applied to the system) the amount of vacancy defects increases exponentially. As the temperature of the system approaches the material's melting point, there can be up to a vacancy per 1000 atoms. [5,105]. Intuitively, the more vacancies present in a material, the lower its density. One might ask: “Where do the atoms go after they leave their lattice site?” The atoms do not just disappear, in fact, the atoms tend to move around in the structure, resulting in an increase of the material's surface area and volume [14]. Typically, as the temperature of the system decreases the amount of vacancies are also reduced. However, it is possible to trap a high concentration of vacancies by rapidly cooling down the system using a method called quenching [5,105]. This technique roughly maintains the lower density, higher volume properties of the previously heated material by not giving the atoms time to move back into the vacancies.

2.3.2.1 Diffusion

Diffusion is the movement of atoms within a system that contributes to the reduction of concentration differences. As temperature or energy is applied to a material, the atoms vibrate more rapidly within their bonds, increasing the chance of an atom breaking free and moving to a new position, or diffusing within the structure. Diffusion in a system depends both on temperature and the concentrations and is therefore closely related to vacancies. When a vacancy is present in a lattice, the surrounding atoms can “jump” from their lattice site into the unoccupied site, causing the vacancy to move. [5, 141-142].

The jumping of local atoms into an adjacent vacancy is called self-diffusion and constitutes the local mitigation of a difference in concentrations. Once one atom jumps into a vacancy, the vacancy moves to a new position where the new adjacent atoms now have the chance to jump as well. This repetition of jumping creates a counterflow between the vacancy and the diffusing atoms so that they move in opposite directions. The movement of the vacancy in self-diffusion is called vacancy diffusion. A pictorial representation of self-diffusion and vacancy diffusion can be seen below in Figure 11. [5,147-148].

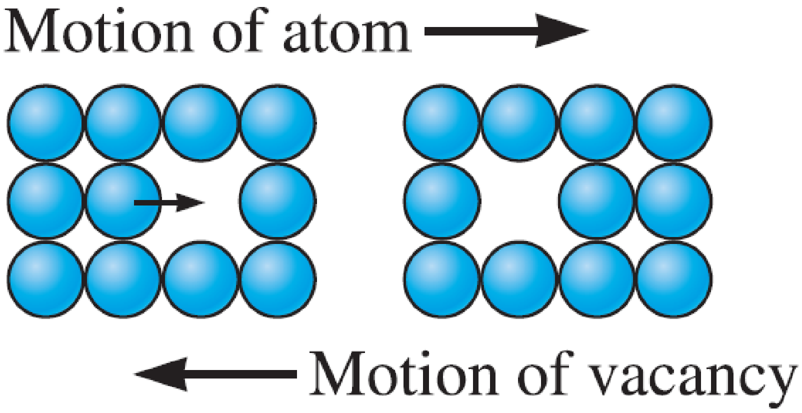


Figure 11: Self-diffusion and vacancy diffusion. Copied from Askeland and Wright (2016) [5]

3 Methods and Materials

This chapter contains an overview of the methods and materials used during the project to produce the results examined in the next chapter. As described in the Introduction, the goal is to analyze the presence of a vacancy in different carbon nanostructures through a purely digital environment. First there is a brief introduction to simulations and the overall workflow used in the research. After that, the main simulation software tools, ASE and LAMMPS, will be introduced and their core functionalities within the project described. Lastly, this chapter covers the post-processing of the simulation results and how they were visualized for analysis.

3.1 Simulations

Before discussing the process used to conduct the simulations, it is helpful to have a base definition of how the term “simulation” is used in this report. A simulation is a computer program that executes a series of methods that model a system. The procedure begins by feeding a specification of the system’s base state to the simulation program’s algorithms. The base state will exist at a starting time t. The algorithms then calculate the state of the system at the next timestep, t+1, whose results are used in the following calculation for the timestep t+2. This continues until the specified amount of time steps have been reached. Each state, from the first to the last, is data that represents a numerical image of the system’s progression through the simulation. It is up to the researcher conducting the simulation to decide how often and when the generated data should be saved. [15].

The two structures that have been selected for the simulations are the carbon nanotube (CNT) and the graphene nanoribbon (GNR). These structures have been selected based on their interesting atomic arrangement, size, and properties. Because these structures exist on such a small scale, it can be difficult to run repeatable experiments in the laboratory environment, making simulations a powerful tool in analyzing these materials and their characteristics. Another advantage of simulations is the ability to have nearly complete control of the environment variables and the structure used. This allows researchers to look at “perfect” structures that might not necessarily exist in nature.

3.1.1 Workflow

Figure 12 depicts the general workflow of conducting a simulation in this project. It begins with the formulation of what the researcher wants to model or test. This portion of the process is the Preparation phase. During this phase the researcher determines the general factors of the simulation. These factors can include: the purpose of the simulation, determining the structures used in the simulation, what will the simulation actually do, and what are the expected results of the simulation. Even though the Preparation phase is not exactly tangible, it still acts as a critical portion of the process. In fact, from the perspective of this project, the more attention spent on preparation, the more meaningful the simulation.

Once the simulation's blueprint has been established, the researcher can move into the digital environment to create the planned structure. The software used for structure generation is a python module called ASE. ASE provides tools to create, manipulate, and export custom atomic structures. Once the structure has been created within ASE, the data is exported into a formatted file that contains information such as the elements in atomic arrangement, positions of atoms, and the size of the structure.

Unfortunately, the file format exported by ASE is not directly compatible with the input format of the simulation software and therefore must be converted. This can either be done by hand or parsed using a BASH script. The properly formatted data file is then passed to the Simulation phase.

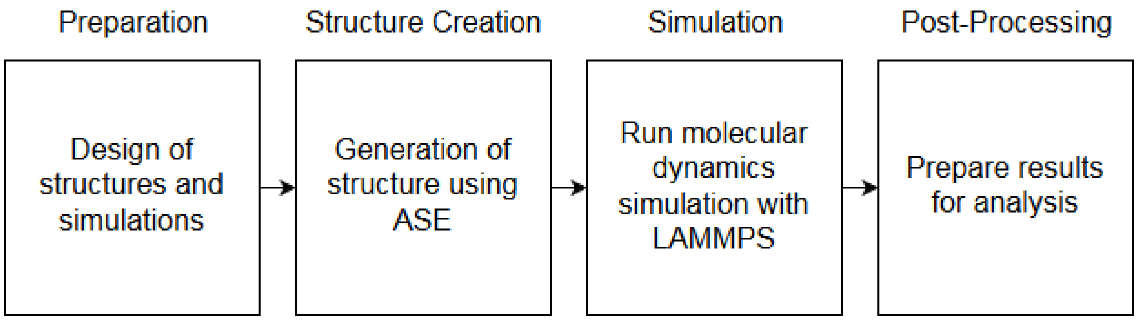


Figure 12: The general workflow of conducting a simulation.

As the name suggests, the Simulation phase is where the actual simulation takes place. LAMMPS is a molecular dynamics computer program and is the core software executing the simulation. Based on the Preparation phase's simulation design, the researcher constructs a command script, which is composed of induvial LAMMPS commands. When executing a command script, the formatted input file is imported and parsed, creating the base state of the system. The simulation is then ran timestep-by-timestep, recording the data, as specified, to an output file.

The LAMMPS output file alone renders little insight to what happened to the structure during and after the simulation. This requires the resulting numerical data to be passed to the Post-Processing phase in order to change it into a form that is easier to analyze. The means of post-processing is completely based on the researcher’s needs and is selected in order to make the numerical results most useful. In some cases, multiple tools or techniques can be used to extract information from different aspects of the data. The post-processing medium chosen for this project is a visualization tool called Ovito. Ovito excels at generating animations and frames of atomic simulation data. This is very useful when looking at the overall change in a structure.

3.2 Atomic Simulation Environment (ASE)

Atomic Simulation Environment (ASE) is an open-source python module that provides tools to set up, execute, and analyze atomic simulations [16]. It is one of the two primary software packages used to conduct atomic simulations. Even though ASE has a plethora of functionalities, and could even be used for all aspects of a simulation, its core purpose in this project is to create and prepare atomic structures for later usage.

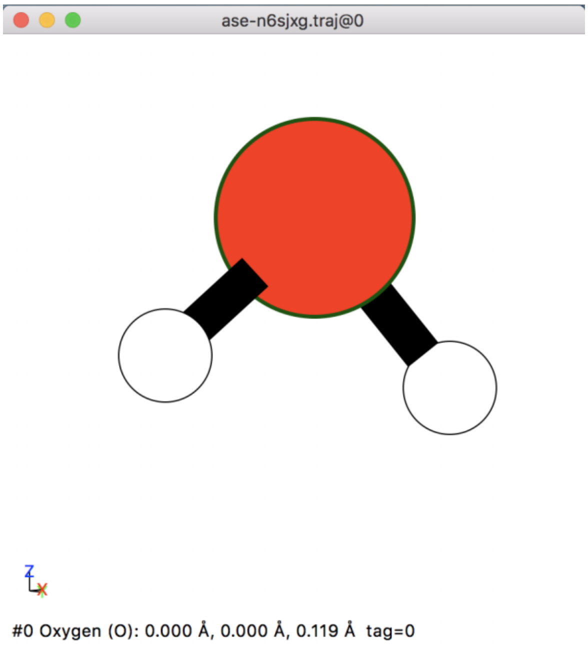
ASE was selected for the task of structure generation due to its familiar command line interface and its ability to visualize the atomic arrangements. Installation of the ASE package is relatively straightforward and only requires an updated version of Python and NumPy for the majority of its basic functionalities. When ASE is installed, the user opens the python interpreter and simply imports the ASE tools related to their needs. Alternatively, if the user already knows how they want to build a structure, they can also create a python script and run it without the interpreter interface.

3.2.1 Tools

When creating an atomic structure in ASE the *Atoms* class is the main interface. An Atoms object essentially contains an array of individual atoms that make-up the structure, as well as methods to retrieve and set data related to them. When creating an Atoms object, the user can set data members such as: the type of elements within the structure, the amount of atoms present, their positions, and boundary conditions to name a few.

Another way to generate structures is with ASE's *build* tool, which contains an assortment of functions for creating common crystal lattices. The *build* functions are a more abstract means to create structures in comparison to the *Atoms* class, where the user must define each atom. If a *build* function exists for the user's desired atomic arrangement, it can be much easier to create larger structures containing many atoms. Because of the higher abstraction level, functions like these take several structure specific parameters, like chemical symbol or lattice constant, and leave the rest of the generation, like atom positions, up to the function. Build functions return an *Atoms* object so that the user can then manipulate, view and customize the structure using the class' methods.

ASE also offers a visualization tool that provides a means to access its graphical user interface (GUI). The *view* tool is needed if the user wants to examine the structure from a graphical perspective. The function takes an Atoms object as its main parameter and by default, opens the system in the GUI. The GUI not only provides an overall picture of the system but can be used to rotate and zoom the image, and even edit the structure itself. The *view* function is also a very helpful verification method because it allows the user to visually cross check if the atomic structure is as intended. Figure 13 shows an example of a simple water molecule (H_2O) being visualized in ASE's GUI.



#0 Oxygen (O): 0.000 Å, 0.000 Å, 0.119 Å tag=0

Figure 13: A water molecule visualized in ASE's GUI.

The ASE *io* (Input/Output) module is used for reading and writing atomic structure files. The functions *read* and *write* were used extensively from this module and are critical if the user needs to work outside of ASE's environment. As the function names imply, *read* is for importing external data files into the current environment, and *write* is used to export structures into data files for external use. When reading and writing data, the handled file must be in an ASE compliable format. There is an extensive list of possible read/write formats, but unfortunately none of them are directly compatible with the LAMMPS input parser. However, the .XYZ file format provides all the required data as well as a somewhat similar file layout as a LAMMPS input file. For this reason, it was easier to convert an ASE output file to a LAMMPS input file and the .XYZ format was selected as the *io* format used with ASE.

3.2.2 Structure Creation and Preparation

In the Preparation phase of the project workflow the researcher defines the structure, and its properties, used in the future simulation. By using ASE creation tools, the implementation of the designed structure can begin. As mentioned before, the two atomic structures utilized are the carbon nanotube (CNT) and graphene.

When creating a sheet of graphene, or a graphene nanoribbon (GNR), there are several base parameters that must be selected beforehand that determine the GNR's physical properties. First, the researcher must decide the lattice orientation of the GNR. As described in previous sections, “zigzag” and “armchair” orientations are the two available options, each one providing different material characteristics. In ASE it is possible to explicitly define the orientation and leave the positioning of atoms to the software. The user then determines the size of the nanoribbon, which is defined by two integer multipliers (one for each dimension of the structure) that repeats the base arrangement in each direction. An example of a four-by-four “armchair” GNR can be viewed in Figure 14(a). Once the lattice has been generated, if further preparation is required, the user can then modify structure. In Figure 14(b), a four-by-five “zigzag” GNR has been modified to contain a vacancy, which is the defect monitored in this project.

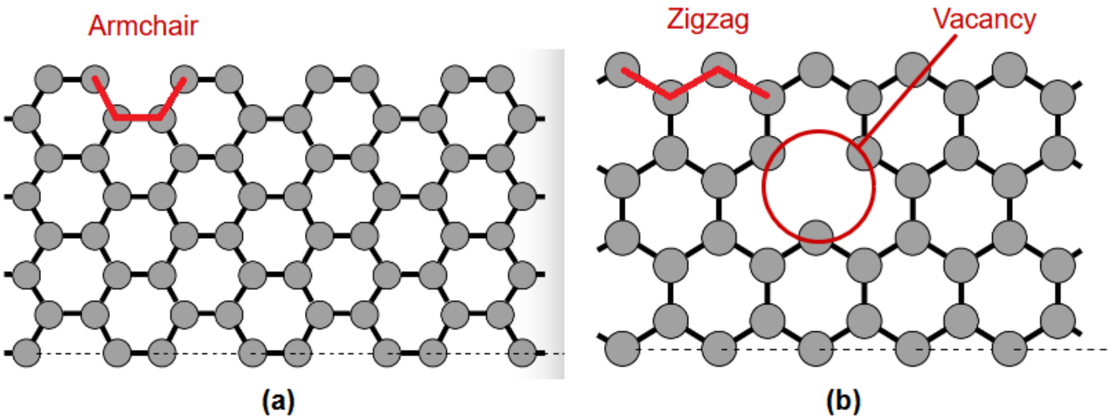


Figure 14: (a) Four-by-four “armchair” graphene nanoribbon. (b) Four-by-five “zigzag” graphene nanoribbon containing vacancy.

Unlike the GNR, when building a CNT, the structure’s lattice orientation is based on two integers that represent the *n* and *m* values when rolling a sheet of graphene (reference section 2.2.2.1 for an overview of the CNT structure). The combination of n and m determines whether the CNT has a “zigzag”, “armchair”, or chiral lattice orientation. The size of the tube is not only based on the (n,m) values but also on a *length* parameter that establishes how many times that base structure is repeated, similar to the length parameter GNR generation. In Figure 15 three different types of CNT are displayed. Figure 15(a) displays a chiral (6,3) CNT, with the length of 4, containing 336 atoms. Figure 15(b) displays a “zigzag” (6,0) CNT, with the length of 4, containing 96 atoms. Figure 15(c) displays a “armchair” (6,6) CNT with the length of 4; this structure typically contains 96 atoms but due to the vacancy defect depicted, it only contains 95.

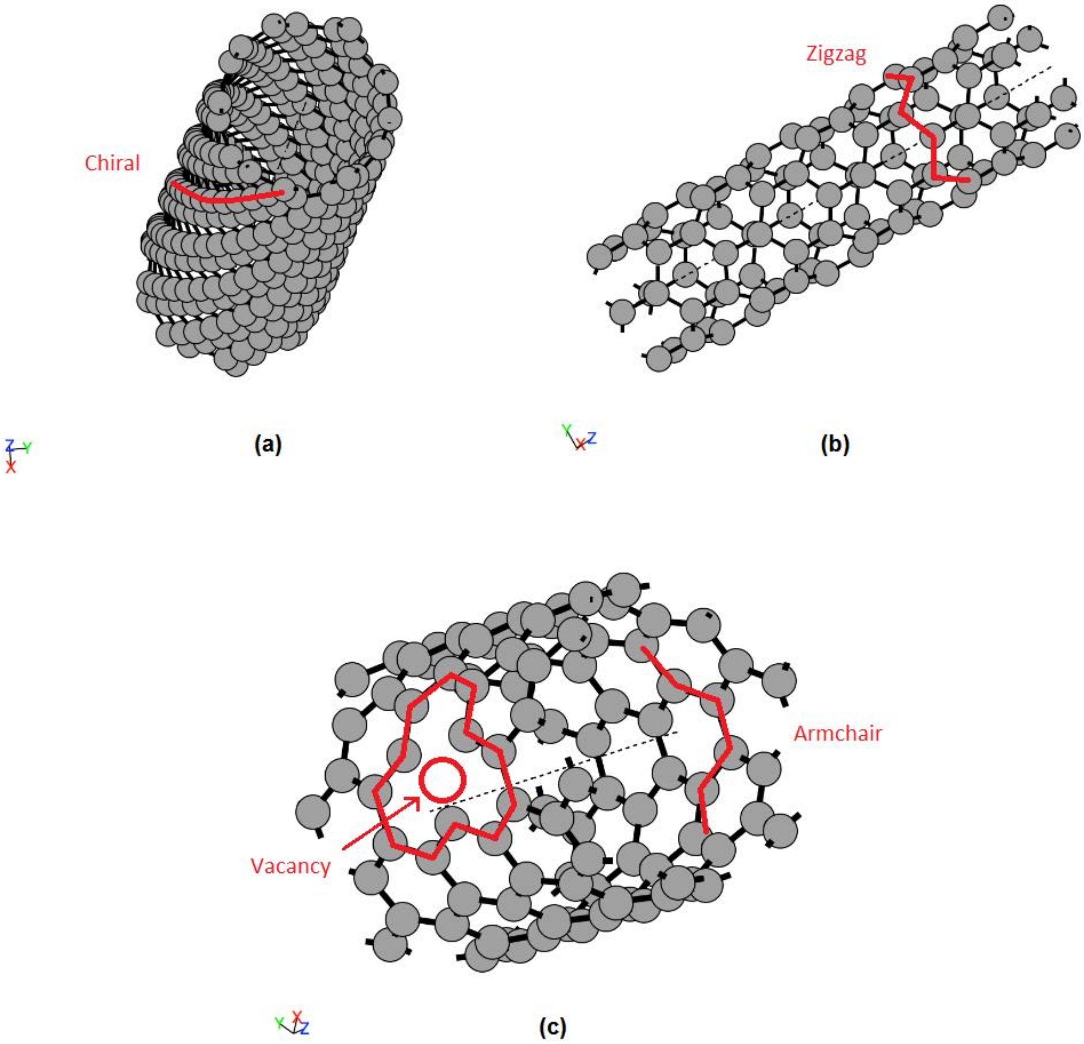


Figure 15: (a) A chiral (6,3) CNT with the length of 4. (b) A “zigzag” (6,0) CNT with the length of 4. (c) A “armchair” (6,6) CNT with length of 4 containing a vacancy.

3.3 Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)

LAMMPS is a molecular dynamics software and is the principle mechanism for running simulations in this project. The software is up-to-date and maintained by developers from Temple University and Sandia National Labs (a United States Department of Energy laboratory) and has been used in countless research groups since it was contrived in the 1990s. LAMMPS can model a variety of metallic, biological, and atomic systems ranging from several atoms to millions. While it can be ran on a laptop, LAMMPS excels when applied to powerful machines that utilize parallel computing. Theoretically, the size and complexity of a simulation can be boundless, only being limited by the resolution of the simulation and the computing power of the host machine. From a software perspective this makes LAMMPS very scalable and allows it to be applied to nearly all computers given they fulfill the needed resources. [17]. Luckily, because the simulations conducted in this project are relatively small-scale there was no need for ample computing resources. This allowed all aspects of the simulation process to be executed on a single personal computer.

3.3.1 Input

In the ASE section, it explains how a structure is created and exported to a file for future use. Because the ASE output file is incompatible with LAMMPS input, the .XYZ data file must be converted into a format that LAMMPS can understand. LAMMPS reads external data files using the *read_data* command which is contained in the simulation command script (discussed in a later section). *read_data* receives a specified file and parses through it, retrieving the data required to create the atoms used in the simulation.

Even though certain settings and sections are optional and independent of a specific order, the format of that data file is very important. There are two components of the file: the header and the body. The header must come before the body and often contains information that needs to be established before the body is read. The header is composed of values and keywords that are read line-by-line, where the keywords are always preceded by the values. There are three basic keyword phrases that most data file headers contain: *atoms*, *atom types*, and *lo/hi* boundaries. The *atoms* key word simply states the number of atoms in the structure and *atom types* phrase tells the software how many different types of atoms there are.

The *lo/hi* boundaries define the size of the simulation box, which can be thought of as an area in space where the experiment (simulation) takes place. The simulation box must be able to contain the structure used in the simulation, meaning that the positons of all the present atoms begin within its bounds. Depending on the dimensionality of the atomic structure, there can be up to three *lo/hi* bounds defined in the header (simulation cannot exceed 3D). Each bound corresponds to the possible position values (coordinates) of an atom in that dimension. In Figure 16 you can see a conceptual representation of atoms, depicted by the blue orbs, within a simulation box.

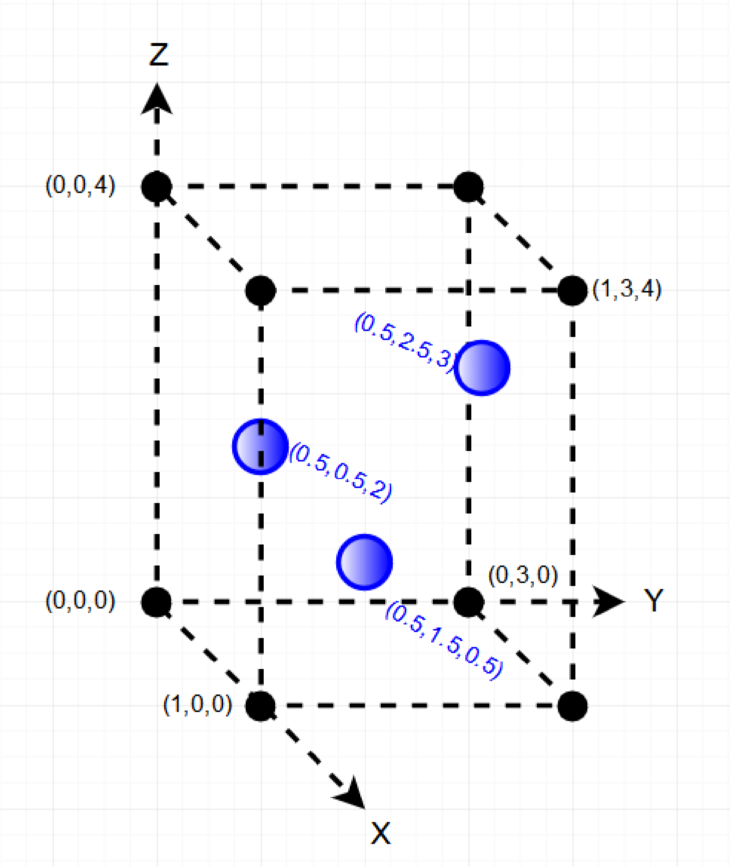


Figure 16: Conceptual representation of a simulation box containing three arbitrary atoms (blue orbs).

LAMMPS would interpret the box above by reading the following values and key phrases: for the X dimension: *0.0 1.0 xlo xhi*, for the Y dimension: *0.0 3.0 ylo yhi*, and for the Z dimension: *0.0 4.0 zlo zhi*. Figure 16 also shows how all the atoms participating in the simulation must be within the boundaries.

The body of a data file is composed of sections. Like the header, there are certain key words that tell LAMMPS *read_data* parser what the data is. Each section of the body begins with a keyword and is followed by the data pertaining to that section. The simulations ran in this project required at least two sections in the data file body: *Masses*, and *Atoms*.

The *Masses* section contains the atomic masses of the present atom types, as well as the mass ID. For example, if C is the only type of atom in the structure, its mass (in mass units) would be *12.01* and its type ID *1*. The *Atoms* sections contains the individual atom's data. Each atom has its own line inside the section, so if there are 100 atoms in the structure, then there will be 100 lines consisting of each atom's information. Typically a line contains the atom's position (*x*, *y*, *z* coordinates), type ID, and its unique ID.

3.3.2 Commands

A LAMMPS simulation is composed of a series of commands often contained within a script file. The command script is executed line-by-line until the parser reaches the end of the file. The script is usually broken into four sections: Initialization, Atom Definition, Settings, and Run Simulation.

The Initialization section contains parameters that should be set before atoms are defined. One of the first commands declared, is the *units* command which sets the different units used during the simulation. There are several different styles of units, each one specialized for different simulation scenarios. The *metal* unit style suited this project the most, with the mass in grams/mole, the distance in Angstroms, and the time in picoseconds. These were ideal for the size and scale of the simulations.

The *dimension* command is simple but important; it defines the dimensionality of the simulation box. The *boundary* command sets the simulation box's boundary types. Depending on the number of dimensions set by *dimension*, each axis needs its type of boundary defined. By default, in a 3D simulation box, each dimension is of the *periodic* type. This means if an atom passes through a boundary wall, it re-enters through the opposing side. If a boundary is not periodic, then when the atom moves through the corresponding simulation box wall, it is deleted on the next time-step.

Another important Initialization command is *pair_style*. The command defines the equations used to compute the interactions between atoms. The *tersoff* pair style is often used in applications containing silicon and carbon. Since the systems used in this project are carbon based, the *tersoff* pair style was a suitable choice for the atomic interactions. However, the *tersoff pair_style* alone does not provide LAMMPS all the required information; an additional Tersoff potential file, containing coefficients used in the calculations must be imported to the command script. The Tersoff potential file used in this project's simulations can be found in appendix 1 in case similar simulations want to be executed.

Following the command script's Initialization section, Atom Definition goes on to define the atoms and the molecular topology of the structure. The *read_data* command discussed in the previous section is usually the primary component of the Atom Definition. Since *read_data* imports the file containing all the atom definitions, there is no need to further define atoms in the command script itself. However, if there is no data file to import, then the Atom Definition would contain the commands needed to produce the atoms/molecules involved in the simulation.

The third section of the command script is the Settings, which contains commands that determine what happens during each simulation timestep. Additionally, Settings can also provide commands that augment ones from previous sections. For example, the *pair_coeff* command reads the Tersoff potential file required for the previously mentioned *pair_style* command. And the *timestep* command sets the duration time between each simulation step, with units defined by the *units* command.

The *fix* command is very flexible and can be used for a variety of functions. A *fix* can be any process applied to the simulation box during timestepping. For example, *fix* was used in this project's simulations to warm-up and cool-down the CNT and graphene structures. Each timestep the temperature would change based on a given interval and the effect on the structures would be recorded.

Another important command that relies on timestepping is the *dump* command, which is the primary means for outputting data for post-processing and analysis. *dump* takes a snapshot of the system in given intervals and records the data that has been specified by the user. By adding additional arguments, the command can be customized to output particular data such as positions and velocities of the atoms.

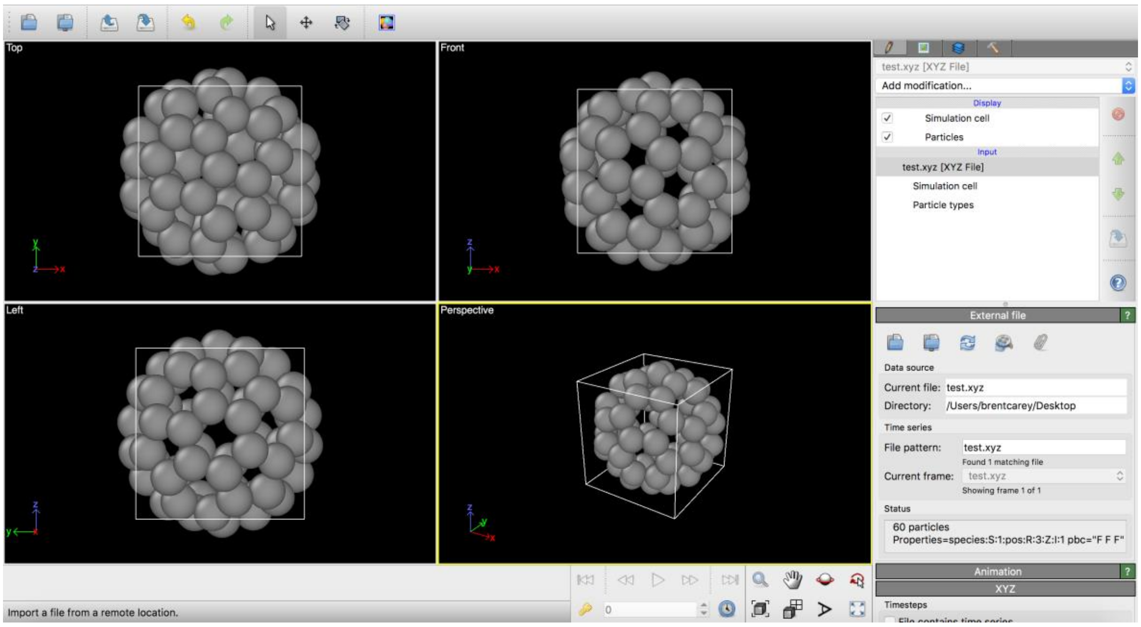
The last section of a command script is the Run Simulation. There are only a few commands that go into this section, the main one being *run*. This command begins the molecular dynamics simulation as defined by all the previous commands. The primary parameter accompanying *run* is the number of timesteps the simulation takes. This command marks the beginning as well as the end of the simulation.

3.4 Post-Processing

The final phase of conducting a simulation is called Post-Processing. LAMMPS itself does not specialize in processing techniques and offers few mechanisms for users to envision their results. Therefore, it is often necessary to employ third party software to extract the greatest insights from the data. The LAMMPS dump format can be read by many different programs, so it is up to the user to select an analysis software that caters to their specific needs.

It is often useful to generate images depicting the atoms at different timesteps. By using such techniques the researcher can visually evaluate how the structure has changed over the duration of the simulation. The Build Structure phase uses ASE to visually verify that the produced arrangement is suitable before simulation. ASE’s GUI could also be applied to post-processing by importing a timestep snapshot from a LAMMPS dump file. However, if a dump file needs to be used by ASE’s GUI, it is difficult to import more than one snapshot at a time, making it tedious to view consecutive timesteps. For this reason, Ovito was employed as an additional software tool, specifically for viewing LAMMPS dump data.

Ovito is a scientific analysis application specialized for the visualization of molecular dynamics simulation data. It is a free, open source software available to all primary operating systems. Ovito contains a wide variety of analysis tools that can be used by researchers to extract relevant information from their simulation. In addition to being directly compatible with LAMMPS data dumps, it can also handle most common formats including XYZ files (generated from ASE). One of the features that was most valuable to this project, was the ability to import and view an entire simulation. Instead of viewing a single timestep snapshot, it is possible to string the separate fames together creating somewhat of a movie. This made it much easier to see the how a vacancy effects the surrounding atoms in the carbon nanostructures. [18].



☑ Simulation cell
☑ Particles

Current file: test.xyz
Directory: /Users/brentcarey/Desktop
File pattern: test.xyz
Found 1 matching file
Current frame: test.xyz
Showing frame 1 of 1

60 particles
Properties=species:S:1:pos:R:3:Z:I:1 pbc="F F F"

Figure 17: Rendering of a C60 Buckminsterfullerene in Ovito’s GUI.

The visualization above in Figure 17 shows an example of the rendering of a C60 Buckminsterfullerene in Ovito. Unlike ASE and LAMMPS, which are primarily terminal interfaced software, Ovitio is mainly a GUI application. Most of its functionalities are accessible through the graphical environment using a mouse rather than a keyboard. Making Ovito ideal for researchers lacking a background in the command line.

4 Results

This chapter covers the results and observations made from two selected simulations conducted during this research. While many simulations were executed during the project, the ones discussed in this chapter are a culmination of the previous trials. Each structure and its adjoined simulation exemplifying natural phenomena that can occur in the given scenarios.

4.1 Carbon Nanotube

The first simulation presented contains a carbon nanotube in the presence of a vacancy defect. Tables 1 and 2 display the basic parameters needed to recreate the simulation for personal use. Table 1 shows the parameters and values used when designing and generating the structure. The CNT's type is based on the *n* and *m* values. Because the value of *m* is equal to *0*, the nanotube is classified as having a "zigzag" lattice orientation. A combination of the *n*, *m*, and *length* values creates a nanotube composed of 96 atoms. However, when a vacancy is applied to the structure the number of atoms is reduced by 1, resulting in the 95 value in the Table 1.

Parameters	**Values**
n	6
m	0
length	4
type	zigzag
atoms	95

Table 1: Carbon nanotube creation parameters and values.

Table 2 contains the simulation related parameters and values. The *lo/hi* parameters are based on the size of the nanotube's structure. At the start of the simulation, the approximate structure of the (6,0) CNT is cylindrical, with the radius of 4.937 Å. By using this radius, the X and Y dimensions of the simulation box can be determined. The Z dimension is based on the height of the structure which can be derived from the *length* value.

Parameters	Values
units	metal
thermo	10
pair_style	tersoff
timestep	0.001
temperature	0 - 3500
run	10000
xlo/xhi	-4.937/4.937
ylo/yhi	-4.937/4.937
zlo/zhi	0/17.04
mass (C)	12.01

Table 2: Carbon nanotube simulation parameters and values.

The simulation consists of applying energy to the system and watching how the vacancy behaves. The energy added is in the form of heat, where the temperature is first increased, then held at the target value. The system begins at 0 Kelvin and is raised to the target value of 3500 Kelvin.

4.1.1 Analysis

When the simulation begins, all the atoms within the structure are at rest. As the system heats up, the atoms within the CNT begin to move in their bonds, giving them a velocity. As the temperature rises, the atoms vibrate more violently, deforming the nanotube’s structure. Because the atoms surrounding the vacancy are only held by two covalent bonds, they are able to move more freely in comparison to the interior atoms held in place by three bonds.

Figure 18 shows the temperature of the system over the duration of the simulation in a temperature vs time graph. The simulation lasts for 10 picoseconds, with the initial heating taking place over the first picosecond. Once the simulation reaches the target value of 3500 Kelvin, the system is held at the corresponding temperature range for the rest of the simulation.

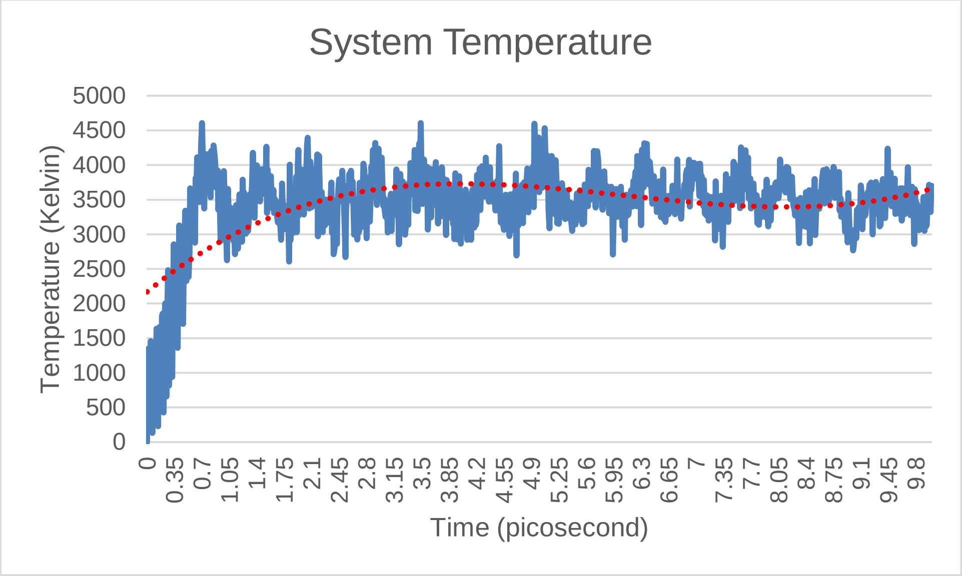


Figure 18: Graph of the temperature vs time during CNT simulation.

During the experiment, the natural phenomena of diffusion occurred three times. Figure 19 shows the first and last frames of the simulation, depicting the change of position for atoms 41, 42, and 45. It is also apparent from the figure below how the vacancy has moved over the course of the simulation into a new lattice site.

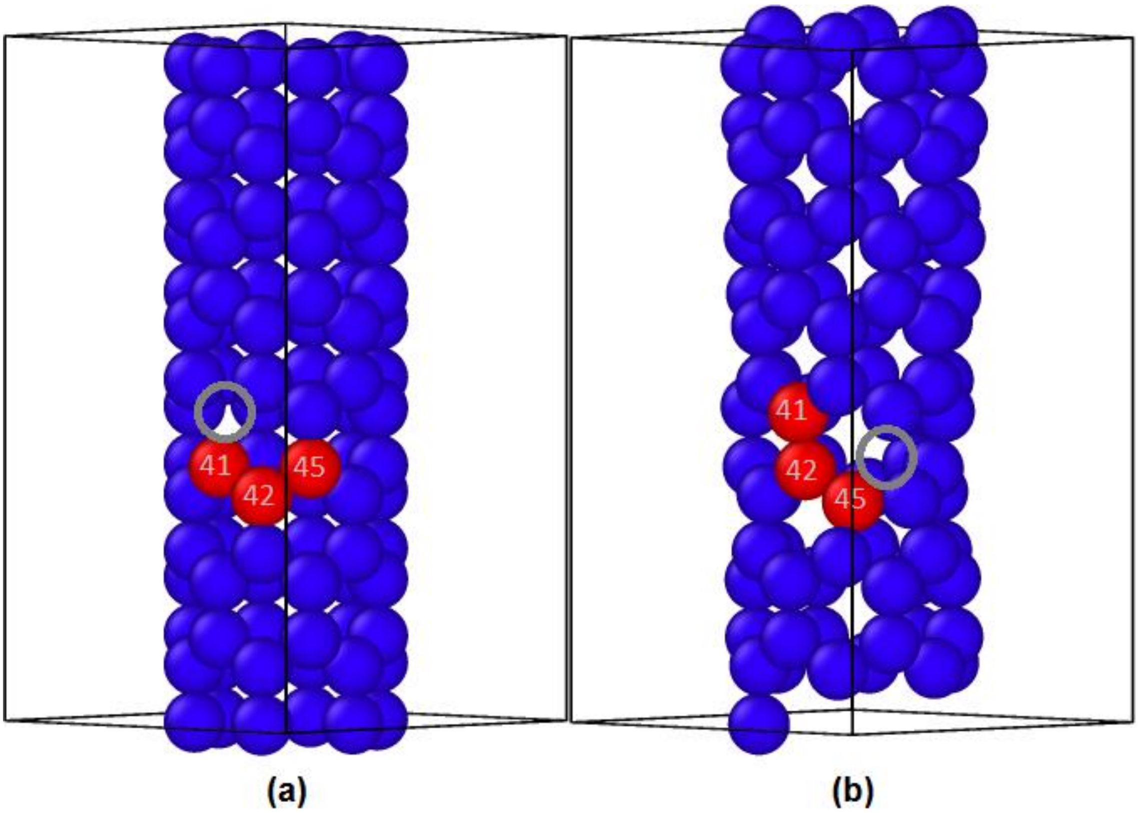


Figure 19: (a) First frame of CNT where the red atoms are in original position. (b) Last frame of CNT where red atoms are in new position after diffusion.

The first diffusion begins 4.26 psec into the simulation and takes place over about 0.04 psec, being in its new lattice site at 4.30 psec. The temperature during this period is near a local maximum in the System Temperature graph (in Figure 18) and may have been the energy fluctuation needed to trigger the diffusion. Figure 20 shows the three simulation frames where the atom diffuses to a new site. In Figure 20(a) at time 4.26, atom 41 has a velocity of 77.0754 Å/psec in the Y direction. At 4.28 psec, atom 41 has broken out of its bonds and is diffusing to a new lattice site as seen in Figure 20(b). Figure 20(c) shows the structure after the diffusion has taken place, where atom 41 and the vacancy have exchanged sites.

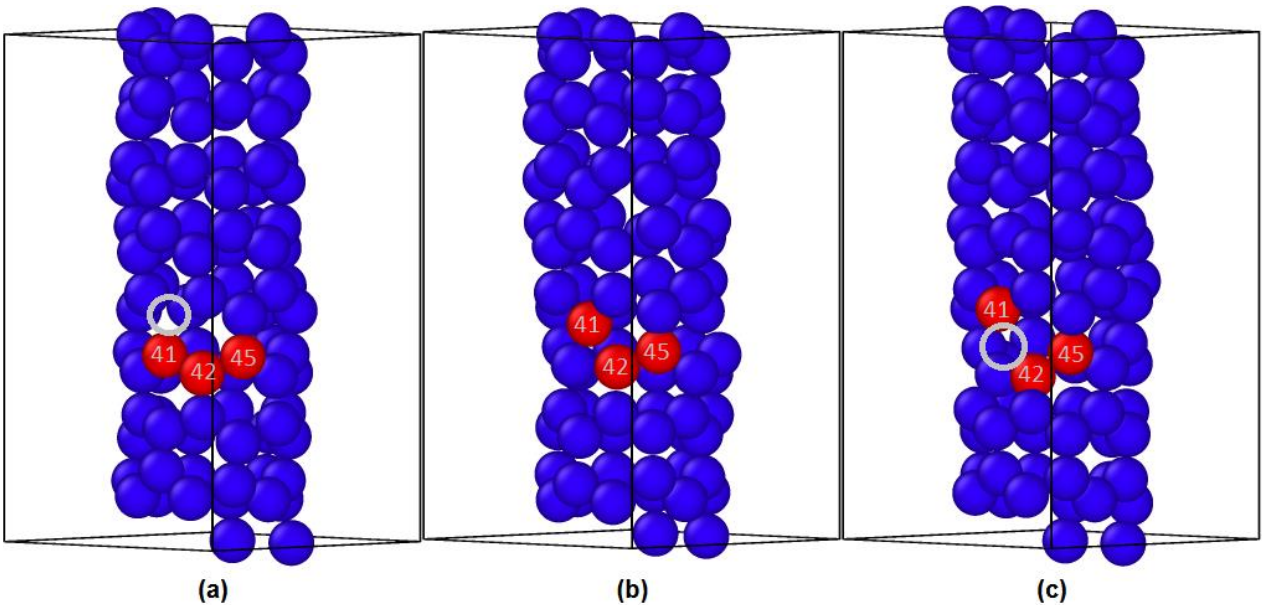


Figure 20: (a) Structure at 4.26 psec and 3730.4801 K before diffusion. (b) Structure at 4.28 psec and 3771.2615 K with atom 41 diffusing. (c) Structure at 4.30 psec and 3765.1789 K after diffusion.

The second diffusion starts at 5.68 psec and lasts for approximately 0.06 psec. As depicted in Figure 21, atom 42 moves from its original site to take the place of the vacancy. Unlike atom 41's diffusion, there was no exceptionally high velocity in a specific direction. The highest velocities recorded being at time 5.72, having values of -24.312 Å/psec in the X direction, -20.1106 Å/psec in the Y direction, and 40.493 Å/psec in the

Z direction. However, this could be due to the lack of resolution in the simulation, resulting in missing values in the recorded data. Similar to the diffusion of atom 41, atom 42 diffuses at a local maximum in temperature. The frame in Figure 21(b) depicts the structure at the local peak temperature of 4203.6568 K.

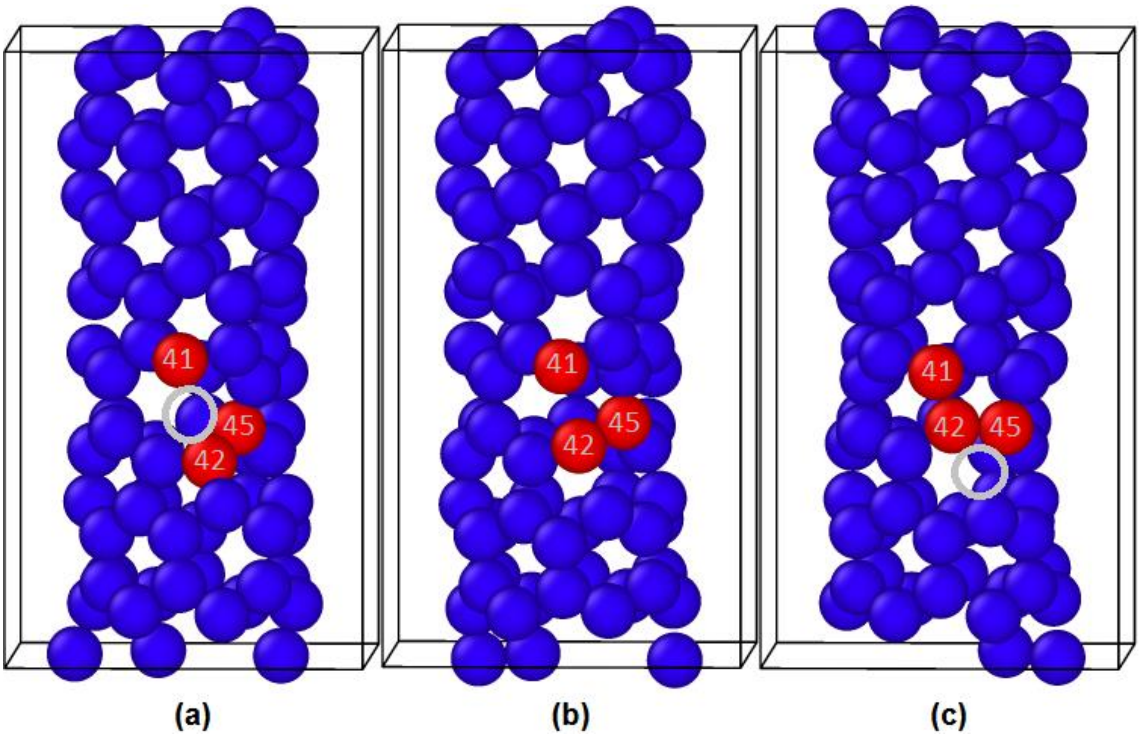


Figure 21: (a) Structure at 5.68 psec and 3543.3575 K before diffusion. (b) Structure at 5.70 psec and 4203.6568 K with atom 42 diffusing. (c) Structure at 5.74 psec and 4199.1692 K after diffusion.

The last diffusion took place from time 8.54 to 8.58, taking 0.04 psec for the change in lattice site. Figure 22 shows the movement of atom 45 from its original site to the vacancy site. While the temperature is again at a local maximum the corresponding values are lower than in the previous two diffusions, with the highest being 3605.6517 K at 8.55 psec. Atom 45 shows a relatively large velocity in the X direction with a value of -46.2777 Å/psec at time 8.56. This may correspond with the energy needed to break the previous bonds in the frame displayed in Figure 22(a).

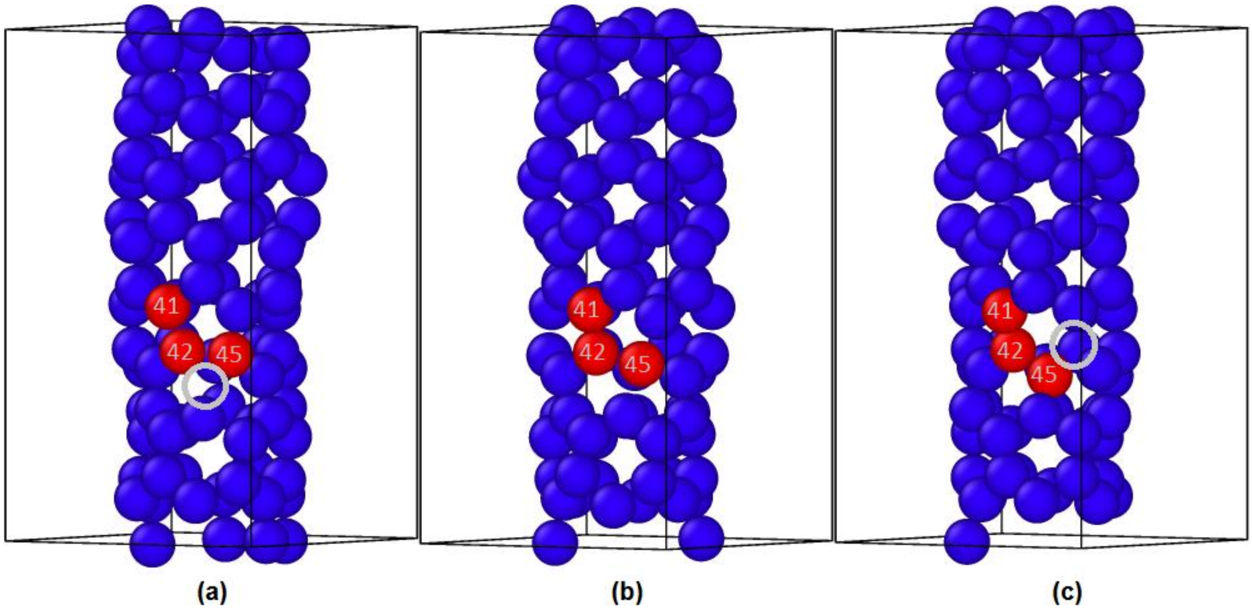


Figure 22: (a) Structure at 8.54 psec and 3551.3555 K before diffusion. (b) Structure at 8.56 psec and 3490.8268 K with atom 45 diffusing. (c) Structure at 8.58 psec and 3554.0276 K after diffusion.

4.2 Graphene Sheet

The second simulation shows the results of heat being applied to a system containing a graphene nanoribbon. Like the previous simulation the structure contains a vacancy and energy, in the form of heat is applied to the system. The GNR has an armchair lattice orientation and is built by repeating that base structure using a length value and a width value. Table 3 shows the parameters and values need to replicate the GNR used in this simulation. With the *length* and *width* of *6* and the type being armchair the nanoribbon generated contains 144 atoms. When an interior atom is removed to create a vacancy, the structure is left with 143 atoms.

Parameters	Values
length	6
width	6
type	armchair
atoms	143

Table 3: Parameters and values to build GNR.

The parameters and values needed to replicate the graphene nanoribbon simulation are found in Table 4. Unlike the nanotube, a nanoribbon is a 2D structure. The table below shows the dimensionality of the simulation as well as the simulation box parameters. Because the simulation box is 2D, only two boundary conditions are required for the simulation. Due to the orientation of the structure when built, the atoms can be defined using the X and Z axes. Therefore, in Table 4, only the definition of *xlo/xhi* and *zlo/zhi* are seen.

Parameters	Values
units	metal
dimension	2
thermo	10
pair_style	tersoff
timestep	0.001
temperature	0 - 4500
run	10000
xlo/xhi	-5/20
zlo/zhi	-5/26
mass (C)	12.01

Table 4: Parameters and values to replicate simulation.

4.2.1 Analysis

Like in the CNT simulation, the system begins at a base state where all the atoms are at rest and the temperature is at 0 K. The target temperature for the simulation was 4500 K, which was 1000 K higher than the CNT simulation. The System Temperature graph is displayed in Figure 23 and shows the temperature over the duration of the simulation, being 10 psec. The initial heating to the target temperature of 4500 K occurs in the first 3 psec. After which, the temperature fluctuates between 3800 K and 5500 K for the remainder of the simulation. During the initial heating the temperature spikes to nearly 9000 K, causing the atoms on the edge of the nanoribbon to break away from the main structure. This could possibly be because the structure began metaling but as the temperature decreased the stray atoms rejoined the bulk.

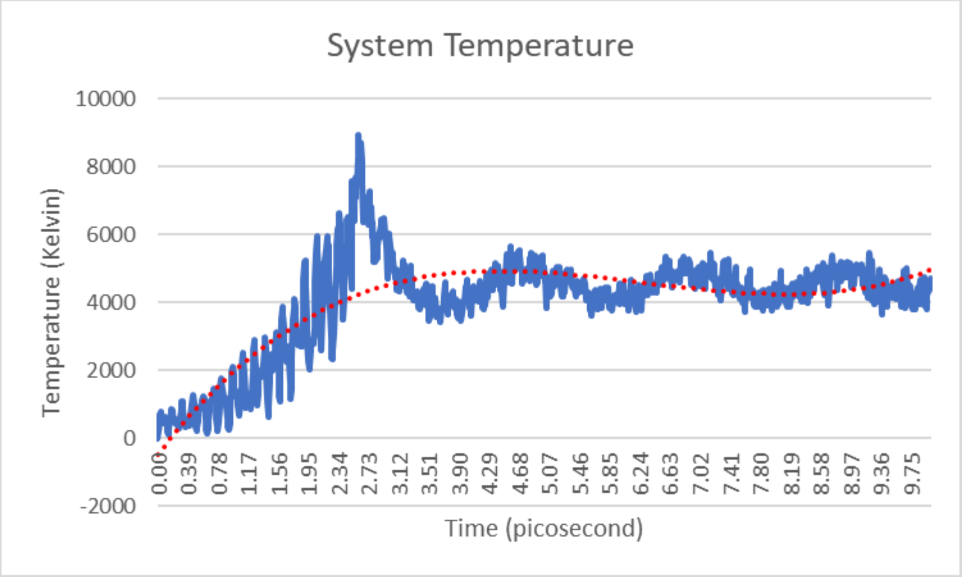


Figure 23: Graph of the temperature vs time during GNR simulation

During the duration of the simulation there was one case of diffusion. It occurred after the initial heating when the system reached its target temperature of 4500 K. Figure 24(a) displays the structure at time 3.20, just before atom 84 starts diffusing. The diffusion itself begins at time 3.22, and lasts 0.04 psec, like the cases in the CNT. At 3.22 psec and 4373.2381 K atom 84 appears to break at least one of its bonds and begins diffusing. In Figure 24(b), atom 84 is in the middle of diffusing and in Figure 24(c) the atom occupies its new lattice site.

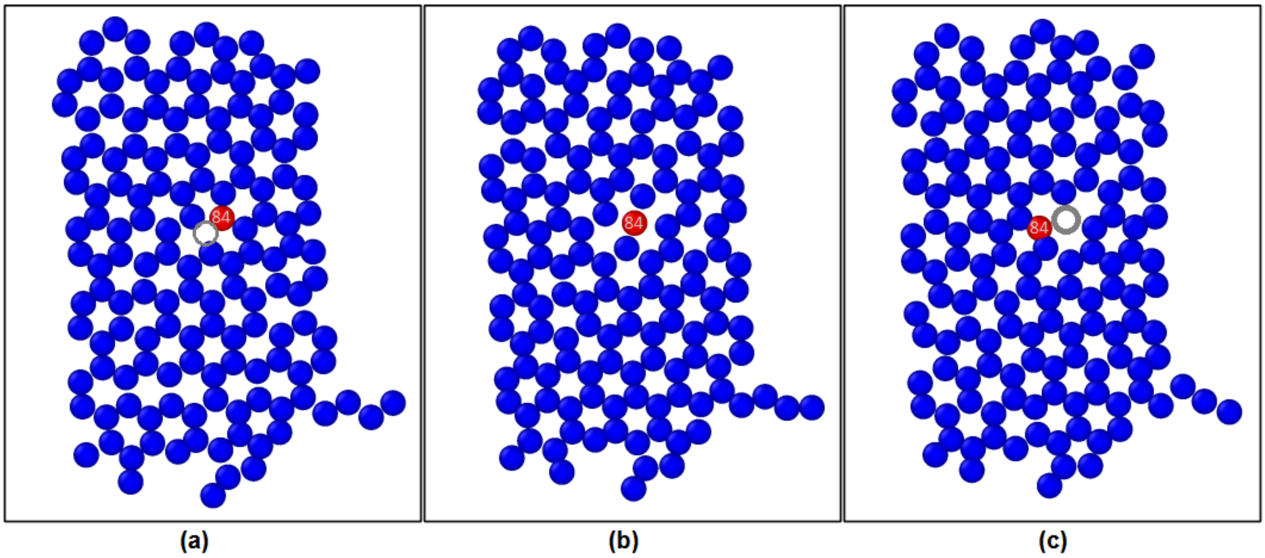


Figure 24: (a) Structure at 3.20 psec and 4813.8803 K before diffusion. (b) Structure at 3.24 psec and 4526.7286 K with atom 84 diffusing. (c) Structure at 3.26 psec and 4190.6406 K after diffusion.

5 Discussion

5.1 Evaluation

After carrying out the simulations described in this project it can be deduced that digital experiments are a valuable tool in research. The project took modern, real-world materials and laid down the basis for how to build them in a digital environment. Additionally, the results presented demonstrate that natural phenomena can be seen from running simulations employing the structures. It can be observed from the simulations in the previous chapter that carbon nanostructures, such as nanotubes and nanoribbons, that have a vacancy tend to experience diffusion, especially when energy is applied to the structure. This same natural phenomenon was discussed in The Science and Engineering of Materials textbook, a reputable source used to teach Materials Science [5, 141-152].

It is especially important to point out that the majority of a simulation is not explicitly defined. Besides some parameter values and scripting, the molecular dynamics is left to the software. This means that while hypotheses and predictions can be made before the simulation is ran, there is no guaranteed outcome, making much of the research informed trial and error. This makes simulations even more closely attuned with laboratory experiments. However, it is a fair conclusion to say that the more knowledge and experience the researcher has in the topic, the more useful and predictable a simulation can be.

Even though the results of this research had legitimate aspects, there were several facets that deserve scrutiny. For example, the temperatures used in the simulations were extremely high. In many situations the temperature of the system far exceeded the melting point of the material without actually melting the structure. Although, this could be due to the short duration the system was exposed to the extreme heat, not having enough time for the interior atoms to break out of their bonds. Nonetheless, further inspection of the temperatures and melting points should be performed in order to better determine the applicability and legitimacy of the results.

Another puzzling aspect of the results was the lack of diffusion in the structures. With the CNT having three instances and the GNR only having one. These were the maximum number of diffusions for each structure during the entire project. It was predicted that diffusing atoms would be far more frequent once the temperature approached the material's melting point, but it did not seem to make a difference. It is possible that due to the short length of the simulations, only 10 picoseconds, there wasn't enough time for more diffusions to happen.

5.1.1 Continuation

If research on this topic was to continue there would be several aspects that could be improved and further developed to create more interesting results. One such improvement would be the optimization of the CNT and GNR. There are several optimization techniques that can be used to calculate better lattice parameters making the simulation's results more stable and realistic.

It would also be interesting to see how the structures behave under the presence of more than one vacancy. Running experiments that monitor the rate of diffusion in the structures as well as thermal and surface area related properties. The possibility of introducing different defects, like substitutional atoms, would also be quite compelling. If the substitutional atoms existed in the structures at any significant concentration, the properties of the material could be altered, like the steel alloy composed of iron and C.

Lastly, running simulations containing larger, more realistic structures would make the results useful for other researchers studying these materials. For example, adding fullerene caps to the ends of the nanotubes would undoubtably change its properties. This would be an interesting comparison to the capless structures used in this project.

5.2 Reflection

Overall, I believe that this thesis had a successful outcome, producing valid results and a means for other amateur researchers to start conducting their own simulations. From a personal perspective I can confidently say that this project has been an incredible learning experience. At the beginning of the project I did not even know that graphite was made out of carbon. Now that the research has concluded, I can set up and execute molecular dynamics simulations that can contain most of C’s allotropes.

With the research topic being from the field of materials science, I struggled through the bulk of the project. Being unfamiliar with most aspects of chemistry and mathematics, made it difficult to progress through the research. However, this same struggle provided an exceptional basis for personal learning and development. The project was preformed in a very linear fashion, taking each portion step-by-step. It was important to complete a section before moving forward, because in most cases its understanding was critical to the following.

5.2.1 Problems

As mentioned above, the main obstacle of this project was the lack of knowledge in the field of study. Coming from a background in Information Technology there was a steep learning curve when familiarizing myself with the topic and environment. For example, almost all the software tools used in this project (LAMMPS, ASE, OVITO) required a base knowledge in molecular dynamics, chemistry, physics, or all of the above. While the distributors of the software did provide documentation and tutorials, the explanations were targeted towards professional academic researchers. It was a very long and difficult process to even acquire the minimal understanding to effectively use the software’s tools.

As an individual, my meager theoretical background hindered many aspects of the research. On several occasions the use and understanding of mathematical techniques would have been important to gain valuable insights about how phenomena was modeled. For example, the Tersoff *pair_style* provides equations that are used to compute interactions between atoms. Unfortunately, because of my low level of math knowledge, I was not able to gain an intuition about how these equations operate.

Lastly, like all things, time was a constraint. With more time it would have been possible to further familiarize myself with the topics and tools. This would not only make the simulations more efficient and effective, but it would also provide better grounds for analysis of the results. However, even without a better understanding of the theoretical aspects, if given more time, continued optimization of the structure and simulation parameters alone would yield better results.

6 Conclusion

Many research groups in academia are currently studying the properties of carbon nanostructures and their applications. Conducting laboratory experiments on an extremely small scale can be expensive and difficult to control. Simulations such as the ones executed in this project demonstrate how molecular dynamics simulations can be applied to augment research. Using simulations and modeling, the phenomena in question can be first tested in a digital environment to provide initial insights that can be used later during a laboratory experiment.

The primary goal of this project was to analyze the presence of vacancies in graphene and carbon nanotubes using a completely digital environment. The main software tools employed were Atomic Simulation Environment (ASE) and Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS). ASE was used for the generation of the structures, while LAMMPS carried out the molecular dynamics simulations. As a secondary objective, the project aimed to provide an general introduction to simulations and display the value that they can offer to scientific research. This project’s results demonstrated how vacancies can cause the natural phenomena of diffusion when energy is added to the structures containing them. Additionally, the project as a whole acts as a guide for amateur researches and science enthusiasts that want to start doing simulations of their own.

References

1 Senese F. Who discovered carbon? [online]. General Chemistry Online. Frostburg, Maryland: Frostburg State University; September 2000. URL: http://antoine.frostburg.edu/chem/senese/101/inorganic/faq/discovery-of-carbon.shtml. Accessed 5 April 2018.

2 Ren Z, Lan Y, Wang Y. Aligned Carbon Nanotubes. New York City, NY: Springer Publishing Company; 2013.

3 Helmenstine A. Chemical Composition of the Human Body [online]. ThoughtCo. New York City, NY: Dotdash; October 2017. URL: https://www.thoughtco.com/chemical-composition-of-the-human-body-603995. Accessed 7 April 2018.

4 Ashish. Why Is Graphite Soft, But Diamond Is So Hard? [online]. ScienceABC; 2017. URL: https://www.scienceabc.com/pure-sciences/graphite-soft-diamond-structure-properties-hard-carbon-allotrope-tetrahedral-layers.html. Accessed 8 April 2018.

5 Askeland D, Wright W. The Science and Engineering of Materials. Boston, MA: Cengage Learning; 2016.

6 Plait P. Want to Make a Diamond? Slam an Asteroid Into the Earth! [online]. Slate. New York City, NY: The Slate Group; March 2016. URL: http://www.slate.com/blogs/bad_astronomy/2016/03/17/asteroid_impacts_create_tiny_diamonds.html?via=gdpr-consent. Accessed 10 April 2018.

7 Woodford C. Graphene [online]. EXPLAINTHATSTUFF; March 2018. URL: https://www.explainthatstuff.com/graphene.html. Accessed 12 April 2018.

8 Novoselov K. Professor Konstantin Novoselov interviewed about Graphene [online]. Graphene Flagship; October 2017. URL: https://graphene-flagship.eu/material/Pages/The-history-of-graphene.aspx. Accessed 11 April 2018.

9 Fogelstrom M. Mikael Fogelstrom: Graphene, from a layer of atoms to applications [online]. TED Institute. New York City, NY; December 2014. URL: https://www.youtube.com/watch?v=675eM-V8t08. Accessed 13 April 2018.

10 Harris P. Carbon nanotube science and technology [online]. Peter Harris's Homepage. Reading, England: University of Reading; September 2017. URL: http://www.personal.rdg.ac.uk/~scsharip/tubes.htm. Accessed 14 April 2018.

11 Jansen R, Wallis P. Manufacturing, Characterization and Use of Single Walled Carbon Nanotubes. Material Matters 2009;4(1):23-27.

12 Wang A. Spinning Methods for Carbon Nanotube Fibers. Cincinnati, OH: University of Cincinnati; March 2014.

13 Pasquali M. Spinning nanotube fibers at Rice University [online]. Huston, TX: Rice University; January 2013.
URL: https://www.youtube.com/watch?v=4XDJC64tDR0. Accessed 16 April 2018.

14 Coffey T. Intro to Vacancy Defects [online]. Boone, NC: Appalachian State University; June 2017.
URL: https://www.youtube.com/watch?v=s2M2eYha8u4. Accessed 20 April 2018.

15 Winsberg E. Computer Simulations in Science [online]. Stanford, CA: Stanford University; April 2015.
URL: https://plato.stanford.edu/entries/simulations-science/. Accessed 25 April 2018.

16 ASE-developers. The atomic simulation environment—a Python library for working with atoms. Journal of Physics: Condensed Matter 2017;29(27).

17 Plimpton S. Fast Parallel Algorithms for Short-Range Molecular Dynamics. Journal of Computational Physics 1995;117:1-19.

18 Stukowski A. Visualization and analysis of atomistic simulation data with OVITO–the Open Visualization Tool. Modelling and Simulation in Materials Science and Engineering 2010;18(1).

Tersoff Potential File

```
# DATE: 2009-05-05 CONTRIBUTOR: Felix Ulomek, ulomek@imm.rwth-aachen.de CITATION: Erhart and Albe, Phys Rev B, 71, 035211 (2005)
# Si and C mixture, parameterized for Tersoff potential
# this file is from ulomek(at)imm.rwth-aachen.de
# values are from Erhart/Albe - Phys Rev B, 71, 035211 (2005)

# m=n=beta=1
# R,D,gamma,c,d ->same
# lambda1=beta*sqrt(2*S)
# lambda2=beta*sqrt(2/S)
# lambda3=2*mu=0
# costheta0=-h
# A=D0/(S-1)*exp(lambda1*r0)
# B=S*D0/(S-1)*exp(lambda2*r0)

# Tersoff parameters for various elements and mixtures
# multiple entries can be added to this file, LAMMPS reads the ones it needs
# these entries are in LAMMPS "metal" units:
#   A,B = eV; lambda1,lambda2,lambda3 = 1/Angstroms; R,D = Angstroms
#   other quantities are unitless

# format of a single entry (one or more lines):
#   element 1, element 2, element 3,
#   m, gamma, lambda3, c, d, costheta0, n,
#   beta, lambda2, B, R, D, lambda1, A

C   C   C    1 0.11233 0 181.91 6.28433 -0.5556 1
             1 1.93090093 175.426651 2 0.15 4.18426232 2019.8449

Si  Si  Si   1 0.114354 0 2.00494 0.81472 -0.259 1
             1 1.53810493 219.521624 2.82 0.14 2.83318929 2145.7128
```

```
Si  Si  C    1 0.011877 0 273987 180.314 -0.68 0
             0 0 0 2.4 0.2 0 0

Si  C   C    1 0.011877 0 273987 180.314 -0.68 1
             1 1.76807421 225.189481 2.4 0.2 3.26563307 1779.36144

C   Si  Si   1 0.011877 0 273987 180.314 -0.68 1
             1 1.76807421 225.189481 2.4 0.2 3.26563307 1779.36144

C   Si  C    1 0.11233 0 181.910 6.28433 -0.5556 0
             0 0 0 2 0.15 0 0

C   C   Si   1 0.011877 0 273987 180.314 -0.68 0
             0 0 0 2.4 0.2 0 0

Si  C   Si   1 0.114354 0 2.00494 0.81472 -0.259 0
             0 0 0 2.82 0.14 0 0
```