







# **UNIVERSIDAD DE INVESTIGACIÓN DE TECNOLOGÍA EXPERIMENTAL YACHAY**

**Escuela de Ciencias Químicas e Ingeniería**

**TÍTULO: Study of diffusion of alkanes through punctually  
modified nanotubes.**

Trabajo de integración curricular presentado como requisito para  
la obtención del título de Químico

**Autor:**

Muriel Sánchez Adrian Alexander

**Tutor:**

Ph. D. Terencio Thibault

Urcuquí, Diciembre 2020



**SECRETARÍA GENERAL**  
**(Vicerrectorado Académico/Cancillería)**  
**ESCUELA DE CIENCIAS QUÍMICAS E INGENIERÍA**  
**CARRERA DE QUÍMICA**  
**ACTA DE DEFENSA No. UITEY-CHE-2020-00059-AD**

A los 16 días del mes de octubre de 2020, a las 12:00 horas, de manera virtual mediante videoconferencia, y ante el Tribunal Calificador, integrado por los docentes:

<b>Presidente Tribunal de Defensa</b>	Dra. LOPEZ GONZALEZ, FLORALBA AGGENY , Ph.D.
<b>Miembro No Tutor</b>	Dra. VARELA SALAZAR, SOLMAR ALEXANDRA , Ph.D.
<b>Tutor</b>	Dr. TERCICIO THIBAUT , Ph.D.

El(la) señor(ita) estudiante **MURIEL SANCHEZ, ADRIAN ALEXANDER**, con cédula de identidad No. **1718042086**, de la **ESCUELA DE CIENCIAS QUÍMICAS E INGENIERÍA**, de la Carrera de **QUÍMICA**, aprobada por el Consejo de Educación Superior (CES), mediante Resolución **RPC-SO-39-No.456-2014**, realiza a través de videoconferencia, la sustentación de su trabajo de titulación denominado: **STUDY OF DIFFUSION OF ALKANES THROUGH PUNCTUALLY MODIFIED NANOTUBES**, previa a la obtención del título de **QUÍMICO/A**.

El citado trabajo de titulación, fue debidamente aprobado por el(los) docente(s):

<b>Tutor</b>	Dr. TERCICIO THIBAUT , Ph.D.
--------------	------------------------------

Y recibió las observaciones de los otros miembros del Tribunal Calificador, las mismas que han sido incorporadas por el(la) estudiante.

Previamente cumplidos los requisitos legales y reglamentarios, el trabajo de titulación fue sustentado por el(la) estudiante y examinado por los miembros del Tribunal Calificador. Escuchada la sustentación del trabajo de titulación a través de videoconferencia, que integró la exposición de el(la) estudiante sobre el contenido de la misma y las preguntas formuladas por los miembros del Tribunal, se califica la sustentación del trabajo de titulación con las siguientes calificaciones:

Tipo	Docente	Calificación
Tutor	Dr. TERCICIO THIBAUT , Ph.D.	10,0
Miembro Tribunal De Defensa	Dra. VARELA SALAZAR, SOLMAR ALEXANDRA , Ph.D.	10,0
Presidente Tribunal De Defensa	Dra. LOPEZ GONZALEZ, FLORALBA AGGENY , Ph.D.	10,0

Lo que da un promedio de: **10 (Diez punto Cero)**, sobre 10 (diez), equivalente a: **APROBADO**

Para constancia de lo actuado, firman los miembros del Tribunal Calificador, el/la estudiante y el/la secretario ad-hoc.

*Certifico que en cumplimiento del Decreto Ejecutivo 1017 de 16 de marzo de 2020, la defensa de trabajo de titulación (o examen de grado modalidad teórico práctica) se realizó vía virtual, por lo que las firmas de los miembros del Tribunal de Defensa de Grado, constan en forma digital.*

**MURIEL SANCHEZ, ADRIAN ALEXANDER**  
**Estudiante**

Dra. LOPEZ GONZALEZ, FLORALBA AGGENY , Ph.D.  
**Presidente Tribunal de Defensa**

Dr. TERENCE THIBAUT , Ph.D.  
**Tutor**


Dra. VARELA SALAZAR, SOLMAR ALEXANDRA , Ph.D.  
**Miembro No Tutor**

CIFUENTES TAFUR, EVELYN CAROLINA  
**Secretario Ad-hoc**

## AUTORÍA

Yo, **ADRIAN ALEXANDER MURIEL SÁNCHEZ**, con cédula de identidad 1718042086, declaro que las ideas, juicios, valoraciones, interpretaciones, consultas bibliográficas, definiciones y conceptualizaciones expuestas en el presente trabajo; así cómo, los procedimientos y herramientas utilizadas en la investigación, son de absoluta responsabilidad de el/la autor(a) del trabajo de integración curricular. Así mismo, me acojo a los reglamentos internos de la Universidad de Investigación de Tecnología Experimental Yachay.

Urcuquí, diciembre de 2020.



Adrian Alexander Muriel Sánchez  
CI: 1718042086





## AUTORIZACIÓN DE PUBLICACIÓN

Yo, **ADRIAN ALEXANDER MURIEL SÁNCHEZ**, con cédula de identidad 1718042086, cedo a la Universidad de Tecnología Experimental Yachay, los derechos de publicación de la presente obra, sin que deba haber un reconocimiento económico por este concepto. Declaro además que el texto del presente trabajo de titulación no podrá ser cedido a ninguna empresa editorial para su publicación u otros fines, sin contar previamente con la autorización escrita de la Universidad.

Asimismo, autorizo a la Universidad que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior

Uruguay, diciembre de 2020.



Adrian Alexander Muriel Sánchez  
CI: 1718042086



## Abstract

This research thesis aims to evaluate by Molecular Dynamics simulations the diffusion process of n-alkanes going through modified carbon nanotubes based on the adjustment of the van der Waals potentials of some nanotube atoms. For the design of the system was used Python 3.7, and Molecular Dynamics simulations were performed by NAMD 2.13. Infinite narrow single-walled carbon nanotubes (SWCNTs) were simulated by periodic boundary conditions, and the addition of a stochastic thermostat allowed the evaluation of diffusion through infinite cylindrical pores with corrected fluid-wall interactions. The behavior and distribution of n-alkanes inside the SWCNTs were characterized by the study of their dihedral angles and centers of mass. There were observed gauche-gauche' pairs which are energetically and sterically less favorable than the stable trans-trans pairs. The diffusion process of alkanes was characterized by their total translational distances and self-diffusion coefficients, the later calculated from their mean squared displacements (MSD). Simulations with nonmodified SWCNTs showed self-diffusion coefficients in the  $10^{-9}$  m<sup>2</sup>/s order of magnitude, while modified nanotubes showed variations in a wide range from these values. The effect of the  $\epsilon$  parameter from van der Waals (vdW) interactions, was different for small and large alkanes because of the influence of conformational rearrangements over the diffusion process. Then, there was favored either the attraction over the modified atoms or the conformational constraints imposed by them. These differences in diffusivities can indicate a route to future engineering of devices for separating alkane molecules based on their size.

**Keywords:** n-alkanes, carbon nanotubes, Molecular Dynamics, diffusion coefficient.



## Resumen

Esta tesis de investigación tiene como objetivo evaluar mediante simulaciones de dinámica molecular el proceso de difusión de alcanos lineales a través de nanotubos de carbono modificados en base al ajuste de los potenciales de van der Waals de algunos de los átomos del nanotubo. Para el diseño del sistema se utilizó Python 3.7, y las simulaciones de dinámica molecular fueron realizadas en NAMD 2.13. Nanotubos de pared simple (SWCNTs por sus siglas en inglés) rectos e infinitos fueron simulados mediante condiciones de frontera periódicas, y la adición de un termostato estocástico permitió el estudio de la difusión a través de poros cilíndricos infinitos tomando en cuenta las interacciones entre el alcano y las paredes del nanotubo. El comportamiento y distribución de los alcanos lineales dentro de los nanotubos fueron caracterizados mediante el estudio de sus ángulos dihedrales y centros de masa. Se evidenció la aparición de pares gauche-gauche' energética y estéricamente menos probables que las conformaciones trans. El proceso de difusión de los alcanos lineales fue caracterizado mediante el cálculo de sus coeficientes de auto-difusión obtenidos a partir de su desplazamiento medio cuadrático (MSD por sus siglas en inglés). Las simulaciones con nanotubos no modificados mostraron coeficientes de auto-difusión en el orden de  $10^{-9} \text{m}^2/\text{s}$  mientras que los nanotubos modificados mostraron variaciones en un gran rango en comparación con estos valores. El efecto sobre la variación de las interacciones de van der Waals (vdW) fue diferente para alcanos grandes y pequeños debido a la diferente influencia que tienen los cambios conformacionales en el proceso de difusión sobre estos. Como resultado o son favorecidos los procesos de adsorción sobre los átomos modificados en las paredes del nanotubo o son favorecidas las restricciones que éstos impusieron sobre los cambios conformacionales. Las diferencias registradas en la difusividad de los alcanos pueden indicar un nuevo camino en el diseño de dispositivos para separar alcanos basados en su tamaño.

**Palabras clave:** n-alcanos, nanotubos de carbono, dinámica molecular, coeficiente de difusión.



## **Acknowledgements**

I would like to thank my advisor, Ph.D. Thibault Terencio for all the time, patience, and willingness to help along the present work; even in what seemed like the end of the world, especially to me. Thank you for giving me all the time and space I needed, and moreover for trusting me.

A very special gratitude to the professors that in the first years of my career inspired me to continue in this branch of science, Ricardo, Hortensia and Antonio. Also, I would like to express my gratings to Stalyn A., Tyson U., and Joan B. for inspiring and empowering me to become the best version of myself, for serving as my role models, and for all the joy and advice they shared with me. To some other kind professors that borrowed me so much of their own souls, giving me hope that everywhere will be a light on, even in their own home, thank you for everything you had done for us. I hope one day I could be as sympathetic and incredible as you all. To my parents, thank you for supporting me along the past five years and for all the love, hard work, and sacrifices made during the last 22 years. And last but not least, thank you Erick, Nicolás and Pepa for never ever, ever let me down.

Thank you.

Adrian Muriel





## **Dedication**

To Daniela, Maykel, Alisson, Steven and Nicol. Never stop dreaming.



# Contents

<b>List of Figures</b>	<b>xviii</b>
<b>List of Tables</b>	<b>xxi</b>
<b>1 General Introduction</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Problem Statement . . . . .	2
1.3 General and Specific Objectives . . . . .	3
<b>2 Theoretical Background</b>	<b>5</b>
2.1 Molecular Dynamics . . . . .	5
2.1.1 Verlet-like algorithms . . . . .	7
2.1.2 Potential functions . . . . .	9
2.2 NAMD . . . . .	12
2.2.1 Bonded Potentials . . . . .	14
2.2.2 Non-bonded Potentials . . . . .	15
2.2.3 Periodic Boundary Conditions . . . . .	17
2.3 Carbon Nanotubes . . . . .	19
2.4 Alkanes . . . . .	22
2.4.1 Chemical-Physical properties . . . . .	22
2.4.2 Conformational study for alkanes . . . . .	24
<b>3 Methodology</b>	<b>27</b>
3.1 Nanotube design . . . . .	27
3.2 Alkane design . . . . .	31

3.3	NVE simulations details . . . . .	31
3.4	NVT simulation details . . . . .	33
3.5	Characterization of alkane conformations during the trajectory . . . . .	34
3.5.1	Torsion angles $[\phi, \psi]$ plot . . . . .	34
3.5.2	End-to-end distance . . . . .	36
3.6	Determination of alkane diffusion coefficients . . . . .	38
<b>4</b>	<b>Results &amp; Discussion</b>	<b>41</b>
4.1	Effect of chain length . . . . .	41
4.2	Effect of Temperature . . . . .	45
4.3	Most favorable conformations . . . . .	47
4.4	Location distribution of alkanes . . . . .	54
4.5	Diffusion coefficients . . . . .	59
<b>5</b>	<b>Conclusions &amp; Outlook</b>	<b>67</b>
<b>A</b>	<b>Diffusion Coefficients under microcanonical conditions</b>	<b>71</b>
<b>B</b>	<b>Algorithms</b>	<b>77</b>
	<b>Bibliography</b>	<b>105</b>

# List of Figures

2.1	Lennard-Jones potential along the distance of separation between two atoms for different $\epsilon_{i,j}$ and $\sigma_{i,j}$ values. . . . .	12
2.2	Bond stretching (upper left), bond angle (upper middle), torsional angle (upper right), van der Waals (bottom left) and Coulombic interactions (bottom right) parameters used to define the CHARMM force field. . . . .	17
2.3	Tetragonal periodic cell with primitive vectors $\vec{a}_1$ , $\vec{a}_2$ , and $\vec{a}_3$ (left), and replicated periodic cell (right). . . . .	18
2.4	From left to right: Lattice vectors and some chiral vectors in a graphene sheet, and a (8,0) carbon nanotube. . . . .	19
2.5	From left to right: SWCNT in the (8,8) armchair, (10,0) zigzag, and (8,4) chiral form, and a MWCNT. . . . .	20
2.6	From left to right: Butane in the anti, gauche and eclipsed conformation. . . . .	25
3.1	Periodic box used to simulate an infinite length carbon nanotube. a)Orthographic view of the carbon nanotube in the periodic box. b)Perspective view of the boundaries between the real and imaginary boxes. . . . .	28
3.2	Nanotubes used in simulations. a)Non-modified carbon nanotube. b)Modified carbon nanotube "A". c)Modified carbon nanotube "B". d)Modified carbon nanotube "C". e)Modified carbon nanotube "D". . . . .	30
3.3	Torsion angles $\phi$ and $\psi$ describing an alkane molecule. a)First pair of torsion angles for octane. b)Second pair of torsion angles for octane. c)All possible pairs of torsion angles describing an octane. . . . .	35
3.4	On the left: Scatter plot for all $(\phi, \psi)$ pairs obtained during a MD simulation. On the right: The equivalent density plot obtained by the Two-Dimensional Kernel Density Estimation in R. . . . .	36

3.5	End-to-end distances for alkanes. a)Octane most stable conformation. b)Octane with 50% folding degree. c)Hypothetical octane with 100% folding degree. . . . .	37
3.6	Schematic representation of MSD calculation over production time. . . . .	40
4.1	Accumulative translational distance for microcanonical simulations of single alkane molecules inside a non modified (NM) CNT with initial temperatures equal to 10 K, 298 K, 318 K and 398 K, respectively. . . . .	42
4.2	Accumulative translational distance for canonical simulations of single alkane molecules inside a non modified (NM) CNT at 298 K. . . . .	43
4.3	Accumulative translational distance for canonical simulations of alkane molecules inside a non modified (NM) CNT, performed at 298 K, and their corresponding number density. . . . .	44
4.4	On the left: Accumulative translational distance for microcanonical simulations with initial temperature 298 K and $\varepsilon$ equal to 0.3 kcal mol <sup>-1</sup> for various modified CNT types. On the right: Accumulative translational distance for canonical simulations at 298 K and $\varepsilon$ equal to 0.2 kcal mol <sup>-1</sup> for various modified CNT types. . . . .	45
4.5	Accumulative translational distance under microcanonical conditions for alkanes inside nonmodified carbon nanotubes at three different temperatures. . . . .	46
4.6	Accumulative translational distance for heptane under microcanonical conditions in a nonmodified carbon nanotube for various initial temperatures. . . . .	47
4.7	Obtained 2D probability density for heptane center of mass at 378 K and 398 K ( $z$ - axis view). . . . .	48
4.8	Density ( $\phi, \psi$ ) plots for single alkane molecules inside nonmodified carbon nanotubes under microcanonical conditions with an initial temperature equal to 298 K. . . . .	49
4.9	Density ( $\phi, \psi$ ) plots for undecane in non modified carbon nanotube with an initial temperature equal to 298 K. . . . .	50
4.10	Folding degree for undecane in a nonmodified carbon nanotube under microcanonical conditions with an initial temperature equal to 298 K. . . . .	51
4.11	Density [ $\phi, \psi$ ] plots for alkane molecules in a nonmodified carbon nanotube under canonical conditions ( $T = 298$ K, $\varepsilon = 0.07$ kcal mol <sup>-1</sup> ). . . . .	52
4.12	Density [ $\phi, \psi$ ] plots for alkane molecules in type C modified carbon nanotube under canonical conditions ( $T = 298$ K, $\varepsilon = 0.20$ kcal mol <sup>-1</sup> ). . . . .	53
4.13	From left to right: Folding degree for pentane, nonane, and eicosane in C-type nanotube, under canonical conditions ( $T = 298$ K, $\varepsilon = 0.20$ kcal mol <sup>-1</sup> ). . . . .	54

4.14	<i>z</i> -axis view of density plots for center of mass positions through simulation time for undecane under microcanonical conditions with an initial temperature equal to 298 K. . . .	55
4.15	<i>z</i> -axis view of density plots for center of mass positions through simulation time for butane (top) and undecane (bottom) under microcanonical conditions with initial temperature equal to 298 K and $\varepsilon = 0.3 \text{ kcal mol}^{-1}$ . . . . .	56
4.16	<i>z</i> -axis view of density plots for center of mass positions through simulation time for pentane (top), heptane (middle) and undecane (bottom) with $\varepsilon$ equal to $0.2 \text{ kcal mol}^{-1}$ for modified atoms under canonical conditions at 298 K. . . . .	57
4.17	<i>z</i> -axis view of density plots for center of mass positions through simulation time for eicosane under canonical conditions at 298 K. . . . .	58





# List of Tables

2.1	Verlet-like algorithms and their integration methods for motion equations <sup>1</sup> . . . . .	8
2.2	Input files required by NAMD to perform a Molecular Dynamics simulation. . . . .	13
2.3	Properties of armchair, zigzag, and chiral SWCNTs. . . . .	21
2.4	Carbon to carbon distances and diameters for armchair carbon nanotubes. . . . .	22
2.5	Physical properties of some n-alkanes. . . . .	23
3.1	Force field parameters used for carbon nanotubes. . . . .	29
3.2	Force field parameters used for alkanes. . . . .	32
4.1	Obtained $D_z$ through nonmodified CNTs and reference $D$ in bulk solution. . . . .	59
4.2	Obtained $D_z$ through A-type CNTs at 298 K with $\varepsilon = 0.20$ kcal mol <sup>-1</sup> . . . . .	60
4.3	Obtained $D_z$ through B-type CNTs at 298 K with $\varepsilon = 0.20$ kcal mol <sup>-1</sup> . . . . .	61
4.4	Obtained $D_z$ through C-type CNTs at 298 K with $\varepsilon = 0.20$ kcal mol <sup>-1</sup> . . . . .	62
4.5	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.05$ kcal mol <sup>-1</sup> . . . . .	63
4.6	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.07$ kcal mol <sup>-1</sup> . . . . .	64
4.7	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.10$ kcal mol <sup>-1</sup> . . . . .	64
4.8	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.13$ kcal mol <sup>-1</sup> . . . . .	65
4.9	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.15$ kcal mol <sup>-1</sup> . . . . .	65
4.10	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.17$ kcal mol <sup>-1</sup> . . . . .	66
4.11	Obtained $D_z$ through D-type CNTs at 298 K with $\varepsilon = 0.20$ kcal mol <sup>-1</sup> . . . . .	66
A.1	Obtained $D_z$ under microcanonical conditions with $\varepsilon = 0.30$ kcal mol <sup>-1</sup> . . . . .	75



# Chapter 1

## General Introduction

### 1.1 Introduction

Alkanes are relatively simple compounds found in a great number of systems, for this reason, their characterization has demonstrated to have plenty of applications. Alkanes are one of the main components of crude-oil<sup>2</sup>, at the same time, the petroleum industry is the base of the economy of oil-dependent nations<sup>3</sup>. Alkanes are also of great importance for the understanding of biological macromolecules because of aliphatic groups constituting lipid tails, amino acids side chains, and carbohydrates<sup>4</sup>. Moreover, polymers can be studied by setting models consisting of simple alkane chains, for instance, a polyethylene molecule can be considered as an enormous n-alkane chain<sup>5</sup>. For such reasons, their characterization is of great utility in the chemical industry.

On the other hand, the advances regarding modification of carbon nanotubes are increasing exponentially. Carbon nanotubes are commonly synthesized by chemical vapor deposition, electric arc and laser deposition methods<sup>6</sup>. Some modifications of carbon nanotubes have been approached by electron donor or electron acceptor group doping, therefore, carbon nanotubes can react amphotericly depending on the doping group<sup>7</sup>. The donor groups are mainly K, Cs, and Rb, for which the deposition was found to occur in the interstitial channels of the nanotube bundles. Similarly, when reacting with acceptor groups like

HNO<sub>3</sub>, H<sub>2</sub>SO<sub>4</sub>, bromine, or iodine, the deposition of such groups may occur inside and outside the carbon nanotube. Moreover, iodine deposition occurs in helical chains along the nanotube that are formed with an approximate periodicity of 5 nm<sup>8</sup>.

Some covalent modifications involve surface electrodeposition, polymer grafting, or metal nanoparticle deposition. Meanwhile, modifications for bulk carbon nanotubes can be performed by chemical methods like reduction of diazonium salts and oxidation of aromatic amines. Besides, it is possible to cause thermally activated, microwave-assisted, and ball milling modifications, among others<sup>9</sup>. Electrodeposition of metal nanoparticles on carbon nanotubes has shown to increase their catalytic activity. The distribution of the nanoparticles can be controlled by varying experimental variables such as the potential applied, or the metal salt solution concentration. To accomplish the deposition, the carbon nanotubes need a previous oxidation stage with the use of HNO<sub>3</sub> and H<sub>2</sub>SO<sub>4</sub><sup>10</sup>. All these modified carbon nanotubes have shown interesting properties, for example, using carbon nanotubes with deposited Pt nanoparticles as electrodes have shown to result in faster diffusion rates and faster electron transfer kinetics causing an enhanced catalytic effect<sup>11</sup>.

## **1.2 Problem Statement**

The day by day advances in the synthesis of modified carbon nanotubes can lead us to the idea that stunningly precise modified carbon nanotubes will be accomplished in the near future. Such modifications may follow some distribution patterns over the original structure of the nanotube, in order to enhance its electric or transport properties. Studies for the expected transport properties of such structures could be based on the use of Molecular Dynamics guided by their intermolecular interactions. Therefore, some distributions for the modified atoms over a straight carbon nanotube structure are studied for the diffusion process of alkanes.

Linear alkanes are selected as the compounds to undergo transportation through the proposed modified

carbon nanotubes, because of their importance in the petroleum industry. Then, the main challenge for characterizing the motion of these molecules inside a proposed modified carbon nanotube would be associated with their van der Waals interactions. Firstly because their nonpolarity and secondly because by experimental approaches, van der Waals interactions remain critically unable of being characterized when it comes to terms of one atom to one atom interactions<sup>12</sup>. Besides, a common way to characterize the flux of a compound is based on its diffusion coefficient. For these reasons, the diffusion coefficients of n-alkanes inside normal and modified carbon nanotubes were evaluated by running Molecular Dynamics simulations.

### 1.3 General and Specific Objectives

This work aims to evaluate by Molecular Dynamics simulations the diffusion process of n-alkanes going through modified carbon nanotubes based on the adjustment of the van der Waals potentials of punctual nanotube atoms.

For the development of the simulations and the characterization of the behavior of alkanes in carbon nanotubes, the following specific objectives are defined:

- Design a model that describes single walled carbon nanotubes and punctually modifies their van der Waals parameters.
- Design a model to emulate an infinite long narrow single-walled carbon nanotube.
- Follow n-alkanes diffusion through modified and nonmodified carbon nanotubes using Molecular Dynamics simulations.
- Analyze the allowed and the most stable conformations experienced by alkanes in confined environments via their folding degree and dihedral angles.

- Characterize the diffusion process of n-alkanes through calculation of their translational distances and self diffusion coefficients.

## Chapter 2

# Theoretical Background

In this chapter, a brief review of some of the fundamental concepts regarding Molecular Dynamics, the program used to perform them (NAMD), and the structures involved in the simulations (carbon nanotubes and alkanes) are presented.

### 2.1 Molecular Dynamics

Molecular Dynamics has simplified the interactions between atoms to the calculation of numerically integrated equations. From Molecular Dynamics simulations, it is possible to predict the trajectories that a group of atoms will follow by the usage of iterative classical Newtonian equations of motion<sup>13</sup>. For a system with conservative forces, the classical equations of motion used in Molecular Dynamics respond to the Euler-Lagrange equation shown in Equation 2.1.

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}_i} \right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}_i} = 0. \quad (2.1)$$

In such a way that the Lagrangian,  $\mathcal{L}$ , is a function of the coordinates,  $\mathbf{q}_i$ , but also of their time

derivative,  $\dot{\mathbf{q}}_i$ . Moreover, the Lagrangian is described by both potential,  $V$ , and kinetic energies,  $K$ , resulting in  $\mathcal{L}(\dot{\mathbf{q}}_i, \mathbf{q}_i) = K - V$ . After replacing  $\mathcal{L}(\dot{\mathbf{q}}_i, \mathbf{q}_i)$  in the Euler-Lagrange equation, the Newtonian equations of motion can be obtained. For instance, considering  $m_i$  the mass of an spherical shaped atom which position of the center of mass is denoted as  $\mathbf{r}_i$ , the resulting force,  $\mathbf{f}_i$ , for any  $i$  atom is shown in Equation 2.2, being  $\ddot{\mathbf{r}}_i$ , the  $\mathbf{r}_i$  second derivative with respect to time<sup>14</sup>.

$$\mathbf{f}_i = m_i \ddot{\mathbf{r}}_i . \quad (2.2)$$

As a counterpart, the Hamiltonian,  $\mathcal{H}$ , maintain the relation  $\mathcal{H} = K + V$ . Now,  $K$  and  $V$  being functions that depend on  $\mathbf{r}_i$ , and momenta,  $\mathbf{p}_i$ , of each  $i$  atom, the next first-order differential equations of motion can be obtained from the Hamiltonian, see Equations 2.3 and 2.4. Where  $\dot{\mathbf{p}}_i$  is the momentum first derivative with respect to time. Knowing positions and momenta of atoms make possible to obtain the calculation of the trajectories for each atom<sup>15</sup>,

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} ; \quad (2.3)$$

$$\dot{\mathbf{p}}_i = \frac{-\partial V}{\partial \mathbf{r}_i} . \quad (2.4)$$

The positions and velocities obtained represent information at the microscopic level that can be used to obtain the information of the macroscopic level, for instance, energy and pressure of the system. The connection between the microscopic information and the macroscopic properties of the system is calculated by the use of Statistical Mechanics<sup>16</sup>. The same that are based on the ergodic hypothesis, which states that the average values over large times for the physical microscopic quantities that characterizes the system are equal to their statistical average values. The microcanonical ensemble ( $NVE$ ) keeps the number of particles,  $N$ , the volume,  $V$ , and the total energy of the system,  $E$ , as constants. As the  $NVE$  ensemble is considered ergodic with respect to the Hamiltonian dynamics<sup>17</sup>, it is the natural ensemble for Molecular Dynamics<sup>18</sup>. Mainly because energy is conserved, the microcanonical ensemble maintains a macroscopic



equilibrium. If the system contains more than two atoms, the differential equations are solved for a sequence of small finite time steps, being the most commonly used the Verlet-like algorithms<sup>19</sup>.

### 2.1.1 Verlet-like algorithms

The interaction between atoms can be described in terms of a potential ( $V$ ). In addition, the equivalence between the force and the potential acting over an atom caused by a set of atoms in its surrounding, is shown in Equation 2.5. Since the mass of the atoms can be considered as constant, Equation 2.6 arises following Newtonian laws of motion. Because the potential over an atom can be approximated by intermolecular and intramolecular potential equations, as will be discussed later, the acceleration over such atom can also be known<sup>19</sup>. Then, an integration method over Equation 2.6 can be used to estimate the positions for the next time step. The magnitude of the time step must be such that, meanwhile this time passes, the forces can be considered as constant. As a result, the time step,  $\delta_t$ , is usually set in the scale of femtoseconds (fs)<sup>20</sup>.

$$\mathbf{F}_i = \frac{-\partial V}{\partial \mathbf{r}_i}; \quad (2.5)$$

$$\mathbf{F}_i = m_i \mathbf{a}_i = m_i \frac{d\mathbf{v}_i}{dt} = m_i \frac{d^2 \mathbf{r}_i}{dt^2}. \quad (2.6)$$

There are three algorithms commonly used to compute molecular dynamic simulations, Verlet, Leapfrog, and velocity Verlet. Their integration methods used to solve Equation 2.6 for the next finite time step,  $t + \delta_t$ , are shown in Table 2.1. Using these velocity explicit algorithms there is a dependence on the acceleration of the system that affects the overall result. Usually, velocity-Verlet algorithm is the most widely used because its second order integration scheme that results on a third order error and also because it leads to no need of knowing future velocity or position quantities for computing quantities at the current time step. Contrary, Leapfrog method, calculates position and velocity not at the same value of the current time variable. Regular Verlet presents difficulties for representing correct collisions response

as there is no explicit velocity control. Furthermore, molecular dynamic simulations may present some uncertainty due to finite precision arithmetic used, the same that leads to approximation in long term calculations<sup>21</sup>.

Algorithm	Integration method <sup>1</sup>
Verlet	$r_{i+1} = (2r_i - r_{i-1}) + \delta_t^2 a(r_i)$
Leapfrog	$v_{i+1/2} = v_{i-1/2} + \delta_t a(r_i)$ $r_{i+1} = r_i + \delta_t v_{i+1/2}$
Velocity Verlet	$r_{i+1} = r_i + \delta_t \left( v_i + \frac{1}{2} \delta_t a(r_i) \right)$ $v_{i+1} = \left( v_i + \frac{1}{2} \delta_t a(r_i) \right) + \frac{1}{2} \delta_t a(r_{i+1})$

Table 2.1: Verlet-like algorithms and their integration methods for motion equations<sup>1</sup>.

For this reason, later on, new approaches to these methods derived into the addition of constraint conditions that limit some atoms to a specific distance, resulting in the SHAKE and RATTLE methods. The fastest vibrations are those of bonds containing hydrogen atoms, setting them as rigid decreases the range of the vibrational frequencies to higher values. The time step is set so it contains each bond vibrational frequency, as a result, larger time steps are allowed by using constrained dynamics<sup>22</sup>.

The SHAKE algorithm, resulting as a modification from the Verlet algorithm, can be characterized by the determination of a corrected new position at each time step based on an uncorrected previous one. Then, a new iterative method very similar to SHAKE called RATTLE was developed being a modified velocity Verlet algorithm, allowing to use the positions and velocities at the current time to calculate velocities and positions at the next one<sup>23</sup>. The SETTLE method was developed by Miyamoto and Kollman to solve systems with undetermined parameters that have bond stretch constraints, like water. This method is applicable for both, Verlet and Verlet velocity algorithms<sup>24</sup>. The NAMD 2.10 code for computing Molecular Dynamics uses RATTLE algorithm for bonds with hydrogen atoms involved, and SETTLE algorithm for water molecules<sup>25</sup>.

### 2.1.2 Potential functions

To compute the new positions of the atoms at each time step, there will be a need of a set of input data containing the current positions and velocities of the atoms, the type of atoms involved, their topology, partial charges, and depending on the conditions of the simulation, some other extra parameters may be required. Then, to numerically solve the integration method, the potential energy is the key aspect. In order to know the force exerted over an atom, it is necessary to know the potential energy of the atom, that will arise from the interaction with the other atoms in its surroundings, see Equation 2.5. As a result, potential energy functions are fitted to reproduce the potential employing either empirical or *ab initio* approaches<sup>26</sup>.

For potential functions to be transferable and reproducible, they are based on the use of tabulated parameters depending on a series of atom types. For this reason, potentials are usually calculated by pairwise potentials. The potential parameters can be obtained empirically by adjusting them to the observed crystal packing, or theoretically by adjusting them to potentials determined from *ab initio* calculations, among others. Potential energy functions together with the parameters that describe each atom type are commonly known as force fields<sup>26</sup>.

#### Force Fields

Force fields have been established for a variety of purposes, based on their specifications they can be considered for simulate the behavior of inorganic molecules, organic molecules, biological macromolecules as lipids, proteins, and carbohydrates, in a broader aspect, some force fields simply integrate general specifications for any type of molecules. Some of the most recognized force fields for modeling alkanes are the Smit-Karaborni-Siepmann force field (SKS), a precursor for the TraPPE force fields<sup>27</sup>, the united-atom NERD force field<sup>28</sup>, the optimized potential for liquid simulations all atoms force field (OLPS-AA)<sup>29</sup>, and the MARTINI coarse grained force field<sup>30</sup>.

Other approaches were used to achieve more fitted results under transition phase conditions, like in

the case of the Transferable Potentials for Phase Equilibria (TraPPE) force field. The same that have some distributions like the united-atom (TraPPE UA)<sup>31</sup> and the coarse-grained (TraPPE CG) for a wide variety of structures<sup>32,33</sup>. However, the two most commonly used force fields for biomolecular simulations are the CHARMM<sup>34</sup> and AMBER<sup>35</sup> force fields. The reason relies on the fact that they have derived themselves into a variety of specific force fields attending to the more particular requirements of each system to simulate.

All the interactions in a system can be described in terms of bonded and non bonded potentials. The former is divided into bond stretching, angle bending and torsion potentials, while the latter consist of the electrostatic and van der Waals potentials, as shown in Equation 2.7. Bonded interactions occur between covalently bonded atoms. The stretching of a bond between two atoms is usually approximated by a harmonic potential, as well as the bending angle between three adjacent atoms. For the torsion potential, is a common practice to describe it as a simplified Fourier series which implies the use of a cosine function<sup>36</sup>. Van der Waals interactions decay relatively fast as the atoms get separated from each other, and frequently are approximated by the Lennard-Jones potential function. Finally, electrostatic interactions are the most demanding computational process as their influence over the other atoms has to be computed for all the non-bonded atoms in the system.

$$V_{total} = V_{bond} + V_{angle} + V_{torsion} + V_{van\ der\ Waals} + V_{Coulombic} . \quad (2.7)$$

### Lennard-Jones Potential

The Lennard-Jones potential energy is a simple but powerful function that relates the distance between two noble gas atom centers with their intermolecular forces, resulting in a good approximation for the van der Waals forces between any pair of  $i$  and  $j$  atoms<sup>37</sup>. It is usually described as shown in Equation 2.8, where  $\epsilon_{i,j}$  is the value of the energy at the potential depth-well, the atom diameter  $\sigma_{i,j}$  is the distance between the two atoms at which the potential energy is equal to zero, and  $r_{i,j}$  is the current distance between

both atoms. Moreover, the distance needed to reach the minimum of the well potential is assumed to be the equilibrium distance,  $r_{(min)i,j}$ , between both atoms. As a result, if the atoms are separated by other distance than  $r_{(min)i,j}$ , they would be induced to reach their equilibrium separation to minimize the energy. The truncated interatomic potential at  $r_{i,j}$  distance is spherically symmetric, which means that it does not depend on the direction of the vector  $\overrightarrow{q_j q_i}$  between the atoms positions,  $\mathbf{q}_i$  and  $\mathbf{q}_j$ , but on their separation distance  $r_{i,j}$ <sup>38</sup>.

$$V_{LJ}(r_{i,j}) = 4\varepsilon_{i,j} \left\{ \left[ \frac{\sigma_{i,j}}{r_{i,j}} \right]^{12} - \left[ \frac{\sigma_{i,j}}{r_{i,j}} \right]^6 \right\}. \quad (2.8)$$

The plot for a generic Lennard-Jones potential energy function is shown in Figure 2.1 (red line), the effect of an increment in the value of  $\varepsilon_{i,j}$  is a greater potential well (blue line), and increasing  $\sigma_{i,j}$  results in a displacement of the equilibrium distance (green line). In a three-dimensional Cartesian coordinate system, following Equation 2.5 to know the forces of repulsion and attraction ruling the system, is needed to obtain the negative value of the potential first derivative with respect to the  $r_{i,j}$  distance, see Equation 2.9, where  $\vec{u}$  is the normalization of vector  $\overrightarrow{q_j q_i}$ <sup>1</sup>. Then, the force produced on atom  $i$  is:

$$\mathbf{F}_{LJ}(r_{i,j}) = \frac{24\varepsilon_{i,j}}{r_{i,j}} \left\{ 2 \left[ \frac{\sigma_{i,j}}{r_{i,j}} \right]^{12} - \left[ \frac{\sigma_{i,j}}{r_{i,j}} \right]^6 \right\} \vec{u}. \quad (2.9)$$

For the curve described in Figure 2.1 can be noticed that when the atoms are closer from each other than  $r_{(min)i,j}$  the repulsion increases rapidly with the reduction in distance and such interaction is the result of electronic overlapping, meanwhile once the distance is greater than  $r_{(min)i,j}$  the forces are attractive because of induced dipole interactions, and finally, as the distance becomes greater and tends to infinity, the energy tends to zero<sup>39</sup>.

---

<sup>1</sup>If  $q_i$  and  $q_j$  are the positions of two atoms, we note  $\overrightarrow{q_j q_i}$  the vector with origin  $q_i$  and end  $q_j$ .

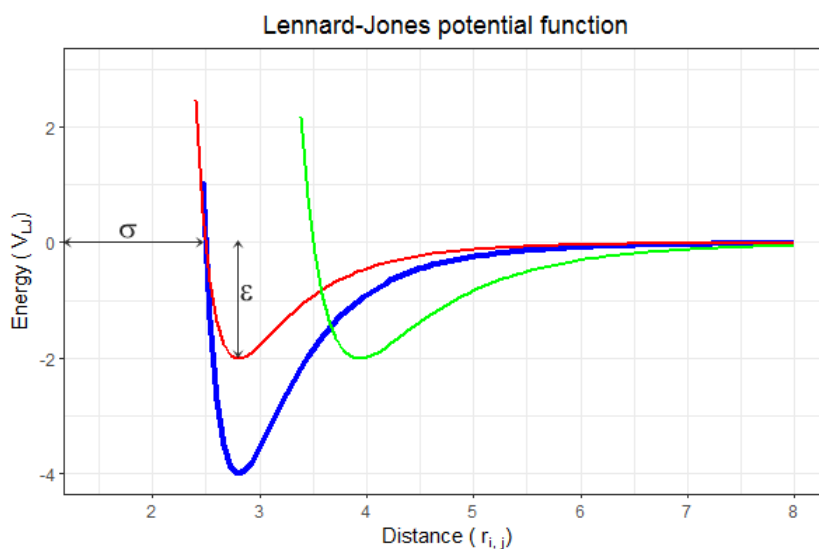


Figure 2.1: Lennard-Jones potential along the distance of separation between two atoms for different  $\epsilon_{i,j}$  and  $\sigma_{i,j}$  values.

## 2.2 NAMD

NAMD is an abbreviation for Nanoscale Molecular Dynamics and is a Molecular Dynamics code that allows the calculation of high-performance simulations on parallel computers<sup>40</sup>. NAMD was developed by the Theoretical and Computational Biophysics Group in the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign<sup>41</sup>. This code allows users to run Molecular Dynamics simulations for large biomolecular systems reaching millions of atoms. NAMD is based on the Charm++ parallel programming system and provides facilities for cross-platform through scripting in Tcl<sup>42</sup>. Parallelization is accomplished by atom decomposition, force decomposition and spatial decomposition<sup>43</sup>.

The basic requirements to run a Molecular Dynamics simulation are a Protein Data Bank file (PDB) with the coordinates of each atom, a Protein Structure File (PSF) with the topology of the molecules, a parameter file with the force field ruling the interactions, and a configuration file with all the specifications

that NAMD needs to accomplish the aim of the simulation. Some basic characteristic of each file are listed in Table 2.2<sup>44</sup>.

File	Extension	Content <sup>44</sup>	Format
PDB	.pdb	Atom types, atom names, coordinates, temperature factor and occupancy for each atom of the system.	Protein data bank format
PSF	.psf	Masses, charges, atom types and topology of the system.	Protein structure file format
Parameter file	.prm / .inp	Numerical constants needed to evaluate forces and energies	CHARMM or AMBER
Configuration file	.namd	Configuration parameters for the conditions of the simulation	Tcl language

Table 2.2: Input files required by NAMD to perform a Molecular Dynamics simulation.

Previous to perform Molecular Dynamics, minimization is carried out by the conjugate gradient algorithm, where the atoms tend to stabilize the structure by minimizing the potential energy of the system. For computing the interactions between atoms in a system, a partial charge is assigned to each atom based on its nuclear charge and the modified electronic density as a result of the molecular configuration adopted. In addition to electrostatic interactions, a Lennard-Jones potential describes the effect of van der Waals interactions<sup>43</sup>. Once the coordinates for each atom in the system are given, a thermostat is usually used to equilibrate the temperature of the system<sup>45</sup>. The trajectories obtained from this simulation depend on the initial coordinates and velocities. Initial velocities are usually set randomly from a Boltzman distribution of classical Statistical Mechanics. During the simulation, the temperature is obtained from the equipartition theorem of Classical Mechanics. Moreover, a common objective of running these simulations is to extract thermodynamic properties from the system<sup>43</sup>.

### 2.2.1 Bonded Potentials

NAMD parameter file requires the use of either CHARMM<sup>34</sup> or AMBER<sup>35</sup> format, as a result, the following constant parameters are needed for the bonded potentials.

#### Bond Potential

NAMD calculates the bonded potential between two covalently bonded atoms by a harmonic potential, as stated in Equation 2.10, where treating the bond as a spring  $K_b$  is the spring constant,  $b$  is the calculated bond length and  $b_0$  the equilibrium length for the bond, see Figure 2.2. The values of  $b_0$  and  $K_b$  between the atom types of  $i$  and  $j$  must be declared in the parameter file. Moreover,  $K_b$  is expressed in kcal/Å<sup>2</sup> and  $b_0$  is expressed in Å<sup>44</sup>.

$$V_{bond} = K_b (b - b_0)^2 . \quad (2.10)$$

#### Angle Potential

Similarly, the angle potential between three atoms is obtained by a harmonic potential, as shown in Equation 2.11, where  $K_\theta$  is the spring constant in kcal mol<sup>-1</sup> rad<sup>-1</sup>. Also,  $\theta$  represents the calculated angle (see Figure 2.2) and  $\theta_0$  the equilibrium angle, both in degrees<sup>44</sup>.

$$V_{angle} = K_\theta (\theta - \theta_0)^2 . \quad (2.11)$$

#### Dihedral Potential

The dihedral potential for four consecutively bonded atoms is computed by a cosine function, as shown in Equation 2.12, where  $K_\chi$  is a constant expressed in kcal mol<sup>-1</sup> and  $n$  is the unitless multiplicity. Also,  $\varphi$  is the angle between the plane containing the first three atoms and the plane containing the last three atoms



(see Figure 2.2) and  $\delta$  is the equilibrium dihedral angle, both expressed in degrees<sup>44</sup>.

$$V_{dihedral} = K_{\chi}(1 + \cos(n(\varphi - \delta))). \quad (2.12)$$

Sometimes biomolecular systems have structures where their planarity must be conserved. For those scenarios an improper angle between three atoms bonded to a common atom is defined. The potential has the form of a harmonic potential, described in Equation 2.13, where  $\psi$  is the improper angle,  $\psi_0$  the equilibrium angle at equilibrium, and  $K_{\psi}$  the spring constant associated which is usually large<sup>44</sup>.

$$V_{improper} = K_{\psi}(\psi - \psi_0)^2. \quad (2.13)$$

### 2.2.2 Non-bonded Potentials

Following the CHARMM<sup>34</sup> or AMBER<sup>35</sup> formats, the non-bonded potential is calculated by the Lennard-Jones potential and the Coulombic interactions, the later being independent of the force field.

#### Van der Waals interactions

The van der Waals interactions are described by the Lennard-Jones potential<sup>41</sup>, expressed in the form shown in Equation 2.14. Evaluating Equation 2.14, once the two atoms have a separation equal to  $r_{(min)i,j}$ , the value of the energy results in  $-\varepsilon_{i,j}$  or what is the same, the energy value at the minimum of the well potential. Therefore, Equation 2.14 is in agreement with the classic Lennard-Jones expression showed in Equation 2.8.

$$V_{LJ}(r_{i,j}) = \varepsilon_{i,j} \left\{ \left[ \frac{r_{min,i,j}}{r_{i,j}} \right]^{12} - 2 \left[ \frac{r_{min,i,j}}{r_{i,j}} \right]^6 \right\}. \quad (2.14)$$

The resulting potential energy will be expressed in kcal mol<sup>-1</sup>. The value of  $\varepsilon_{i,j}$  is expressed in kcal mol<sup>-1</sup>, and their distance  $r_{i,j}$  in Å (see Figure 2.2). Following the Lorentz-Berthelot combining rules,

for each two  $i$  and  $j$  interacting atoms, their corresponding  $\varepsilon_{i,j}$  value will be given by their geometric mean (see Equation 2.15). As NAMD topology uses  $r_{(min)i,j}$  instead of  $\sigma_{i,j}$ , the value of  $r_{(min)i,j}$  will be given by their arithmetic mean (see Equation 2.16)<sup>44</sup>.

$$\varepsilon_{i,j} = \sqrt{\varepsilon_i \varepsilon_j}; \quad (2.15)$$

$$r_{min,i,j} = \frac{r_{min,i}}{2} + \frac{r_{min,j}}{2}. \quad (2.16)$$

Besides, the CHARMM force field considers the non-bonded interactions for atoms in the same molecule but separated by at least three bonds employing the scaled 1-4 exclusion, which considers new adjusted  $\varepsilon_{i,j}$  and  $r_{(min)i,j}$  values<sup>46</sup>. Finally, even when by convention the  $\varepsilon_{i,j}$  values must be set as negative inside the parameter file containing the force field, from now in this work, they will be presented as positive values to keep consistency with Equation 2.14.

### **Coulombic Interactions**

For the Coulombic potential, NAMD uses the partial charges set in the PSF file, the same that must keep the concordance with the topology file used. In addition, the Coulombic potential is used for representing the long range electrostatics<sup>41</sup>. For any pair of  $i$  and  $j$  atoms, the Coulombic potential is shown in Equation 2.17, where  $q_i$  and  $q_j$  are the charges of the atoms (see Figure 2.2).

$$V_{Coulomb} = \frac{q_i q_j}{4\pi\varepsilon_0 r_{i,j}}. \quad (2.17)$$

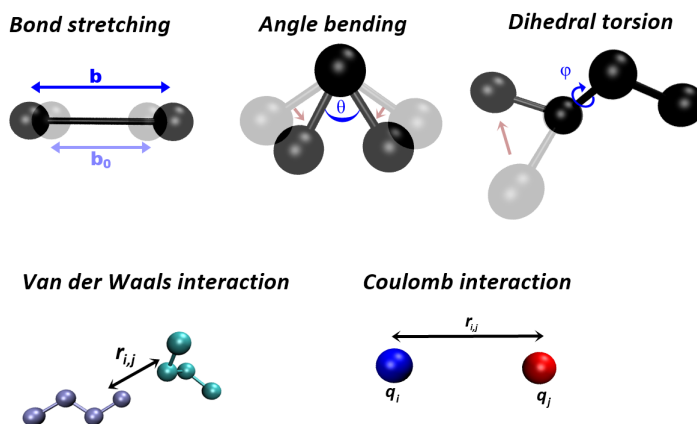


Figure 2.2: Bond stretching (upper left), bond angle (upper middle), torsional angle (upper right), van der Waals (bottom left) and Coulombic interactions (bottom right) parameters used to define the CHARMM force field.

### 2.2.3 Periodic Boundary Conditions

NAMD allows the declaration of periodic boundary conditions (PBC) resulting in the unit cell being replicated infinitely along the dimensions stated in the configuration file. For crystalline systems, depending on the side lengths and angles of the periodic cell, others than a cubic infinite Bravais lattice can be set, which means, the shape of the cell will determine its lattice<sup>47</sup>. Under PBC conditions, if a particle leaves the cell in the  $x$  direction, a copy of this will re-enter the cell by the  $-x$  direction, the same will occur with the other dimensions declared. If this is the case, there will be infinite particles to be considered in terms of non bonded interactions. To avoid this problem, van der Waals forces are truncated at some distances specified by the user. Otherwise, electrostatic interactions are non-truncated by the use of particle-mesh Ewald (PME) summation, allowing to obtain the long-range electrostatic interactions of the periodic system, whereas this implies a lower computational cost. The use of PME has been declared to function as a stabilizer for the system by conformational fluctuation suppressions<sup>48</sup>.

In the Cartesian coordinate system, given a periodic cell, any repeated lattice point can be found by the vector  $\vec{n}_r$ , being the sum of the primitive vectors that define the cell, called  $\vec{a}_1$ ,  $\vec{a}_2$ , and  $\vec{a}_3$  (see Figure

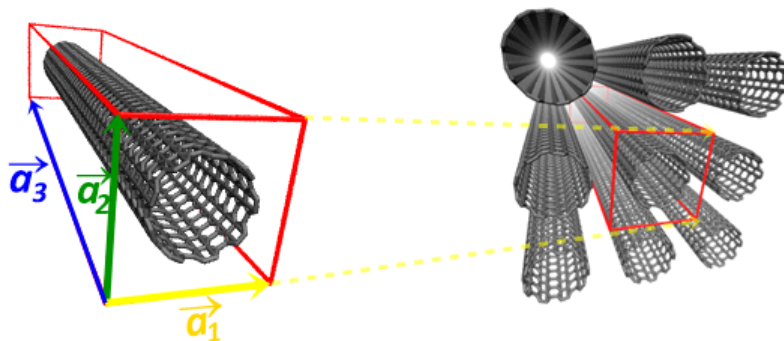


Figure 2.3: Tetragonal periodic cell with primitive vectors  $\vec{a}_1$ ,  $\vec{a}_2$ , and  $\vec{a}_3$  (left), and replicated periodic cell (right).

2.3) multiplied by the integers  $n_1$ ,  $n_2$ , and  $n_3$  respectively, as shown in Equation 2.18. Ewald summation for such periodic cell can be described as the sum of a short-range term, a long-range term, a self-energy term and a surface correction term, respectively<sup>49</sup>.

$$\vec{n}_r = n_1\vec{a}_1 + n_2\vec{a}_2 + n_3\vec{a}_3. \quad (2.18)$$

The short-range term is related to the direct sum of the periodic cell. The long-range term is based on the reciprocal sum for the unit cell of the PBC system obtained by Fourier methods, relying on the  $\beta$  parameter for load distribution in the calculations<sup>41</sup>. The self-energy term depends also on the  $\beta$  parameter that is chosen by the user to give priority to the computational cost of either the direct sum or the reciprocal sum. The surface energy term depends on the term  $\epsilon_s$ , which denotes the dielectric constant for the surroundings of the box. In biomolecular simulations, usually, the medium involved in the periodic unit cell is water which approximate value of  $\epsilon_s$  is 80, resulting in a model of description called tinfoil boundary which main condition is that the  $\epsilon_s$  value must be relatively high<sup>50</sup>.

PME allows the calculation of Ewald summation, in which the calculation of the reciprocal sum is distributed between a grid scheme according to a fast Fourier transform (FFT) via interpolation. NAMD specifically uses smooth PME (SMPE), a method that adopts B-spline functions as the basis functions for the interpolation of the charges<sup>41</sup>. Finally, the division of work between the processors is adaptive, resulting in a efficient scalability of the calculations for bigger systems<sup>48</sup>.

## 2.3 Carbon Nanotubes

Carbon nanotubes (CNTs) have gained so much recognition through the last decades due to their unique thermal, mechanical, optical, and electrical properties<sup>51</sup>. Since their discovery in 1991 by Iijima<sup>52</sup>, their use has become of greater importance in many fields such as nanomaterials, electronics, biomedicine, and this trend keeps increasing<sup>53</sup>. Because CNTs structure is formed in what looks like rolled-up graphene sheets, they share the same bond structure as graphene, but following a cylindrical-like shape which results in the loss of its planarity giving rise to a tube. Indeed each carbon atom is connected to three surrounding carbon atoms raising a network formed by a hexagonal lattice, as can be seen in Figure 2.4 (left).

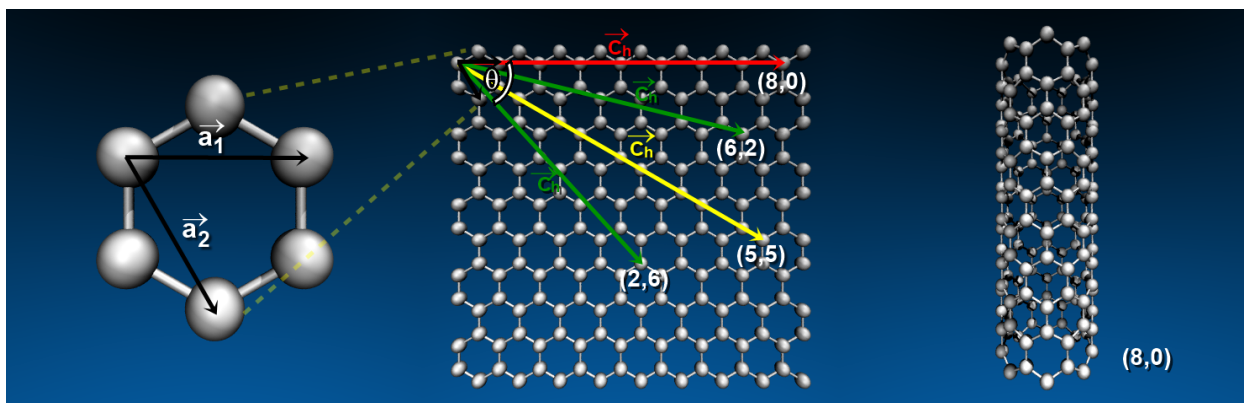


Figure 2.4: From left to right: Lattice vectors and some chiral vectors in a graphene sheet, and a (8,0) carbon nanotube.

The lattice vectors,  $\vec{a}_1$  and  $\vec{a}_2$ , are replicated by the integer numbers  $n$  and  $m$  respectively, giving rise to

the chiral vector  $\vec{C}_h$ . Because  $\vec{C}_h$  depends on the length of  $n$  and  $m$ , it is commonly represented simply as  $(n, m)$ , see Equation 2.19. Moreover, some chiral vector examples are illustrated in Figure 2.4. The angle between  $\vec{C}_h$  and the unit vector  $\vec{a}_1$  is called the chiral angle  $\theta$ . The chiral vector will have a major impact on the morphology of the CNT as it will lead to how the graphene sheet is bonded<sup>54</sup>.

$$\vec{C}_h = n\vec{a}_1 + m\vec{a}_2 = (n, m). \quad (2.19)$$

Furthermore, if there is only one graphene sheet involved in the structure of a CNT it is said to be a single-walled carbon nanotube (SWCNT), likewise, if there are more graphene sheets involved it is said to be a multiple walled carbon nanotube (MWCNT)<sup>55</sup>. For a single carbon nanotube, the disposition of its carbon atoms depends on its chiral vector, in such a way that can be obtained three different morphologies named armchair, zigzag and chiral CNTs<sup>56</sup>, see Figure 2.5. Some of their properties are shown in Table 2.3.

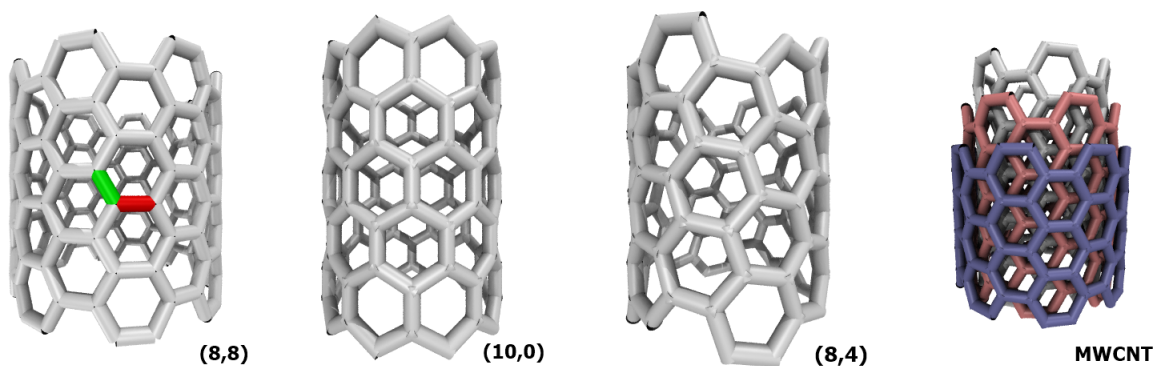


Figure 2.5: From left to right: Single-walled carbon nanotube in the (8,8) armchair, (10,0) zigzag, and (8,4) chiral form, and a multi-walled carbon nanotube.

Therefore, the properties of CNTs depend on both, its diameter and helicity, the former is related to the chiral vector by Equation 2.20 and the latter will depend on the chiral angle and will be responsible for the electrical properties exhibited by the CNT as presented in Table 2.3<sup>54</sup>.

	ARMCHAIR	ZIGZAG	CHIRAL
Chiral Parameters <sup>56</sup>			
n	Integer	Integer	Integer
m	Equal to n	Equal to 0	Integer neither n or 0
$\theta$	30°	0°	0° < $\theta$ < 30°
Electric Character <sup>54</sup>			
Conductor	Always	If n is multiple of 3	If 2n + m is a multiple of 3
Semiconductor	Conducting character predominates	If do not follow the above condition	If do not follow the above condition
Physical Properties for MWCNTs <sup>57</sup>			
Interlayer spacing	3.38	3.41	3.39 (if n = 2m)

Table 2.3: Properties of armchair, zigzag, and chiral SWCNTs.

$$d(n, m) = \frac{a_{c-c} \{3(n^2 + nm + m^2)\}^{1/2}}{\pi}. \quad (2.20)$$

The carbon-to-carbon nearest neighbor distance, referred as  $a_{c-c}$  in Equation 2.20, is usually set to 1.421 Å in every direction. But, based on the not purely cylindrical shape of the CNTs, appears a small distortion from this value. The cross-section of the CNT can be reported as being a mainly polyhedral figure. Moreover, for the armchair CNTs, the carbon-to-carbon distance is reported to be different for those carbons who are directed along the circumference,  $CC_c$ , from those who are directed along the nanotube axis,  $CC_a$ <sup>58</sup>, shown as the red bond and the green bond respectively in Figure 2.5. Some  $CC_c$ ,  $CC_a$ , diameter values computed by the PM3 geometry optimization method referred as  $d(n, m)_{PM3}$ , and diameter values obtained by Equation 2.20 referred as  $d(n, m)_{1.421}$ , are shown in Table 2.4.

Although, such distortion in the carbon-to-carbon distance does not exceed 0.008 Å reason why in this work will be used the 1.421 Å as the general distance between carbons for the armchair CNT. Besides, for an armchair CNT, the diameter can be obtained by Equation 2.21<sup>58</sup>.

$$d(n, n) = \frac{3na_{c-c}}{\pi}. \quad (2.21)$$

(n,m)	$CC_a^{58}$	$CC_c^{58}$	$d(n,m)_{PM3}^{58}$	$d(n,m)_{1.421}$
(4,4)	1.424	1.429	5.518	5.428
(5,5)	1.423	1.426	6.854	6.784
(6,6)	1.422	1.424	8.198	8.136
(8,8)	1.421	1.423	10.895	10.856
(9,9)	1.420	1.423	12.246	12.213

Table 2.4: Carbon to carbon distances and diameters for armchair carbon nanotubes.

### Van der Waals interactions in CNTs

Non-bonded interactions between two parallel carbon nanotubes has been studied following the continuum Lennard Jones model. Regarding van der Waals interactions, was founded that predominates a van der Waals distance of 3.15 Å for both SWCNTs and MWCNTs<sup>59</sup>. Moreover, the well depth of potential between CNTs is established to be 95.16 meV/Å<sup>60</sup>. Even though the diameter of the CNT affects the potential for the van der Waals interactions, especially in the first couple of nm, the van der Waals interactions become constant when a large separation is experienced as a result of large length and diameter carbon nanotubes<sup>61</sup>. Finally, double-walled CNTs van der Waals interactions acting inside the CNT can help in the encapsulation of other carbon-like molecules like fullerenes<sup>62</sup>.

## 2.4 Alkanes

### 2.4.1 Chemical-Physical properties

Alkanes are formed by carbon and hydrogen atoms only, giving rise to saturated compounds. Usually, they are colorless, tasteless, and almost odorless. As the carbon-hydrogen bonds are non polar, alkanes are non polar and their interactions are governed mainly by weak London forces. Thus, for the most common liquid alkanes at room temperature, their density ranges between 0.6 and 0.8 g/cm<sup>3</sup>, as can be seen in Table 2.5. Alkanes are not soluble in water because of their contrary polarities. Otherwise, alkanes are good solvents for non polar organic compounds including fats and oils<sup>63</sup>. One of their most remarkable



characteristics is their lack of reactivity caused by their non polar character and the high strength of their simple carbon-carbon and hydrogen-carbon bonds<sup>64</sup>. Bond energy for the carbon-carbon bond is reported to be  $348 \text{ kJ mol}^{-1}$ , and for the carbon-hydrogen bond the energy is reported as  $413 \text{ kJ mol}^{-1}$ <sup>65</sup>.

n	Name	Boiling point °C	Melting point °C	Density at 20° g mL <sup>-1</sup>
1	methane	-161.5	-183	0.424 <sup>a</sup>
2	ethane	-88.6	-172	0.546 <sup>a</sup>
3	propane	-42.1	-188	0.501 <sup>a</sup>
4	butane	-0.5	-135	0.579 <sup>a</sup>
5	pentane	36.1	-130	0.626
6	hexane	68.7	-95	0.669
7	heptane	98.4	-91	0.684
8	octane	125.7	-57	0.703
9	nonane	150.8	-54	0.718
10	decane	174.1	-30	0.730
11	undecane	195.9	-26	0.740
12	dodecane	216.3	-10	0.749
15	pentadecane	270.6	10	0.769
20	eicosane	342.7	37	0.786

Table 2.5: Physical properties of some n-alkanes<sup>63</sup>. n is the number of carbon atoms in the carbon chain.  
<sup>a</sup>Conditions under pressure, otherwise they would be in gas phase.

Alkanes are naturally obtained from crude oil deposits. Crude oil can be considered a mixture of hydrocarbons, and therefore it includes alkanes, aromatic hydrocarbons, and cyclic aliphatic hydrocarbons called naphthenes. The ratio between those hydrocarbons depends on the geographical location of the crude oil reservoir. In addition to crude oil, natural gas is also present and it is composed of small alkanes in the gaseous phase. Crude oil is subjected to a purification process by fractional distillations and chemical methods. As a result, fractions of similar hydrocarbons are obtained mainly based on their boiling point. Although, the separation of the fractions into individual components is not performed<sup>64</sup>.

Unbranched alkanes are known as linear alkanes or n-alkanes. At laboratory conditions, short n-alkanes containing from five to seventeen carbon atoms are liquids, meanwhile, the larger n-alkanes are crystalline solids with the appearance of a paraffin wax<sup>64</sup>. Some melting and boiling points for n-alkanes are listed in Table 2.5. Internal C-C bonds of linear alkanes have been found to be longer than the external C-C bonds, as reported by gas-phase electron diffraction analysis. This trend has been confirmed also by *ab initio* calculations<sup>26</sup>.

### 2.4.2 Conformational study for alkanes

The relationship between the conformation and the reactivity of a molecule is based on the energy barrier that a molecule needs to accomplish to change from the most stable conformation to the optimal conformation for a reaction to occur. Given four atoms continuously bonded, as the four carbon atoms in a butane molecule, rotation around the bond between the second and third atoms will give rise to a new conformation. Thus, depending on the angle of rotation, infinite different conformations can be accomplished. The conformation in which the first and fourth atoms are the farthest from each other is called staggered conformation, this has a torsion angle of  $180^\circ$  and it presents the lowest possible energy. Otherwise, the conformation in which the first and fourth atoms are the closest to each other is called eclipsed conformation, this presents a torsion angle of  $0^\circ$  and is the highest in energy. Therefore, at room temperature, the staggered conformation would be the most probable conformation founded in a bulk of molecules<sup>63</sup>.

For butane and larger n-alkanes, there are various staggered conformations based on the positions of the carbon atoms. For example, as shown in Figure 2.6 for a butane molecule, there are two possibilities called the anti and gauche conformations. The anti conformation has a torsion angle of  $180^\circ$  between carbons one and four, meanwhile, the gauche conformation presents a torsion angle of  $60^\circ$ <sup>66</sup>. Experimentally, the ratio between anti and gauche conformations for butane is given by a 2:1 proportion, because energetically, anti conformation is  $3.18 \text{ kJ mol}^{-1}$  more stable than gauche conformation<sup>63</sup>.

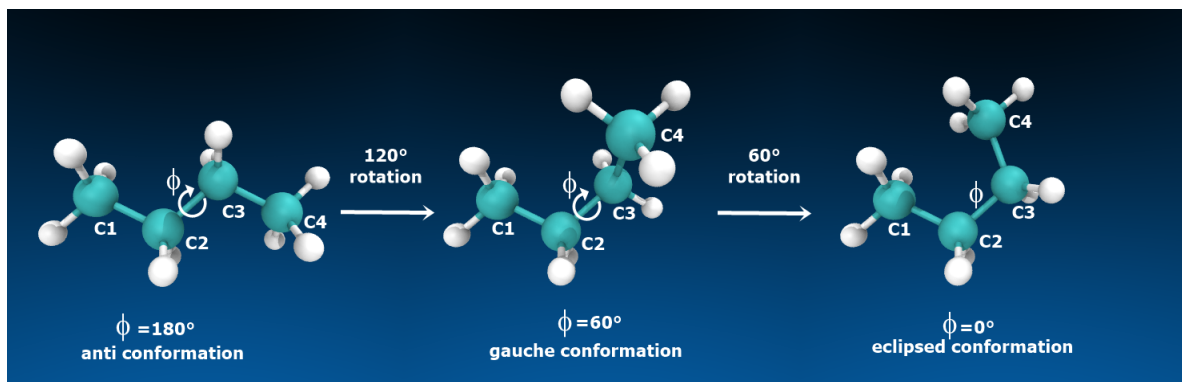


Figure 2.6: From left to right: Butane in the anti, gauche, and eclipsed conformation. The torsion angle  $\phi$  is the angle between the plane formed by C1, C2, and C3 atoms and the plane formed by the C2, C3, and C4 atoms. Starting from the anti conformation, a  $120^\circ$  rotation is needed to obtain the gauche conformation, a second rotation of  $60^\circ$  will result in an eclipsed conformation.

The above mentioned ratio is almost maintained for larger alkanes, such as heptane, which shows by electron diffraction a tendency to twist into one gauche conformation along its carbon backbone with high frequency<sup>67</sup>. Resulting in not totally straight chains, moreover, the percentage of finding its four continuous dihedral angles in gauche conformations is very low, with a 0.4% of the population. The only one gauche conformation favored in the chain is explained by the now established van der Waals attraction that causes lower dispersion energy for the hydrogen atoms in heptane<sup>67</sup>. For n-alkanes containing around seventeen carbon atoms in its backbone, it has been shown that the hairpin conformation is the most stable, again stabilized by the minimization over dispersion energy caused by hydrogen to hydrogen interactions as a result of van der Waals attraction. The hairpin conformation needs the former straight chain to undergo at least one turn, causing several gauche conformations to arise<sup>68</sup>.



## Chapter 3

# Methodology

### 3.1 Nanotube design

Nanotubes were obtained with Avogadro 1.1.0 program, in addition, geometrical optimization was performed using the Universal Force Field (UFF). The coordinates of the nanotube were aligned to be parallel with the z-axis. For the simulations, two different lengths for an armchair (9,9) carbon nanotube were chosen. The length of the first carbon nanotube was 220.312 Å, while its diameter, defined as the distance between the centers of the carbon atoms across the diameter, was 12.231 Å and used in an infinite vacuum system.

The second carbon nanotube used had a length of 99.47 Å, a diameter of 12.208 Å, and was used for simulations with periodic boundary conditions. In this case, additional bonds were set in the topology file in order to join both ends of the carbon nanotube. The established bonds between the real nanotube and the copy images help to avoid deformation in the nanotube. However, all carbon nanotubes were excluded from the dynamics by keeping them fixed in space by the *fixedAtoms* parameter in the configuration file, which avoids calculating most interactions in which all affected atoms were fixed, being the aromatic and modified carbon atoms the affected. As a result, the nanotube always keep a z-direction orientation during

the simulation. The  $z$ -axis length for the periodic box was established as the length of the carbon nanotube plus a correction value of 1.2305 Å that assures that the bond distance between the ends of the nanotube has a value of 1.421 Å as for the rest of carbon atoms.

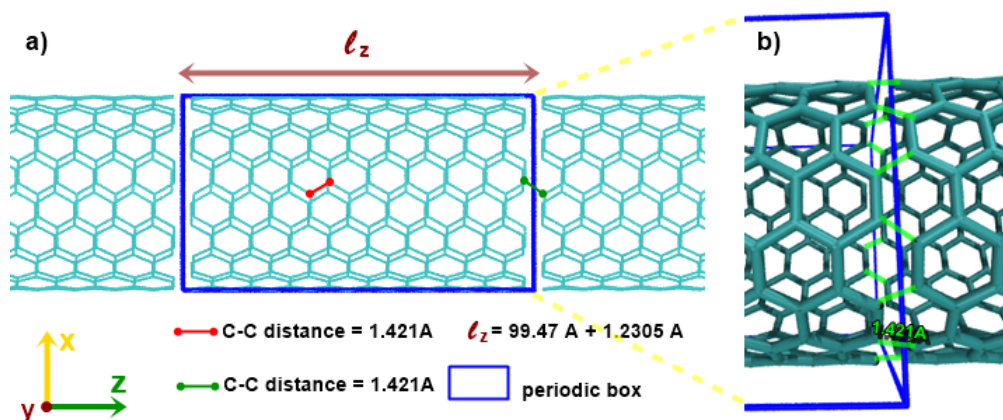


Figure 3.1: Periodic box used to simulate an infinite length carbon nanotube. a) Orthographic view of the carbon nanotube in the periodic box, with copy images replicated in the  $z$ -direction. Note that the lengths of the carbon-carbon bonds inside the real nanotube and the carbon-carbon bonds joining the real and the copy image are the same. b) Perspective view of the boundaries between the real and imaginary boxes. Green lines illustrate the bonds added to join both ends of the nanotubes.

The force field parameters for carbon nanotubes were extracted from the CHARMM22 All-Hydrogen Parameter File for Proteins and Lipids<sup>69</sup>. The force field parameters considered were those for aromatic rings, the same that are listed in Table 3.1. However, for carbon nanotubes with specifically modified atoms, their characteristic constant values were modified. Because the main interactions that govern the behavior of alkanes against the walls of the nanotube are dispersion forces, only the van der Waals contribution for the modified atoms was adjusted. Theoretically, changing such interactions it is possible to simulate the presence of other atoms in the walls of the carbon nanotube, then, their Lennard-Jones potential value for  $\epsilon$  was changed in a range that went from 0.05 kcal mol<sup>-1</sup>, which is a lower value compared to the actual aromatic carbon, to an extremely higher value of 90.0 kcal mol<sup>-1</sup>. Their distribution over the carbon nanotube was based on two main arrangements, the first one follows straight lines parallel to the nanotube

axis, and the second consists of axial rings directed along the circumference of the nanotube that were separated from each other by a certain distance in order to maintain symmetry in the distribution.

The arrangements proposed for the modified atoms in carbon nanotubes are illustrated in Figure 3.2. A, B and C arrangements are linear and axial to the nanotube extension, presenting one, two, and four straight lines of four modified atoms along the nanotube, respectively. Meanwhile, the D arrangement is transversal to the nanotube axis, following the circumference of the nanotube and forming rings of modified atoms separated from each other by seven rings of non modified carbon atoms. For these four structures modifications were made in the atom type, both in the PDB and PSF files. These modified atom types were included in the force field, therefore, their Lennard-Jones potential well depth,  $\varepsilon$ , was modified each time by convenience.

Bond Parameters				
	$K_b$ (kcal/mol/Å <sup>2</sup> )	$b_0$ (Å)		
$C_x - C_x$	305.000	1.3750		
Angle Parameters				
	$K_\theta$ (kcal/mol/rad <sup>2</sup> )	$\theta_0$ (degrees)	$K_{UB}$ (kcal/mol/Å <sup>2</sup> )	$S_{0(UB)}$ (Å)
$C_x - C_x - C_x$	40.000	120.00	35.00	2.41620
Dihedral Parameters				
	$K_\chi$ (kcal mol <sup>-1</sup> )	$n$ (unitless)	$\delta$ (degrees)	
$C_x - C_x - C_x - C_x$	3.1000	2	180.00	
Lennard-Jones Parameters				
	$\varepsilon$ (kcal mol <sup>-1</sup> )	$r_{min}/2$ (Å)	$\varepsilon_{(1-4)}$ (kcal mol <sup>-1</sup> )	$r_{min}/2_{(1-4)}$ (Å)
$C_{arom}$	0.070000	1.992400	–	–
$C_{mod}$	$X^a$	1.992400	–	–

Table 3.1: Force field parameters used for carbon nanotubes. Note that  $C_x$  can be both, either an aromatic atom or a modified atom in the nanotube. <sup>a</sup>The value of X varies in function of the simulation.

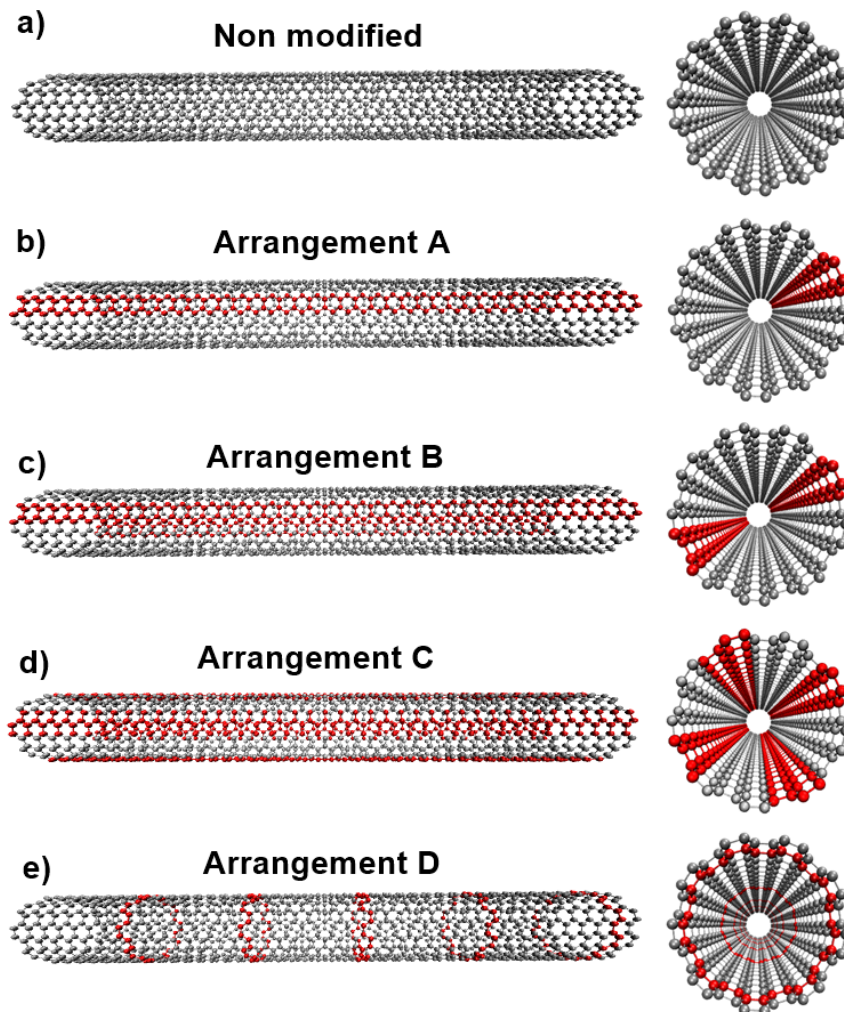


Figure 3.2: Nanotubes used for simulations. On the left, view along y-axis. On the right, view along z-axis. Gray atoms represent non modified carbon atoms. Red atoms represent modified atoms. a) Non-modified carbon nanotube. b) Modified carbon nanotube "A" containing one straight row of modified atoms formed by four consecutive modified atoms in each ring of the nanotube. c) Modified carbon nanotube "B" containing two rows of modified atoms in opposite sites of the circumference. d) Modified carbon nanotube "C" containing four rows of modified atoms located equidistantly in the circumference. e) Modified carbon nanotube "D" with modified rings (atoms consecutively bonded along the circumference) separated by seven non modified rings of atoms.



## 3.2 Alkane design

Alkanes were obtained from Avogadro 1.1.0 program and their geometry was optimized by the use of the UFF force field. Only n-alkanes corresponding to the molecular formulas  $C_4H_{10}$ ,  $C_5H_{12}$ ,  $C_6H_{14}$ ,  $C_7H_{16}$ ,  $C_8H_{18}$ ,  $C_9H_{20}$ ,  $C_{10}H_{22}$ ,  $C_{11}H_{24}$ , and  $C_{20}H_{42}$  were used. Then, they were transferred into the nanotube by vector calculations involving their Cartesian coordinates using Python 3.7. The topology for the system containing the alkanes inside the carbon nanotube was generated in VMD by using the TopoTools plugin<sup>70</sup>. The force field parameters used for alkanes were extracted from the CHARMM22 All-Hydrogen Parameter File for Proteins and Lipids<sup>69</sup>. The force field parameters used are listed in Table 3.2.

## 3.3 NVE simulations details

In a first scenario, Molecular Dynamics simulations were performed with NAMD 2.13 in Windows64, and the microcanonical ensemble (NVE) was used to follow the conformational changes for a single n-alkane molecule diffusing through a nanotube at various initial temperatures. Carbon nanotubes were fixed in space, allowing only the motion of alkane molecules. A spherical cutoff of radius  $10 \text{ \AA}$  was employed for all the pair interactions, with a smoothing function acting from a radius of  $9 \text{ \AA}$  for the forces. The non bonded forces were evaluated each two steps, and each 10 time steps, a pair list containing the atoms separated by less than a distance of  $12 \text{ \AA}$  were generated. No margin was used for the patch length. A dielectric constant of 1.0 was used for the system which implies no modification of the electrostatic interactions, this value was set as the use of any larger value will lessen the electrostatic forces acting in the system<sup>44</sup>. An initial minimization over 10 ps was performed. A time step of 1 fs was used for integration over a total production time of 100 ps. The coordinates of the alkanes were recorded every 50 steps, and energy values were recorded every 100 steps.

The initial temperatures for each simulation were 10 K, 273 K, 288 K, 298 K, 308 K, 318 K, 328 K,

Bond Parameters				
	$K_b$	$b_0$		
	(kcal/mol/Å <sup>2</sup> )	(Å)		
$C_{CH3} - C_{CH2}$	222.500	1.5280		
$C_{CH2} - C_{CH2}$	222.500	1.5300		
$C_{CH3} - H$	322.000	1.1110		
$C_{CH2} - H$	309.000	1.1110		
Angle Parameters				
	$K_\theta$	$\theta_0$	$K_{(UB)}$	$S_{0(UB)}$
	(kcal/mol/rad <sup>2</sup> )	(degrees)	(kcal/mol/Å <sup>2</sup> )	(Å)
$C_{CH3} - C_{CH2} - C_{CH2}$	58.000	115.00	8.00	2.56100
$C_{CH2} - C_{CH2} - C_{CH2}$	58.350	113.60	11.16	2.56100
$H - C_{CH3} - C_{CH2}$	34.600	110.10	22.53	2.17900
$H - C_{CH2} - C_{CH2}$	26.500	110.10	22.53	2.17900
$H - C_{CH3} - H$	35.500	108.40	5.40	1.80200
$H - C_{CH2} - H$	35.500	109.00	5.40	1.80200
Dihedral Parameters				
	$K_\chi$	$n$	$\delta$	
	(kcal mol <sup>-1</sup> )	(unitless)	(degrees)	
$C_{CH3} - C_{CH2} - C_{CH2} - C_{CH3}$	0.1500	1	0.00	
$C_{CH3} - C_{CH2} - C_{CH2} - C_{CH2}$	0.1500	1	0.00	
$C_{CH2} - C_{CH2} - C_{CH2} - C_{CH2}$	0.1500	1	0.00	
$H - C_{CH3} - C_{CH2} - H$	0.1600	3	0.00	
$H - C_{CH2} - C_{CH2} - H$	0.1950	3	0.00	
Lennard-Jones Parameters				
	$\epsilon$	$r_{min}/2$	$\epsilon_{(1-4)}$	$r_{min}/2_{(1-4)}$
	(kcal mol <sup>-1</sup> )	(Å)	(kcal mol <sup>-1</sup> )	(Å)
$C_{CH3}$	0.080000	2.060000	0.010000	1.900000
$C_{CH2}$	0.055000	2.175000	0.010000	1.900000
$H$	0.022000	1.320000	–	–

Table 3.2: Force field parameters for alkanes. Note that  $C_{CH3}$  represents a carbon atom in a methyl group, while  $C_{CH2}$  represents a carbon atom in a methylene group.

338 K, 348 K, 358 K, 368 K, 378 K, 388 K, 398 K, 408 K, and 418 K. The n-alkanes used were those containing from four to eleven carbon atoms in their structure. The nanotubes used were the non-modified carbon nanotube and the modified carbon nanotubes A, B, and C, shown in Figure 3.2. For the modified atoms, the  $\epsilon$  values used were 0.3 kcal mol<sup>-1</sup>, 0.9 kcal mol<sup>-1</sup>, 1.2 kcal mol<sup>-1</sup>, 1.5 kcal mol<sup>-1</sup>, 3.0 kcal mol<sup>-1</sup>, 10.0 kcal mol<sup>-1</sup>, 30.0 kcal mol<sup>-1</sup>, and 90.0 kcal mol<sup>-1</sup>.

### 3.4 NVT simulation details

Molecular Dynamics were performed using NAMD 2.13 in Windows64, the canonical ensemble (NVT) was used by means of the addition of the Langevin thermostat that consist in the addition of a friction force and a random force that will simulate the damping of particles due to friction<sup>71</sup>. The Langevin stochastic thermostat was used with a damping coefficient of 5 ps<sup>-1</sup>. Each simulation had the aim to follow the conformational changes for n-alkane molecules diffusing through a nanotube and to obtain their self-diffusion coefficient. Carbon nanotubes were fixed in space, allowing only the motion of alkane molecules. Periodic Boundary conditions were also established in order to simulate an infinite nanotube.

The periodic box had a length of 20 Å in the  $x$ -direction, a length of 20 Å in the  $y$ -direction, and a length of 100.695 Å in the  $z$ -direction, all of them centered in the coordinates (0.0, 0.0, 49.735). A spherical cutoff of 12 Å was used for the potentials, with a smoothing function that started from a radius of 10 Å. Each step the non bonded interactions were computed. Each 10 time steps, a pair list containing the atoms separated by less than a distance of 15 Å was generated. A margin of 100 Å was used to define the patch length. A dielectric constant of 1.0 was used for the system. All bonds were set as rigid. Particle Mesh Ewald (PME) was activated for long-range Coulombic interactions as it is required to be defined by NAMD every time PBC are established. Initial energy minimization was performed over 100 ps. After that, the initial velocity was established as 298 K and maintained by the thermostat. A time step of 1 fs was used for integration over a total production time of 4 ns. The coordinates of the alkanes, as well as the energy

values, were recorded every 500 steps.

First, simulations were carried out with only one alkane molecule within the A, B, and C modified carbon nanotubes with an  $\varepsilon$  value of  $2.0 \text{ kcal mol}^{-1}$  for the modified atoms. Then, some other molecules of the same alkane were included into the nanotube and Molecular Dynamics under the same conditions were performed. Second, Molecular Dynamics were performed for alkane molecules within the modified nanotube type D, with  $\varepsilon$  values of  $0.05 \text{ kcal mol}^{-1}$ ,  $0.07 \text{ kcal mol}^{-1}$ ,  $0.10 \text{ kcal mol}^{-1}$ ,  $0.13 \text{ kcal mol}^{-1}$ ,  $0.15 \text{ kcal mol}^{-1}$ ,  $0.17 \text{ kcal mol}^{-1}$ , and  $0.20 \text{ kcal mol}^{-1}$  for the modified atoms. For all these simulations, the linear alkanes used were those containing from four to eleven carbon atoms, and eicosane which has twenty carbon atoms in its backbone chain.

## 3.5 Characterization of alkane conformations during the trajectory

### 3.5.1 Torsion angles $[\phi, \psi]$ plot

The dihedral angles for each alkane were calculated from the coordinates of each carbon atom at each recorded time step. As butane contains only four carbon atoms, just one dihedral angle was obtained. From pentane, two dihedral angles were obtained. From hexane were obtained three dihedral angles, and thus, as the chain length of the alkane increases the number of dihedral angles does the same. Based on the Ramachandran plots<sup>72</sup>, which plot data sets for two continuous dihedral angles ( $\phi$  and  $\psi$ ) that rotates around the alpha carbon in each internal amino acid forming a protein<sup>73</sup>, a similar plot for the dihedral angles of each alkane was used. This  $[\phi, \psi]$  plot allowed to visually summarise the most energetically favorable regions for dihedral angles in linear alkanes.

Each pair of torsion angles  $\phi$  and  $\psi$  are described by three continuous planes formed between five adjacent carbon atoms, where each plane shares an edge with the next one, such planes are illustrated in blue, red, and violet in Figures 3.3.a and 3.3.b. The red plane serves as the reference in measuring both angles. From the five carbon atoms involved in each pair,  $\phi$  is described by the first four of them,

meanwhile,  $\psi$  relies on the last four. Thus,  $\phi$  is the torsion angle around the second and third carbons, consequently,  $\psi$  defines the rotation around the third and fourth carbons, see Figure 3.3.a. If enough carbon atoms are present in the alkane, the next pair of  $(\phi, \psi)$  angles will be displaced by one carbon atom as shown in Figure 3.3.b. Using the same criteria, all the possible pairs of angles were calculated using Python 3.7. For example, the totality of possible torsion angles in an alkane molecule is shown in Figure 3.3.c. The number of possible  $(\phi, \psi)$  pairs can be obtained by subtracting four to the length of the carbons chain of the alkane.

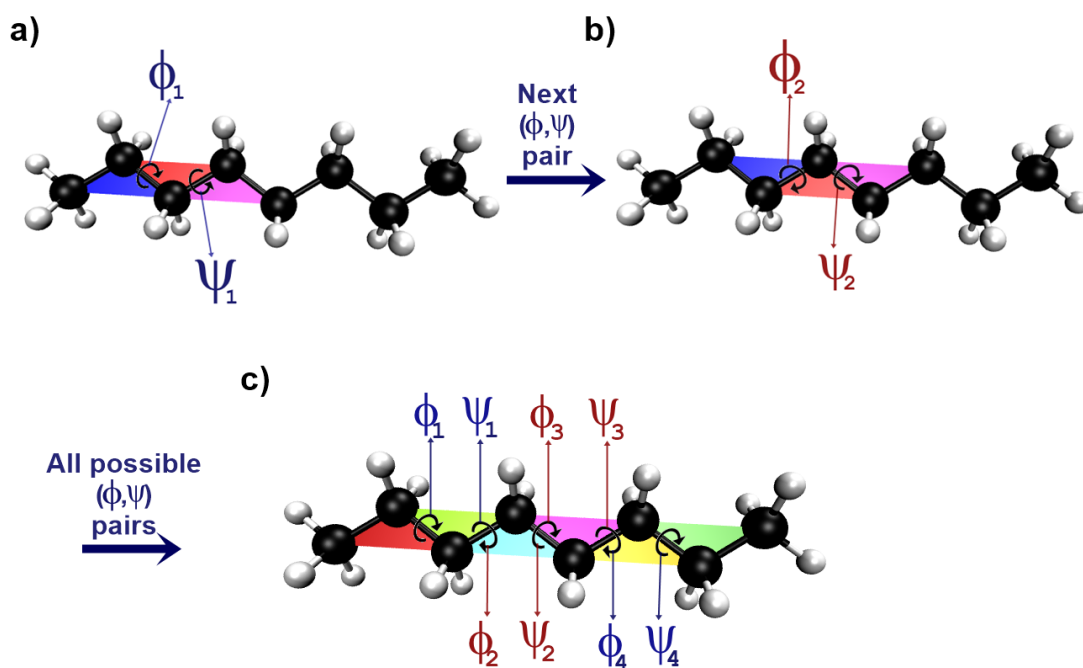


Figure 3.3: Torsion angles  $\phi$  and  $\psi$  describing an alkane molecule. a) First pair of torsion angles for octane.  $\phi_1$  represents the rotation around the second and third carbons, and  $\psi_1$  represents the rotation around the third and fourth carbons. b) Second pair of torsion angles for octane.  $\phi_2$  represents the rotation around the third and fourth carbons, and  $\psi_2$  represents the rotation around the fourth and fifth carbons. c) All possible pairs of torsion angles describing an octane. Note that for an octane molecule four  $(\phi, \psi)$  pairs can be obtained

From all  $(\phi, \psi)$  pairs obtained during a MD simulation, the most favorable zones were estimated by using a Two-Dimensional Kernel Density Estimation in R version 3.6.3. The plot was divided into three zones, the first cutoff was set as 99.5% of probability mass to enter in the allowed region and the second cutoff was set as 95.0% of probability mass to enter in the favoured region, with an error of  $\pm 0.05\%$  for the former and  $\pm 0.1\%$  for the later. A sample density estimation is shown in Figure 3.4.

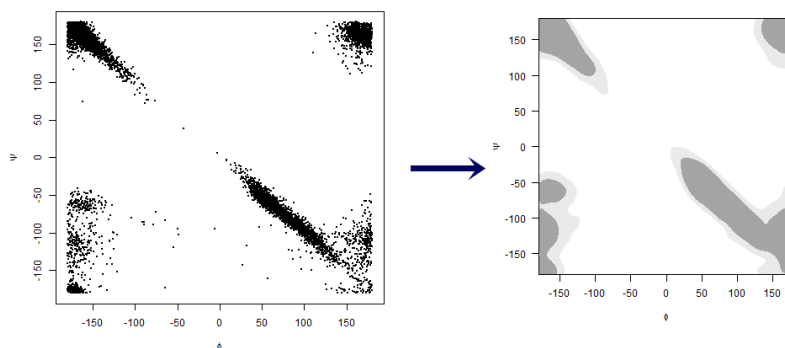


Figure 3.4: On the left: Scatter plot for all  $(\phi, \psi)$  pairs obtained during a MD simulation. On the right: The equivalent density plot obtained by the Two-Dimensional Kernel Density Estimation in R.

### 3.5.2 End-to-end distance

The end-to-end distance was obtained at each recorded time by computing the Euclidian distance between the coordinates of the centers of the initial and final carbon atoms forming each alkane. For those simulations with more than one alkane diffusing at the same time, their arithmetic mean was used. The instant folding degree was computed by the use of Equation 3.1, where  $d_0$  is the end-to-end distance at the equilibrium conformation (where all the dihedral angles are  $180^\circ$ ) for the alkane, and  $d_t$  is the mean end-to-end distance for the molecules at the time  $t$ . If the carbon-carbon bond lengths are set rigid,  $d_0$  is the maximum distance reached and the values will range between 0 and 100. Even though, as two atoms

can not occupy the same space, the folding degree will never be 100%.

$$F_d = \frac{(d_0 - d_t)}{d_0} \times 100. \quad (3.1)$$

Figure 3.5.a shows an octane molecule with all its dihedral angles at  $180^\circ$ , which folding degree is equal to 0%. Figure 3.5.b shows an octane molecule in which the end-to-end distance is equal to half  $d_0$ , in that case, its folding degree is equal to 50%. Figure 3.5.c shows the impossible case where the initial and final carbon atoms of an octane molecule occupy the same space, in that case, the folding degree would be 100%.

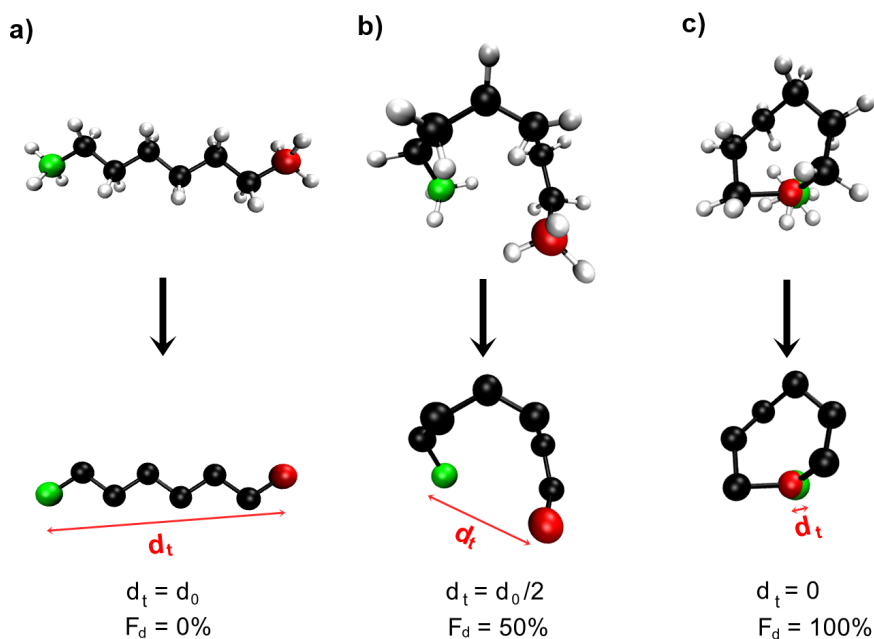


Figure 3.5: End-to-end distances for alkanes. a) Octane most stable conformation. b) Octane with 50% folding degree. c) Hypothetical octane with 100% folding degree. Top: Perspective view of octane conformations including hydrogen atoms. Bottom: view without hydrogen atoms, their end-to-end distance  $d_t$  is illustrated as the red double arrow that goes from the center of the initial and final carbon atoms, drawn in green and red, respectively.

### 3.6 Determination of alkane diffusion coefficients

The sequence of centers of mass through the observed time from MD simulations allowed the calculation of the mean square displacement (MSD) as a function of time. The MSD was computed only in the direction of the pore, which means, in the  $z$ -direction. The Einstein-Smoluchowski relationship for one-dimensional diffusion along  $z$ , made possible to obtain a macroscopic quantity like the diffusion coefficient,  $D_z$ , from a microscopic quantity, the mean square displacement of the molecules<sup>74</sup>. Thus, MSD was obtained from the  $z$ -positions of the center of mass for each alkane molecule over time. Even though, mean square displacement can be computed using any time step in the simulation as the time origin, the same was computed as a function of the lag time,  $\eta$ , which is the time from the beginning of the production time at which starts a smaller window of data, called  $\omega$ . As a result, it was possible to obtain equally weighted data sets for MSD, see Equation 3.2.

$$MSD(\eta) = \frac{1}{\omega_{to} - \omega_{from} - \tau} \sum_{t_0=\omega_{from}}^{\omega_{to}-\tau-1} \frac{1}{N} \sum_{i=1}^N (z_i(t_0 + \tau) - z_i(t_0))^2 ;$$

*with*  $\eta = n\tau$  for  $n = 0, 1, 2, \dots, P/2\tau$ .

(3.2)

Where  $\omega_{from}$  is the initial step of the window,  $\omega_{to}$  is the final step of the window,  $z_i(x)$  represents the  $z$ -component of the spatial position of the  $i^{th}$  molecule at time  $x$ ,  $t_0$  is the time origin,  $\tau$  is the lag time between two positions (the time after  $t_0$  at which the spatial positions will be compared against the position at  $t_0$ ), and  $N$  is the number of alkane molecules involved. Note that  $\tau$  was kept constant during the entire production time, but it indicated how  $\eta$  incremented. Thus, at each lag time  $\eta$ , the initial step and the final step of the window were increased by a  $\tau$  time from the last window, therefore, resulting in the increment of the values of the time origins used. The window length was half the production time,  $P$ , and as  $\eta$  values were chosen those minor to half the production time, then, good statistics for the results were expected as the MSD for each  $\eta$  had the same number of spatial position points contributing to them, as shown in



Figure 3.6.

Then, a linear least squares regression was performed from the MSD values evaluated at continuous  $\eta$  values for time. From the Einstein-Smoluchowski relationship, Equation 3.3 is used to describe the equation from the linear regression. Note that the added term to the original Einstein equation,  $b_0$ , responds to the shift in the curve caused by the initial steps of the simulation, but have no physical meaning beyond. Then, the value for the self-diffusion coefficient along  $z$  was obtained from the slope of Equation 3.3. The error associated with this self-diffusion coefficient was obtained from the standard error of the regression slope.

$$MSD(t) = (2D_z)t + b_0 \quad (3.3)$$

As MSD needed long time simulations to show an asymptotic and linear behavior over time, the first  $n$  steps were discarded to ensure equilibration of the system. For NVE simulations, the first 20 ps from production time were discarded as equilibration time,  $\tau$  was set as 4 ps, and  $\eta$  values were set from 0 to 40 ps, as a result, 11 different  $\eta$  points were used. Similarly, for NVT simulations, the first 200 ps from production time were discarded,  $\tau$  was set as 190 ps, and  $\eta$  values were set from 0 to 1900 ps, again, 11 different  $\eta$  points were used.

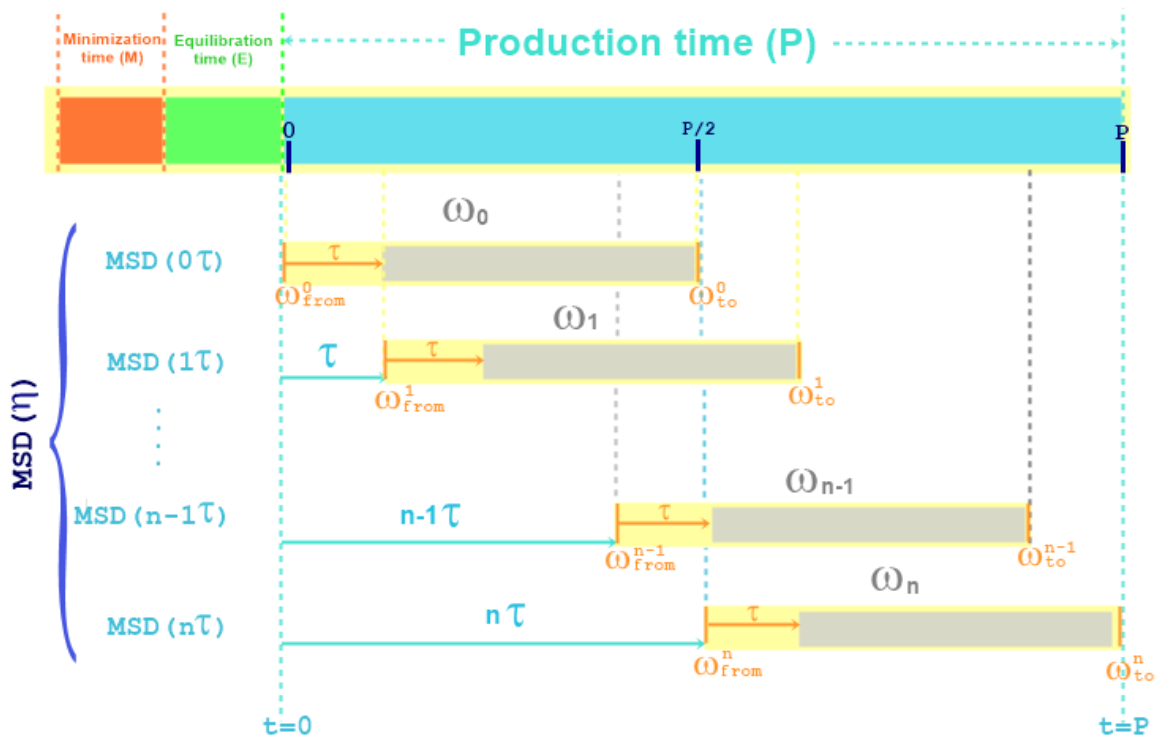


Figure 3.6: Schematic representation of MSD calculation over production time. Production time  $P$  is divided into smaller windows  $\omega_n$  of length  $P/2$ , each window goes from  $\omega_{from}$  to  $\omega_{to}$ . The time used as a reference for MSD at each window is  $\eta$ , which is the time difference between the start of the window and the start of the production time. Note that such a difference is always a multiple of  $\tau$ , being  $\tau$  the lag time for each molecular displacement used to compute MSD inside the window. The first calculated displacement at each window is shown as the orange arrow that covers the yellow zone, besides, the gray zone would be progressively covered by calculations of the square displacement between positions separated by  $\tau$  until reach  $\omega_{to}$ .

## Chapter 4

# Results & Discussion

This section presents the results obtained along with their discussion in each subsection. In such a way that will be discussed, first, the effect of increasing the chain length of the alkanes, then, the effect of the simulation temperature, followed by the favorable conformations that undergo the alkanes during the diffusion process, then will be investigated the effect that the van der Waals potentials from the carbon nanotube modified atoms exerts on the attraction between the alkane molecules and the nanotube, and finally, the obtained diffusion coefficients for the alkanes will be presented.

### 4.1 Effect of chain length

The translational distance for each recorded time was obtained in order to quantify the total distance traveled by the alkane molecules inside a carbon a nanotube. Figure 4.1 exhibits the accumulative translational distances for simulations under the microcanonical conditions detailed in the methodology. The accumulative distance does not strictly follow a decreasing trend as the chain length increases, as can be seen in the case of pentane and heptane with an initial temperature equal to 10 K. Even though, this decreasing trend is remarkably stable at higher initial temperatures.

As the total simulation time was about 100 ps, the traveled accumulative distances were around 100 Å for the case with an initial temperature equal to 10 K and this value increased as the initial temperature increased as well. Then, for an increase of six carbon atoms (comparing butane with decane), there is a reduction of 27.26% in the accumulative translational distance. At 298 K the reduction of the accumulative distance was around the 38.35% as the number of carbon atoms in the alkane increased from four to eleven. Similarly, the reduction between butane and decane at 318 K was around the 30.37%, and finally, at 398 K there is a reduction of the 58.96% between butane and undecane.

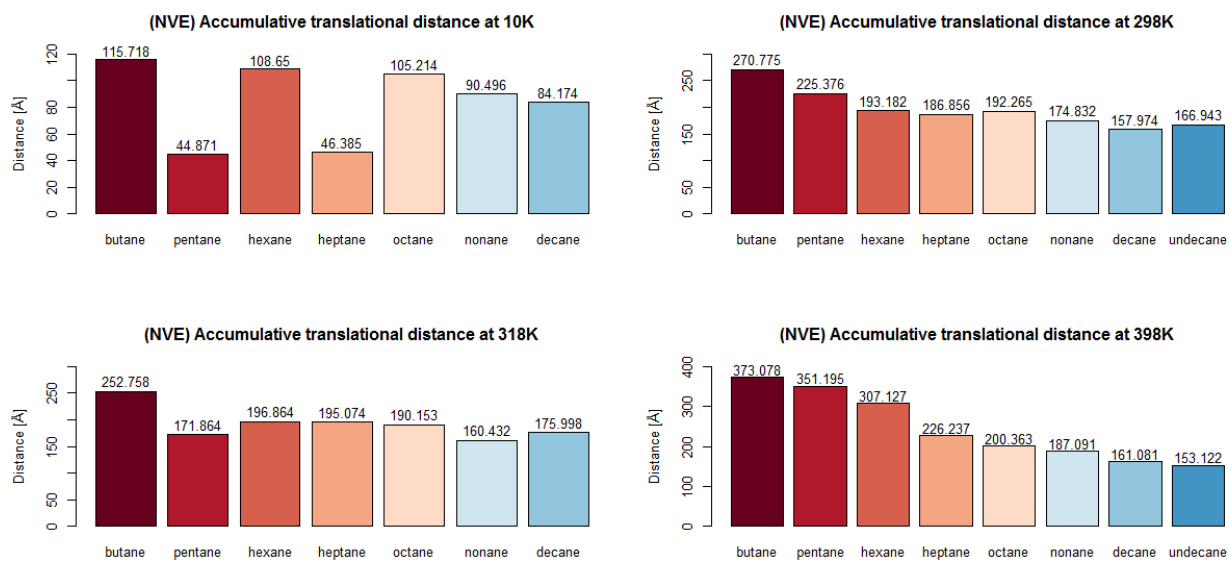


Figure 4.1: Accumulative translational distance for microcanonical simulations of single alkane molecules inside a non modified (NM) CNT with initial temperatures equal to 10 K, 298 K, 318 K and 398 K, respectively.

For single-molecule simulations under canonical conditions, as the total simulation time was 4 ns, the accumulative translational distances were above 5000 Å, as shown in Figure 4.2, where the tendency to decrease the travelled distance as the carbon chain length increases did not show any irregularity. Then, as the number of carbon atoms forming an alkane increase, the difference in the accumulative translational

distance with the previous alkane reduces, in such a way that the reduction between butane and pentane is about the 10%, but the reduction between decane and undecane is only about the 3.9%.

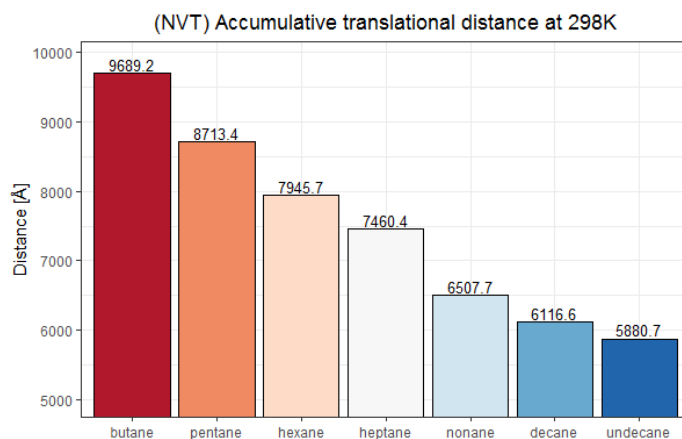


Figure 4.2: Accumulative translational distance for canonical simulations of single alkane molecules inside a non modified (NM) CNT at 298 K.

In order to know about the influence of the number of molecules involved in the diffusion process, the same canonical conditions were established for simulations with a larger number of alkane molecules inside the carbon nanotube. The accumulative translational distance was obtained as the sum of the average distances traveled by all the alkane molecules in the system at each recorded step and can be seen in Figure 4.3, where the number density for the alkane molecules in each simulation is shown as the red line. In the case of butane, there is a reduction of 21,9% in the accumulative translational distance when the number density increases from  $1.63 \times 10^{-4} \text{Å}^{-3}$  to  $1.79 \times 10^{-3} \text{Å}^{-3}$ , or equivalently, when the simulation became multi-molecule. Similarly, undecane simulations registered a reduction over the traveled distance of around 22.5% when more alkane molecules were added inside the CNT.

Now, between multi-molecule simulations, pentane total traveled distance got reduced by a 10.2% compared with butane, besides, undecane registered a reduction of the 38.5% compared with butane (see Figure 4.3). For eicosane, being the case with the greater increment in the number of carbon atoms

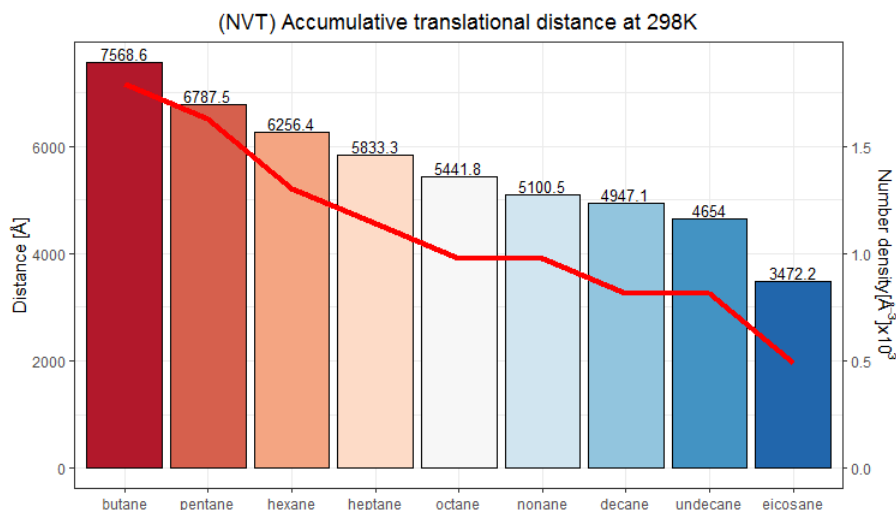


Figure 4.3: Accumulative translational distance for canonical simulations of alkane molecules inside a non modified (NM) CNT, performed at 298 K, and their corresponding number density. The bar plot shows the sum of the average translational distance traveled at each recorded time, expressed in Å. The red line shows the number density of the alkanes at each simulation, expressed in Å<sup>-3</sup>.

among the chains studied, the accumulative distance was reduced by more than a half compared with butane, specifically the 54%, as the number of carbons increased by sixteen atoms. Then, as the n-alkane backbone grows in length, there is a minor contribution of such growth to the reduction of the accumulative distances traveled.

When the simulations were performed using the modified carbon nanotubes and microcanonical conditions, the effect of the chain length was higher for those with modification type A, resulting in a reduction of 69.9% for undecane total translational distance from butane. Besides, the gap between translational distances got reduced with the type C modification due to the reduction of the translational distance obtained for butane, see Figure 4.4 (left). Under canonical conditions, the nonmodified cases yielded to the higher translational distances and there was not observed much variability between the different CNT modification types. However, between these little changes, the B-type modification led to

the higher translational distance for shorter n-alkanes while D-type modification led to the higher for the larger n-alkanes. The reduction in the accumulative translational distance between butane and eicosane was around 53% for all the nanotube types, as shown in Figure 4.4 (right).

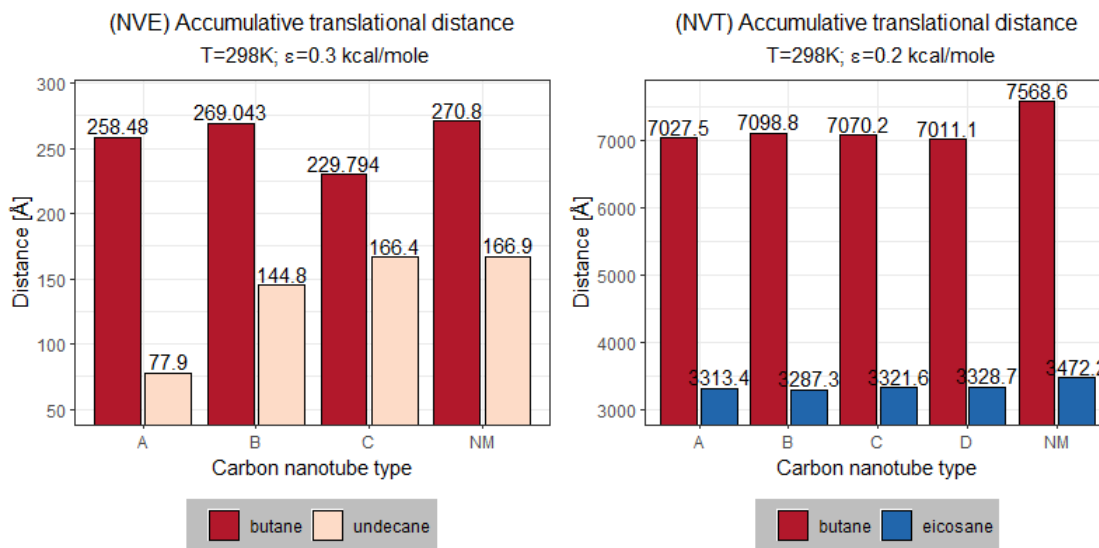


Figure 4.4: On the left: Accumulative translational distance for microcanonical simulations with initial temperature 298 K and  $\epsilon$  equal to  $0.3\text{ kcal mol}^{-1}$  for various modified CNT types. On the right: Accumulative translational distance for canonical simulations at 298 K and  $\epsilon$  equal to  $0.2\text{ kcal mol}^{-1}$  for various modified CNT types.

## 4.2 Effect of Temperature

To evaluate the effect of temperature, the accumulative translational distance was compared for different simulations performed under the same conditions but only varying temperature. Figure 4.5 displays a set of accumulative translational distances for different alkanes at three temperatures: 10 K, 273 K and 398 K. As can be observed, the increment in the total distance is proportional to the increment of temperature for all the cases shown. At higher temperatures there is a correlation between the translational distance and the alkane length, on the contrary at lower temperatures there is no such direct correlation, in fact

the translational distance depends on the alkane. Because of the inconsistency showed by pentane and heptane at lower temperatures, while studying the carbon chain length effect on the accumulative distances traveled, their temperature effect was examined in depth.

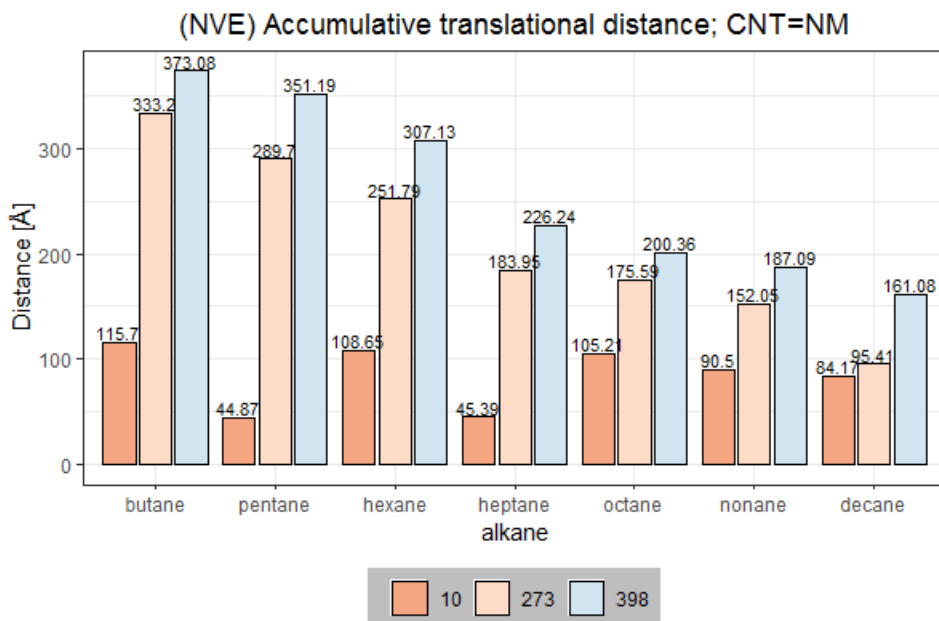


Figure 4.5: Accumulative translational distance under microcanonical conditions for alkanes inside non-modified carbon nanotubes at three different temperatures.

Figure 4.6 displays the accumulative translational distance for a heptane molecule inside a nonmodified carbon nanotube. There is a small increment for the first simulations above 273 K, but a large increment as it goes near to 378 K, which resulted to be the maximum for the total distance traveled over the range studied. The effect of temperature on diffusion in carbon nanotubes was characterized in a previous work by Jakobtorweihen et al<sup>75</sup>, in which it was shown that the self-diffusion usually increases with temperature, for the rest of cases where the self diffusivity decreases it gives rise to a small maximum in the temperature dependence. Such effect was called the levitation effect, being caused by an increment on fluid-wall collisions that reduces the self-diffusion coefficient. This increment over alkane-wall collisions



is displayed as a 2D density plot for heptane at 378 K (left) and 398 K (right) in Figure 4.7, where can be observed how at higher temperatures the heptane molecules are distributed closer to the CNT walls over the recorded time. For some other studied alkanes, the temperature of this peak varies in the range between 338 K and 398 K.

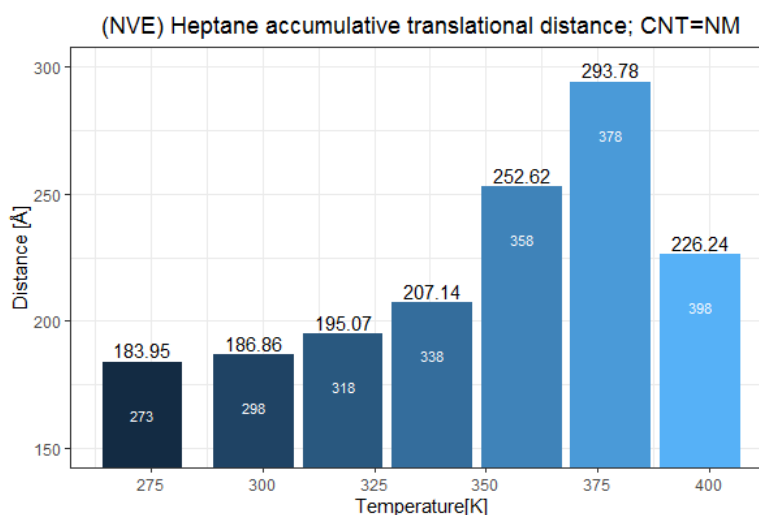


Figure 4.6: Accumulative translational distance for heptane under microcanonical conditions in a non-modified carbon nanotube for various initial temperatures.

### 4.3 Most favorable conformations

Alkanes in solution prefer to be in extended conformations with all their dihedral angles in trans conformations. Even though, when they are confined into small cavities, such as nanotubes, the conformations exhibited may differ from the expected. For the model studied, in which alkanes are confined into narrow nanotubes, a new redistribution of these trans conformations is expected. Some two-dimensional contour plots for the density probability of continuous dihedral angles, called  $[\phi, \psi]$  plots, obtained under microcanonical conditions are displayed in Figure 4.8.

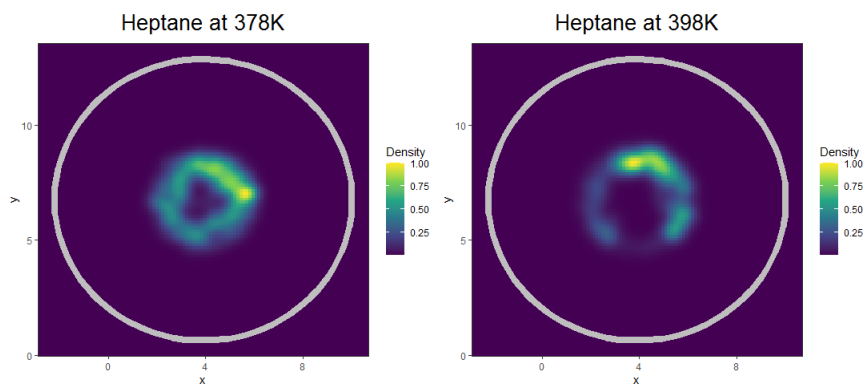


Figure 4.7: Obtained 2D probability density for heptane center of mass at 378 K and 398 K ( $z$ -axis view).

With 298 K as the initial temperature for the alkanes containing from four to ten carbon atoms, under microcanonical conditions, the majority of their dihedral angles are found to be in the trans region. Moreover, the dihedral angles for pentane showed the higher deviation from the trans-trans pairs, ideally  $\pm 180^\circ$  for both  $\phi$  and  $\psi$  angles, which is evidenced as a greater area surrounding the corners of its density  $[\phi, \psi]$  plot compared with the others. Conversely, hexane  $[\phi, \psi]$  pairs showed the minor deviation from the trans-trans pair as the area around this zone is the smallest. Surprisingly, undecane showed gauche conformations in addition to the former only trans-trans pairs registered for the rest of alkanes under the same conditions, see Figure 4.9.

For undecane, the density  $[\phi, \psi]$  plot revealed the addition of gauche' conformations, which are formed by dihedral angles around  $-60^\circ$ . As a result, gauche'-trans conformational pairs were added in the density plots. Further details are shown in the individual  $[\phi, \psi]$  plots, which showed that the gauche' conformations were raised around the bond formed between the third and fourth carbon atoms as well as the bond formed between the seventh and eighth carbon atoms.

The vast majority of the simulations performed under the same microcanonical conditions with CNTs modifications A, B, and C, showed similar conformations as those presented in Figure 4.8, meaning that in most of the cases there were no deviations from the trans-trans pair regardless of temperature or van der

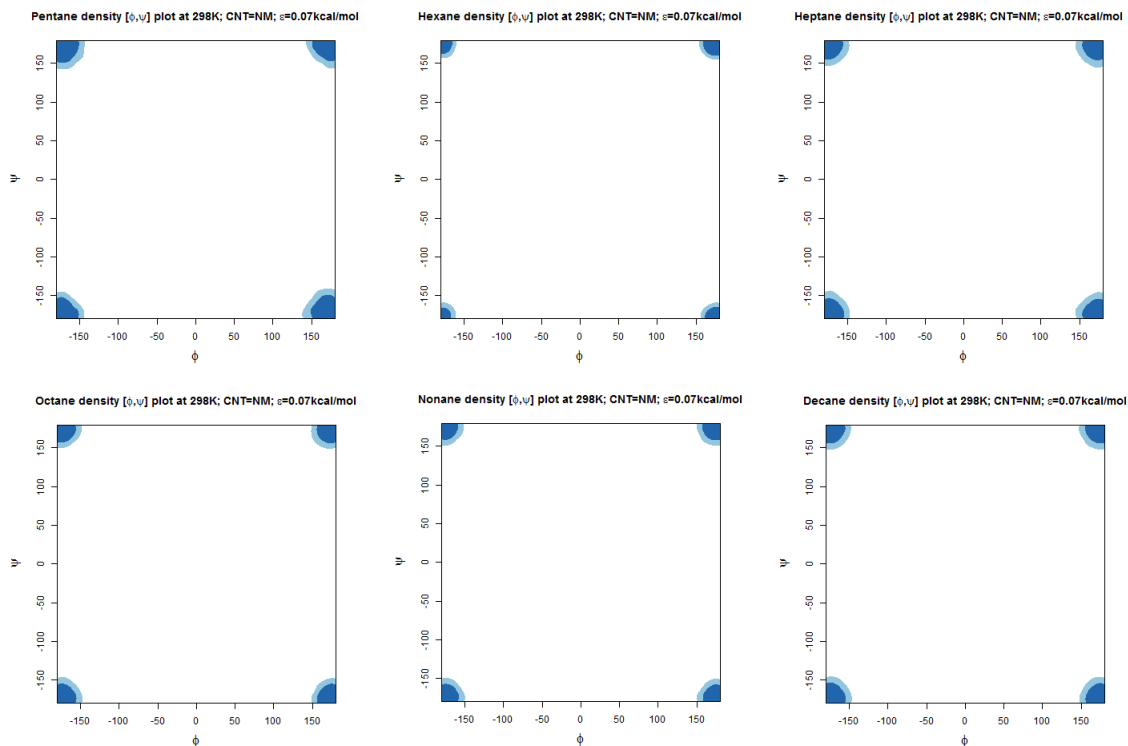


Figure 4.8: Density  $[\phi, \psi]$  plots for single alkane molecules inside nonmodified carbon nanotubes under microcanonical conditions with an initial temperature equal to 298 K. From left to right: In the first row, pentane, hexane, and heptane density  $[\phi, \psi]$  plots. In the second row, octane, nonane and decane density  $[\phi, \psi]$  plots.

Waals interactions exerted by the modified atoms with high  $\varepsilon$  values at short simulation times.

Because of the experienced conformational variability, the end-to-end distance was affected, moreover experimental studies showed that in average each gauche conformation reduces in  $0.32 \text{ \AA}$  the length of the n-alkane while this is encapsulated into a cavity (like nanotubes) or inside capsular self-assembly of cavitands<sup>76</sup>. Therefore, the folding degree for those n-alkanes with gauche conformations in their backbone resulted to be higher than zero, as in the example shown in Figure 4.10. Even though the mean folding degree is under 10%, there are maximum peaks over 20% that will represent the conformation of the alkane once all the registered gauche' conformations took place at the same time. For the simulations

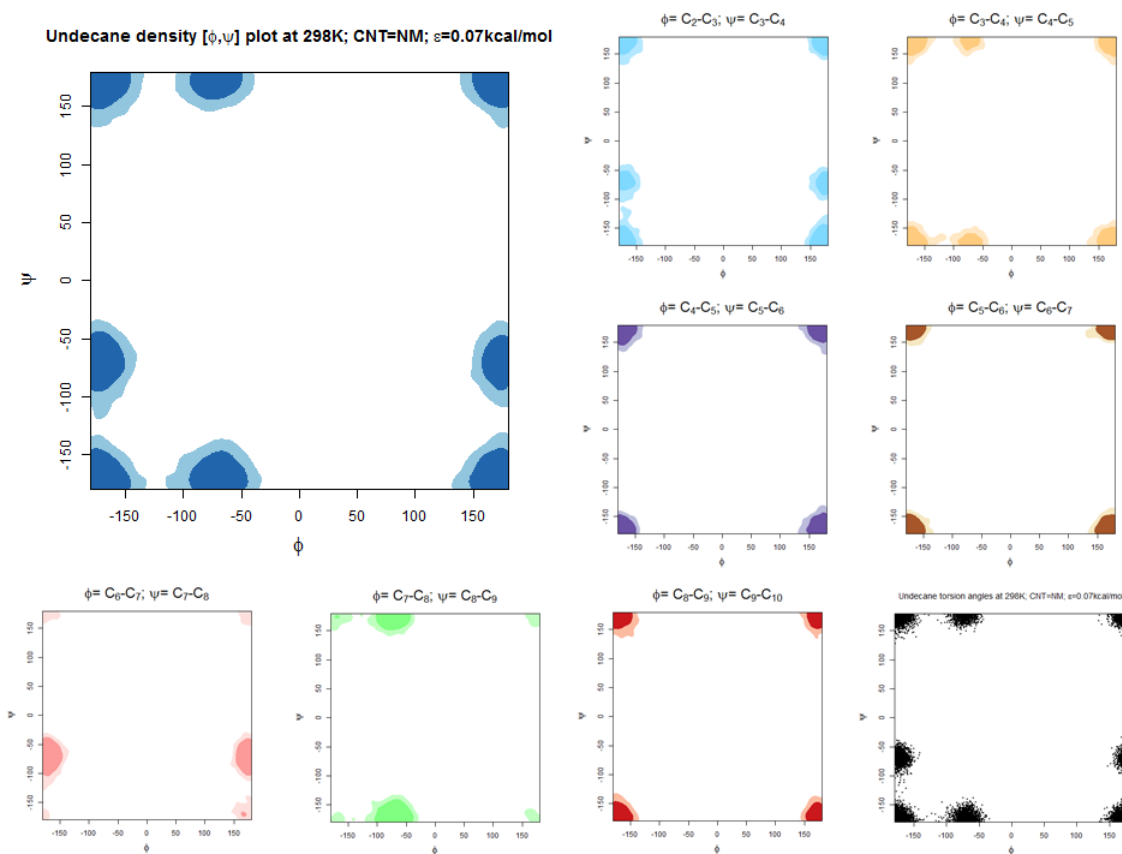


Figure 4.9: Density  $[\phi, \psi]$  plots for undecane in non modified carbon nanotube with an initial temperature equal to 298 K. The top left-hand corner plot (blue) corresponds to the resume plot for all the  $[\phi, \psi]$  pairs. The other density  $[\phi, \psi]$  plots correspond to an specific pair. Each of them indicates the bond that defines the rotation for the consecutive  $\phi$  and  $\psi$  dihedrals, which are constituted from the first to fifth (light blue), second to sixth (orange), third to seventh (violet), fourth to eighth (brown), fifth to ninth (rose), sixth to tenth (green), and seventh to eleventh (red) carbon atoms in the backbone of the alkane. The bottom right-hand corner scatter plot (black) corresponds to the resume plot for all the  $[\phi, \psi]$  pairs obtained.

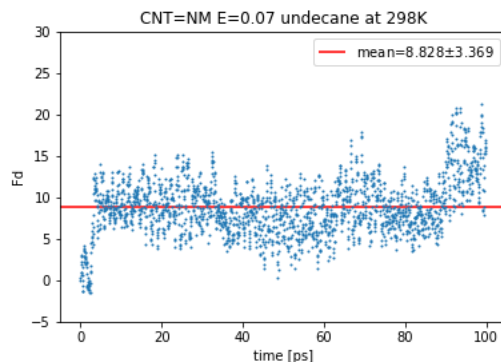


Figure 4.10: Folding degree for undecane in a nonmodified carbon nanotube under microcanonical conditions with an initial temperature equal to 298 K.

that only showed trans-trans conformational pairs, the mean folding degree rounded 0%.

Previous studies have shown that in simulations of alkane transport within nanotubes the backbone chains adsorb onto the interior nanotube walls resulting in the alkanes undergoing more curved conformations as the attractive interactions with the wall of the nanotube tend to maximize<sup>77</sup>. As shown in Figure 4.11, for longer time periods, the conformer populations inside the nanotube are roughly stable resulting in the increase of gauche conformation population over the trans predominant conformations for non confined alkane molecules. As can be observed, the energetically unfavored gauche-gauche, gauche'-gauche' and gauche-gauche' pair conformations are now present in all the alkanes studied. For shorter alkanes there were obtained more favored  $\phi$  and  $\psi$  pairs, than for the larger ones, which can be observed in the smaller gauche regions accomplished by nonane, decane, undecane, and eicosane.

The confinement imposed by simulations performed with the presence of modified atoms induced conformational changes from the mainly trans conformer in bulk solution registered for alkanes. Again, all possible pairs between gauche, gauche', and trans conformers appeared. As the alkane chain length increased, the consecutive dihedral angles continued to exhibit gauche conformations in the alkane backbone but showed less probability of forming a gauche-gauche' pair. Note that the allowed area for these

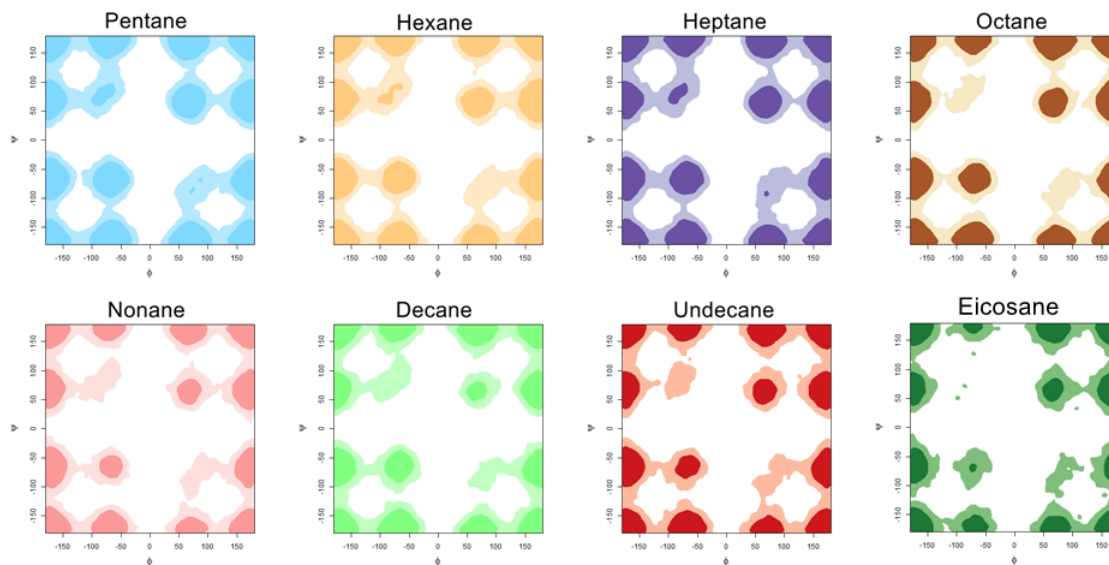


Figure 4.11: Density  $[\phi, \psi]$  plots for alkane molecules in a nonmodified carbon nanotube under canonical conditions ( $T = 298 \text{ K}$ ,  $\varepsilon = 0.07 \text{ kcal mol}^{-1}$ ).

simulations with modified CNTs (Figure 4.12) is smaller than for those of non modified CNTs (Figure 4.11). As a result the increase of  $\varepsilon$  resulted in a reduction of the range of dihedral angles filled by n-alkanes.

The rearrangement of dihedral angles in n-alkanes affected their end-to-end distance, as a result, their global folding degree increased as a consequence of more gauche conformations. As shown in the density  $[\phi, \psi]$  plots from Figure 4.12, short alkanes undergo gauche conformations in a wider range of angles than the larger ones. For instance, under C modification, pentane experienced a mean folding degree of  $(6.136 \pm 1.919)\%$  which is accountable for the deviation experienced from trans conformers. As the backbone increased in length, e.g. nonane, the favored regions in the gauche zone decreased resulting in lower end-to-end distances. Even though, can be appreciated some peaks for its folding degree that almost reach the 20% that are responsible for a mean folding degree of  $(5.812 \pm 2.943)\%$ , which even being small than the one for pentane showed much more variation, see Figure 4.13. As the alkane chain grew in size the mean folding degree tends to decrease, e.g. eicosane showed a value of  $(3.71 \pm 1.56)\%$  based on the

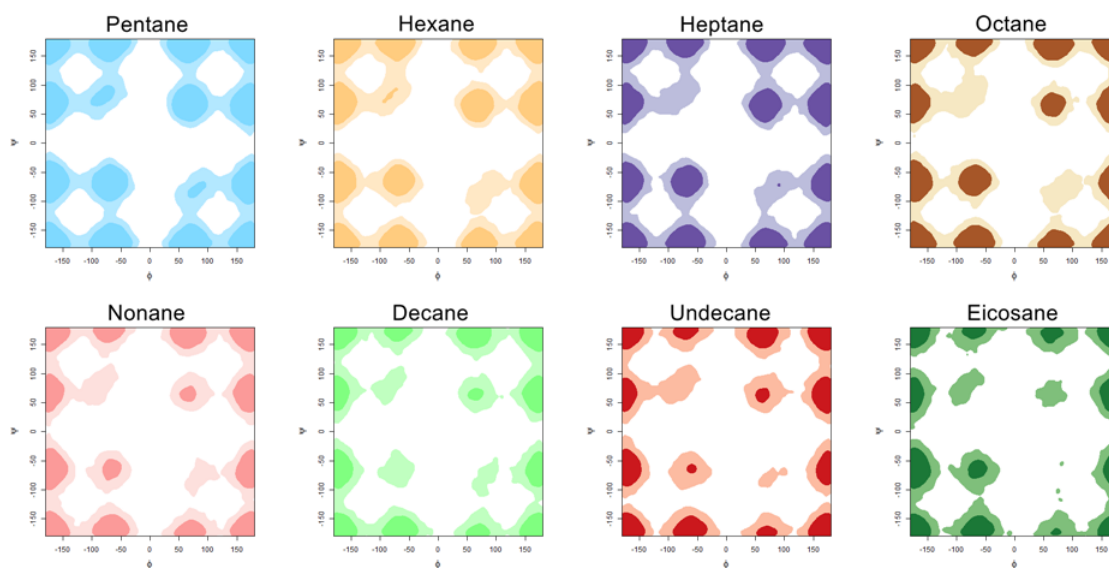


Figure 4.12: Density  $[\phi, \psi]$  plots for alkane molecules in type C modified carbon nanotube under canonical conditions ( $T = 298$  K,  $\varepsilon = 0.20$  kcal mol $^{-1}$ ).

conformers caused by both of its chain ends. As the center of the alkane did not show favored gauche regions there was no rough variation such as the one experienced for shorter n-alkanes and the recorded end-to-end distances yielded to instant folding degrees below 10%.

Even in those cases where may be inferred that exist a global single handedness helical conformation because of the preference over the gauche'-gauche' pairs (both around  $-60^\circ$ ) or gauche-gauche pairs (both around  $+60^\circ$ ), the lack of a chiral center for the n-alkanes will give rise to both left and right-handed helices that will be observed under longer periods of time<sup>78</sup>. Then, for instance, the displayed simulation results for eicosane molecules in Figure 4.11 presented a higher probability to be found in gauche-gauche pairs, conversely, the obtained results for eicosane in Figure 4.12 showed a preference over gauche'-gauche' pairs, but that is just a kinetic result over the 4 ns used for the simulations.

To summarize, for simulations using the proposed modified carbon nanotubes (A, B, and C) the overall populations of dihedral pairs get reduced in probability density for the gauche-gauche, gauche'-

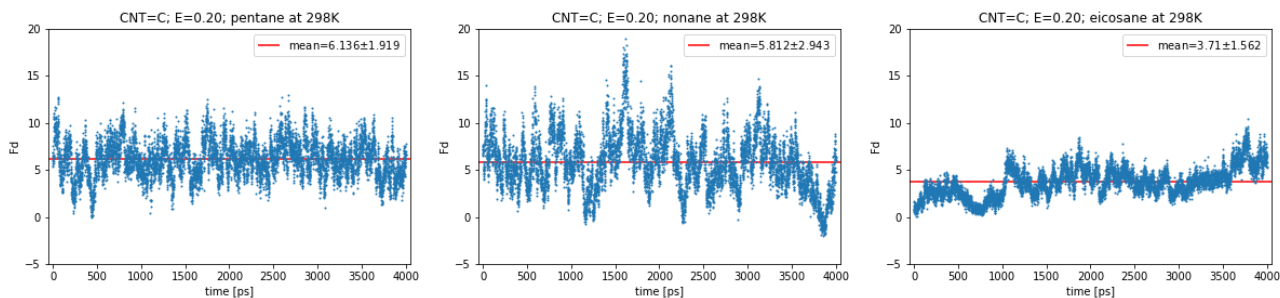


Figure 4.13: From left to right: Folding degree for pentane, nonane, and eicosane in C-type nanotube, under canonical conditions ( $T = 298 \text{ K}$ ,  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$ ).

gauche', and gauche-gauche' conformational pairs. The small probability density showed in the  $[\phi, \psi]$  plots for these conformations is mainly due to the ends of the alkanes, meanwhile, the main conformations experienced in the middle of the alkane corresponded to the trans-trans pairs. Also, the highly improbable and sterically conflicting gauche-gauche' conformations get reduced in comparison with the non-modified carbon nanotube.

#### 4.4 Location distribution of alkanes

In nonmodified CNTs the obtained distance of separation between an alkane molecule and its closest atom from the CNT wall is almost the same at each time step because of the adsorption of the alkanes on the walls, which is directly related to the intermolecular equilibrium distance. Giving rise to a circular-like shape inside the nanotube when plotting the probability density of the centers of mass overall the simulation from a  $z$ -axis view, as seen in Figure 4.14. Also, it can be observed how the center of mass of alkanes is never located in the center of the CNT. The distance from the CNT wall to the most probable area (in direction towards the center of the CNT) is directly related to the vdW equilibrium distance,  $r_{min}$ , the same that is described in the Lennard-Jones parameters in the force field used in the simulations.

As can be seen in Table 3.1, only the  $\varepsilon$  value from Lennard-Jones parameters of the modified atoms in



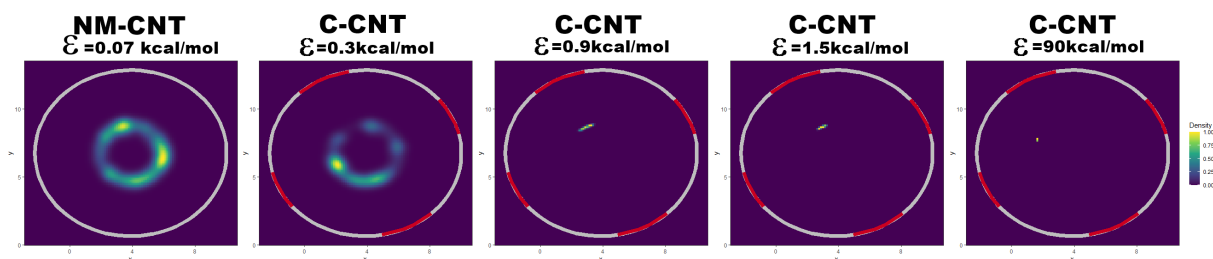


Figure 4.14:  $z$ -axis view of density plots for center of mass positions through simulation time for undecane under microcanonical conditions with an initial temperature equal to 298 K. From left to right: Non modified CNT with  $\varepsilon = 0.07 \text{ kcal mol}^{-1}$ , type C CNT with  $\varepsilon = 0.3 \text{ kcal mol}^{-1}$ , type C CNT with  $\varepsilon = 0.9 \text{ kcal mol}^{-1}$ , type C CNT with  $\varepsilon = 1.5 \text{ kcal mol}^{-1}$ , and type C CNT with  $\varepsilon = 90 \text{ kcal mol}^{-1}$ .

carbon nanotubes was changed, keeping the equilibrium distance ( $r_{min}$ ) the same for both aromatic carbons and modified atoms. As a result, even for the simulations with CNT types A, B, C, or D the separation between the alkane and the CNT can not be smaller than for the nonmodified case.

Under microcanonical conditions, the effects of the modified atoms along the carbon nanotube over the n-alkanes were evidenced in the affected mobility inside the CNT. As the value of  $\varepsilon$  increased, the favored area (in the  $x$ - $y$  plot) for the probability to locate the center of mass of the alkane decreased. For simulations using the C type CNT with a value of  $\varepsilon$  equal to  $0.3 \text{ kcal mol}^{-1}$ , the mentioned circular shape presented irregularities. Furthermore, increasing the value of  $\varepsilon$  to  $0.9 \text{ kcal mol}^{-1}$  prevented the former shape to appear as the alkane remained confined in a smaller area close to a cluster of modified atoms. Increasing  $\varepsilon$  to  $1.5 \text{ kcal mol}^{-1}$  resulted in the reduction over the previously favored zone, founded to be located close to a cluster of modified atoms. Finally, over the influence of an extremely high value for  $\varepsilon$ , the obtained area showed almost null displacement over the  $x$ -axis and the  $y$ -axis in the grid.

For short alkanes into type A carbon nanotubes, the adsorption process over modified atoms was evidenced from a value of  $\varepsilon$  equal to  $0.3 \text{ kcal mol}^{-1}$ , where the displacement occurred only near the cluster of modified atoms, see Figure 4.15. A stronger effect was experienced by longer alkanes, like undecane, leading to an almost null displacement along the  $x$ -axis or  $y$ -axis. Moreover, for type B carbon nanotubes

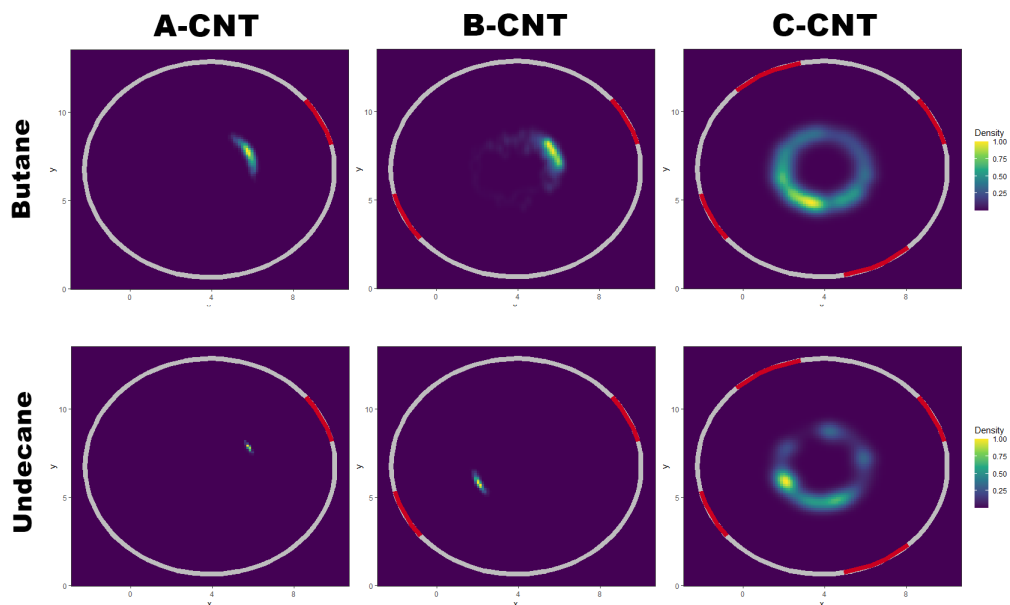


Figure 4.15:  $z$ -axis view of density plots for center of mass positions through simulation time for butane (top) and undecane (bottom) under microcanonical conditions with initial temperature equal to 298 K and  $\varepsilon = 0.3 \text{ kcal mol}^{-1}$ . From left to right: Type A CNT, type B CNT, and type C CNT.

the displacement along  $x$ -axis or  $y$ -axis increased, allowing the motion over the nonmodified section of the nanotube walls (in gray) for short alkanes, while for larger alkanes they kept close to the modified region (in red). For type C nanotubes there is a major displacement over the contour of the inner walls of the nanotube forming a ring similar to the obtained in the nonmodified case, but with some predominant regions favored by the presence of the clusters of modified atoms.

Under canonical conditions, the increased number of molecules led to the results shown in Figure 4.16. For the nonmodified cases the location distribution is quietly regular distributed along the inner contour of the nanotube, and there is a higher probability of finding the alkanes in the center of the nanotube than in comparison with the single-molecule cases (Figure 4.14). For the A-type modification, the highest probability density was located in front of the modified atoms, nonetheless, because of molecules passing one another inside the nanotube there are some minor favored regions over the inner contour for short

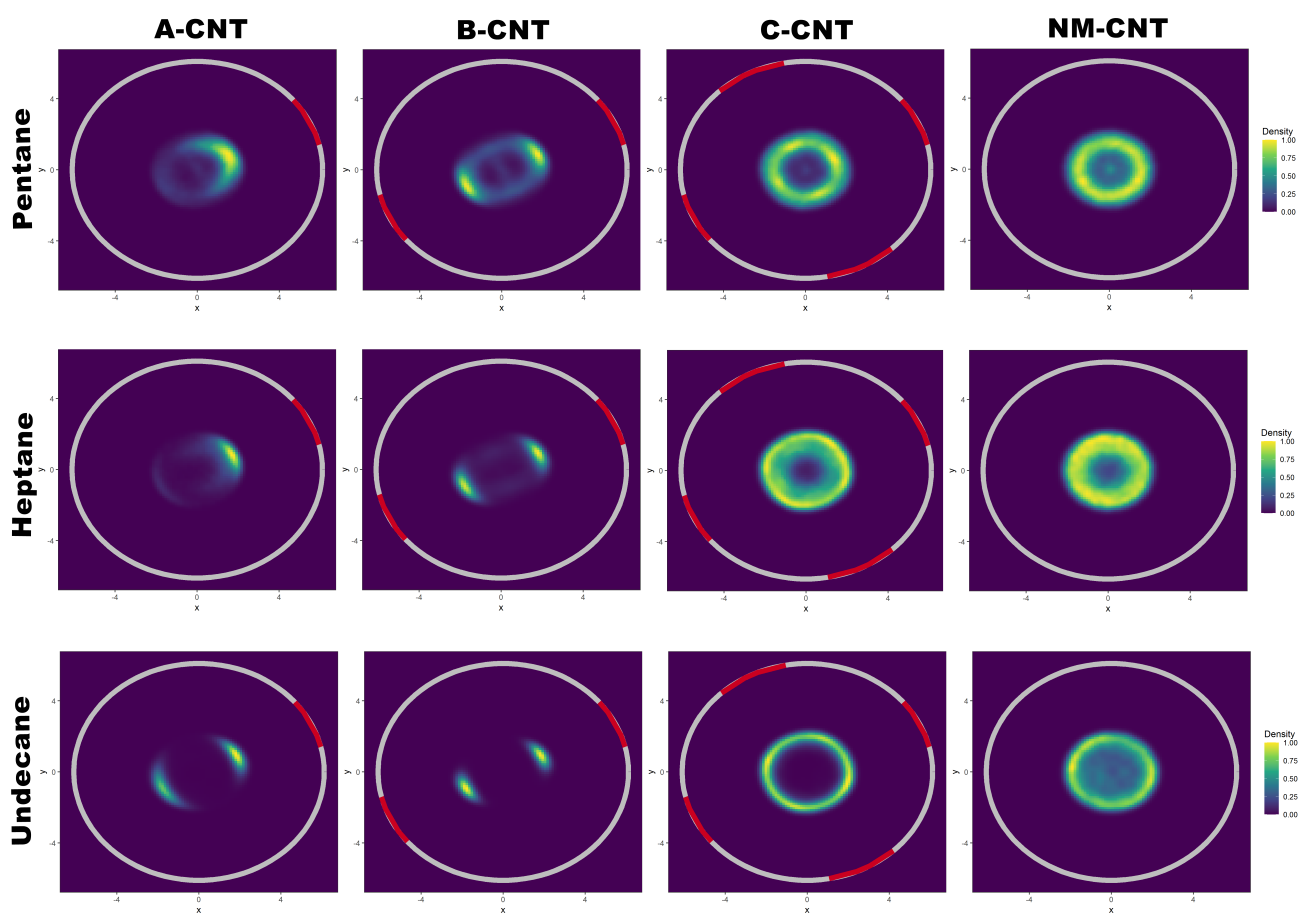


Figure 4.16:  $z$ -axis view of density plots for center of mass positions through simulation time for pentane (top), heptane (middle) and undecane (bottom) with  $\varepsilon$  equal to  $0.2 \text{ kcal mol}^{-1}$  for modified atoms under canonical conditions at 298 K. From left to right: type A CNT, type B CNT, type C CNT, and nonmodified CNT.

alkanes. For large alkanes there is another important region with a high probability of finding the alkanes located on the opposite side to the modified atoms.

For short alkanes in B-type CNTs, there is an increment over the locations surrounding the inner contour of the nonmodified segments of the nanotube, giving rise to what looks like two confronting rings. However, as the alkane length increases, also do the predisposition of alkanes to locate strictly close to the clusters of modified atoms. For the C type CNTs, the higher probability zones formed a ring with some irregularities for the shorter alkanes, where the effect of modified atoms can be differentiated, and for the longer alkanes a well-shaped ring is observed because of less conformational changes on their backbones that could have affected their centers of mass.

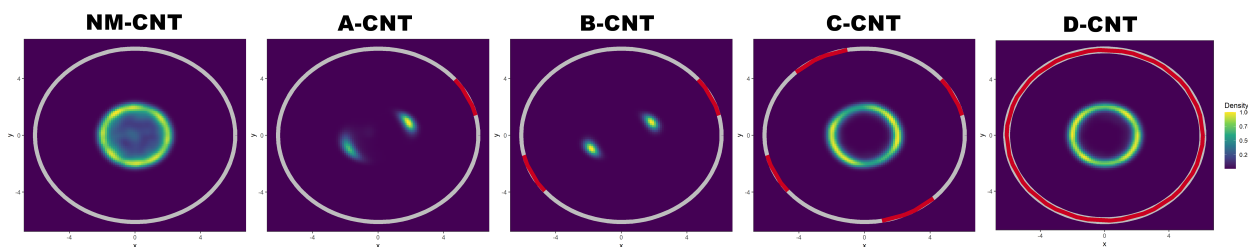


Figure 4.17:  $z$ -axis view of density plots for center of mass positions through simulation time for eicosane under canonical conditions at 298 K. From left to right: nonmodified CNT, type A CNT, type B CNT, type C CNT, and type D CNT. For all modified atoms was set  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$

The larger alkane studied under canonical conditions was eicosane, for which the probability density of being located in the center of a nonmodified nanotube was smaller compared to that being in the contour of the inner wall of the CNT, see Figure 4.17. Besides, such behavior gets confirmed for the type C and type D nanotubes, on which a well-shaped ring can be distinguished as a result of exhibit null probability of finding eicosane molecules in the center of the nanotube. The same results were obtained for all alkanes into the type D modifications.

## 4.5 Diffusion coefficients

The self-diffusion coefficients for diffusion along the nanotube axis,  $D_z$  were obtained. Because of the relatively short simulation time and the single-molecule character of those under microcanonical conditions, their self-diffusion coefficients are listed in Appendix A. The obtained self-diffusion coefficients for the simulations under canonical conditions using nonmodified carbon nanotubes are listed in Table 4.1, being  $N$  the number of alkane molecules in the periodic box. Obtaining, in some cases, similar values to those found in the literature. Previous work from Ebberg<sup>5</sup> reported self-diffusion coefficients for butane and decane in bulk solution via Molecular Dynamics at short times (less than 100 ps). Padilla et al.<sup>79</sup> and Yu et al.<sup>80</sup> reported self-diffusion coefficients for some of the n-alkanes contemplated in this work, both via Molecular Dynamics at higher times that spanned the order of ns. Those values are displayed as the literature self-diffusion coefficients in Table 4.1.

Alkane	N	$D_z \times 10^9$ m <sup>2</sup> /s	$\pm sd \times 10^9$ m <sup>2</sup> /s	$D_{bulk(literature)} \times 10^9$ m <sup>2</sup> /s
butane	18	2.292	0.102	6.140 <sup>5</sup>
pentane	10	1.232	0.078	6.100 <sup>5</sup>
hexane	8	2.629	0.175	3.970 <sup>80</sup>
heptane	7	2.843	0.115	2.928 <sup>80</sup>
octane	6	2.745	0.086	2.190 <sup>80</sup>
nonane	6	1.752	0.089	1.669 <sup>80</sup>
decane	5	1.312	0.229	1.353 <sup>79</sup>
undecane	5	1.533	0.712	1.003 <sup>80</sup>
eicosane	3	3.167	0.603	0.262 <sup>80</sup>

Table 4.1: Obtained  $D_z$  through nonmodified CNTs at 298 K and reference  $D$  in bulk solution.

The closest values to those previously reported in literature were the  $D_z$  of heptane, nonane and decane, which relative errors were 3%, 5%, and 3% respectively. The furthest result was obtained for eicosane, being 11 times bigger than the expected. The more accurate results for  $D_z$  under canonical conditions

are justified by the implementation of the Langevin stochastic thermostat in the models containing carbon nanotubes. Chen et al. reported that the use of a stochastic thermostat within carbon nanotubes structures serves to effectively emulate tubes with corrugated walls that will lead to a correct momentum exchange between the atoms and the walls of the nanotube, but also will reproduce the diffuse reflections needed<sup>81</sup>.

In resume, the effect of corrugated walls on the one-dimensional diffusion process is of vital importance for accurately simulate random walk diffusion through cylindrical pores. Specifically, the random walk for our n-alkanes through CNTs followed a Fickian diffusion as the mean square displacement was proportional to time, conversely to the non observed single-file diffusion where a square root time dependence would be expected. Single-file diffusion could have been achieved by CNTs with smaller diameters and their fluid-wall collisions would be governed by specular reflections<sup>82</sup>.

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	1.692	0.103
pentane	16	2.424	0.063
hexane	12	1.805	0.249
heptane	10	0.780	0.176
octane	7	1.286	0.348
nonane	10	1.953	0.117
decane	8	2.511	0.171
undecane	12	2.279	0.066
eicosane	6	2.789	1.019

Table 4.2: Obtained  $D_z$  through A-type CNTs at 298 K with  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$ .

The value of  $\varepsilon$  of  $0.20 \text{ kcal mol}^{-1}$  used for the modified atoms in Tables 4.2, 4.3, and 4.4 is comparable with various parameters for nitrogen from the CHARMM22 force field. Including representations of primary, secondary and tertiary nitrogen atoms, covering the N-terminal of amides, besides proline, some macrocycles, histidine, triptophane, and carbon from cyano groups, among others<sup>46</sup>. Smaller  $\varepsilon$  values encompass the range of values used to describe oxygen, the same that usually is found at  $0.12 \text{ kcal mol}^{-1}$

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.385	0.092
pentane	16	2.357	0.166
hexane	12	2.658	0.118
heptane	10	2.117	0.227
octane	7	2.400	0.154
nonane	10	0.800	0.351
decane	8	4.444	0.120
undecane	12	0.088	0.296
eicosane	6	0.358	0.031

Table 4.3: Obtained  $D_z$  through B-type CNTs at 298 K with  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$ .

but can increase up to  $0.165 \text{ kcal mol}^{-1}$  for oxygen in carbon dioxide. Heavier atoms like sulfur has a  $\varepsilon$  of  $0.45 \text{ kcal mol}^{-1}$ . Moreover, some ions are described by  $\varepsilon$  values in the studied range as can be seen in Tables 4.5 - 4.11, for instance sodium has a value of  $0.0469 \text{ kcal mol}^{-1}$ , potassium a value of  $0.087 \text{ kcal mol}^{-1}$ , calcium a value of  $0.120 \text{ kcal mol}^{-1}$ , chloride a value of  $0.150 \text{ kcal mol}^{-1}$ , and cesium a value of  $0.190 \text{ kcal mol}^{-1}$ .

The  $D_z$  values for modified CNTs A, B and C with  $\varepsilon$  equal to  $0.20 \text{ kcal mol}^{-1}$  are shown in Tables 4.2, 4.3, and 4.4, respectively. Comparing the results obtained by the simulations using modifications A, B, C, and nonmodified CNTs, the presence of B-type modification led to maximum  $D_z$  values for hexane and decane. Simulations using C-type carbon nanotube increased  $D_z$  for butane, octane, and nonane, meanwhile under the effect of modification A, pentane and undecane, founded their maximum  $D_z$  values. The nonmodified CNT resulted in higher  $D_z$  values for heptane and eicosane.

The  $D_z$  values under the effect of D-type modification are shown for different  $\varepsilon$  values in Tables going from 4.5 to 4.11. For butane two local maximum appeared, the first at  $0.07 \text{ kcal mol}^{-1}$ , which is the value for a nonmodified CNT, followed by a higher second maximum at  $0.17 \text{ kcal mol}^{-1}$ , and finally a low value at  $0.20 \text{ kcal mol}^{-1}$ . For pentane, the higher  $D_z$  was obtained at  $0.17 \text{ kcal mol}^{-1}$ , exhibiting low  $D_z$  values

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.742	0.219
pentane	16	1.663	0.298
hexane	12	0.825	0.239
heptane	10	1.070	0.099
octane	7	3.948	0.173
nonane	10	2.399	0.132
decane	8	1.439	0.162
undecane	12	1.349	0.169
eicosane	6	2.878	0.162

Table 4.4: Obtained  $D_z$  through C-type CNTs at 298 K with  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$ .

from 0.10 to 0.15  $\text{kcal mol}^{-1}$ . In the case of hexane, the higher  $D_z$  was obtained at the nonmodified  $\varepsilon$  value of 0.07  $\text{kcal mol}^{-1}$ . For heptane two peaks were obtained, a small one at 0.10  $\text{kcal mol}^{-1}$  and a higher at 0.17  $\text{kcal mol}^{-1}$ . Octane exhibited a local peak at 0.05  $\text{kcal mol}^{-1}$  and a global maximum at 0.15  $\text{kcal mol}^{-1}$ . The higher  $D_z$  values for nonane were obtained at the lower  $\varepsilon$  ones, being in the range from 0.05  $\text{kcal mol}^{-1}$  to 0.10  $\text{kcal mol}^{-1}$ . For decane the maximum  $D_z$  was obtained at 0.05  $\text{kcal mol}^{-1}$ , followed by lower values until a small peak arose at 0.15  $\text{kcal mol}^{-1}$ . Undecane showed an increase over  $D_z$  until reach 0.15  $\text{kcal mol}^{-1}$  and after that it just started to decrease. Finally, eicosane registered low  $D_z$  values that unexpectedly reached a maximum at 0.20  $\text{kcal mol}^{-1}$ .

The effect of  $\varepsilon$  over alkanes diffusion can be distributed into two main components, the first one is the conformational rearrangement of the alkane backbone and the second one is the adsorption over the nanotube walls. In such a way that the total influence of modified atoms depends on how important are the conformational changes for the alkane in the diffusion process. The effect of low  $\varepsilon$  values over small alkanes will be the result of their few degrees of freedom for rotation, causing the alkane molecule to move freely. Conversely, for bigger alkanes as their degrees of freedom for rotation are higher, the effect of  $\varepsilon$  is mainly exerted over the conformational rearrangements by establishing constraints over the rotation of the



Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	1.072	0.282
pentane	16	1.789	0.132
hexane	12	1.674	0.385
heptane	10	1.663	0.096
octane	12	1.766	0.219
nonane	10	1.724	0.092
decane	8	3.213	0.384
undecane	12	1.090	0.220
eicosane	6	0.309	0.241

Table 4.5: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.05 \text{ kcal mol}^{-1}$ .

dihedral. Therefore, setting higher  $\varepsilon$  values than the base value for CNTs carbon atoms, will induce their dihedral angles to not vary from the trans conformers, without inhibit the diffusion process. Conversely, for shorter alkanes with high  $\varepsilon$  values, as the conformational rearrangement does not exert much more influence, then the predominant process will be adsorption leading to a reduction over the total diffusion along the nanotube. Such differences can lead to future applications of tunable CNTs for separating molecules based on their size.

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.196	0.237
pentane	16	1.743	0.137
hexane	12	2.522	0.205
heptane	10	2.337	0.128
octane	12	0.997	0.135
nonane	10	2.358	0.193
decane	8	1.295	0.132
undecane	12	0.966	0.302
eicosane	6	0.621	0.041

Table 4.6: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.07 \text{ kcal mol}^{-1}$ .

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	1.655	0.178
pentane	16	1.587	0.138
hexane	12	1.791	0.497
heptane	10	2.746	0.076
octane	12	0.746	0.130
nonane	10	2.368	0.093
decane	8	0.636	0.230
undecane	12	2.240	0.054
eicosane	6	1.581	0.191

Table 4.7: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.10 \text{ kcal mol}^{-1}$ .

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.364	0.210
pentane	16	0.399	0.049
hexane	12	1.721	0.205
heptane	10	1.611	0.264
octane	12	1.683	0.074
nonane	10	0.639	0.059
decane	8	0.347	0.066
undecane	12	2.028	0.018
eicosane	6	0.063	0.014

Table 4.8: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.13 \text{ kcal mol}^{-1}$ .

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.381	0.144
pentane	16	0.855	0.141
hexane	12	1.237	0.117
heptane	10	1.760	0.103
octane	12	2.315	0.117
nonane	10	0.966	0.178
decane	8	1.300	0.029
undecane	12	3.043	0.069
eicosane	6	0.020	0.009

Table 4.9: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.15 \text{ kcal mol}^{-1}$ .

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	2.758	0.084
pentane	16	1.894	0.258
hexane	12	0.806	0.137
heptane	10	3.043	0.144
octane	12	2.088	0.236
nonane	10	1.303	0.275
decane	8	0.115	0.054
undecane	12	1.035	0.056
eicosane	6	0.094	0.034

Table 4.10: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.17 \text{ kcal mol}^{-1}$ .

Alkane	N	$D_z \times 10^9 (\text{m}^2/\text{s})$	$\pm sd \times 10^9 (\text{m}^2/\text{s})$
butane	18	1.318	0.165
pentane	16	1.557	0.207
hexane	12	1.770	0.253
heptane	10	1.411	0.109
octane	12	1.000	0.256
nonane	10	0.416	0.033
decane	8	1.033	0.047
undecane	12	1.706	0.065
eicosane	6	3.117	0.334

Table 4.11: Obtained  $D_z$  through D-type CNTs at 298 K with  $\varepsilon = 0.20 \text{ kcal mol}^{-1}$ .

## Chapter 5

# Conclusions & Outlook

The main objective of this research was to design a model to evaluate the diffusion process of n-alkanes through punctually modified single-walled carbon nanotubes (SWCNT) via Molecular Dynamics (MD) simulations based on the adjustment of the atomic van der Waals (vdW) parameters. The study included the comparison of nonmodified carbon nanotubes performance over punctually modified carbon nanotubes in n-alkanes conformational changes and the calculation of self-diffusion coefficients.

The arrangement of the modified atoms over the carbon nanotubes walls was proposed as 1, 2, and 4 equidistant strands parallel to the CNTs axis (A, B, and C arrangements) or equidistant rings that follow the circumference of the CNT (D arrangement). Infinite long narrow nanotubes were accomplished by periodic boundary conditions.

The n-alkanes influence of their chain length study in nonmodified CNTs showed a tendency to decreasing the total translational distance as the number of carbon atoms in the alkane backbone increases for temperatures above 298 K. At low temperatures (10 K) some incongruities were founded, especially for pentane and heptane. The reduction due to the alkane chain growing over the total translational distance is more sensitive to short alkanes, and as the number of carbon atoms increases the addition of carbons has a minor effect in the total translational distance. Regardless of the alkane length, all 4 types of modifications

pointed to a reduction over the total translational distance, meaning that the nonmodified case presented the higher total translational distances.

The study for temperature effect showed a general tendency to increase the total translational distance as the temperature increases. This direct relation was broken down for some results in the range between 318 K and 398 K where local maximum peaks appeared, varying its temperature in depend of the alkane length. This behavior was explained by the levitation effect first reported for diffusion in zeolites by Yashonath et al.<sup>83,84</sup> that is caused by an increase in fluid-wall collisions.

For short time simulations, a preference over trans conformation was obtained. For long time simulations, conformational variability was obtained. In general, it was shown that the induced confinement inside the cylindrical pores generated significant conformational changes from the dominant extended form observed for alkanes in bulk solution towards gauche dominated conformers inside cavities.

When a nonmodified CNT is used, the alkanes tend to locate around the contour of the inner walls and they are located with a lower probability in the center of the CNT. For A-type and B-type modifications the influence of the modified clusters leads the alkane molecules to locate in the surroundings of the cluster of modified atoms. For C-type modifications, alkane atoms are more probably located near the inner contour of the walls and rarely on the center of the CNT, furthermore, for D-type modifications there is almost null probability to found the alkanes in the center.

Self-diffusion coefficients were computed by the Einstein-Smoluchowski relationship, obtaining in the majority of the cases congruent results with very small differences from the literature reported values. Self-diffusion coefficients for modified CNTs did not show a global tendency in comparison with nonmodified CNTs, moreover, it depended on the alkane. For instance, hexane, heptane, and decane got a higher  $D_z$  under B-type CNT; butane, octane, nonane, and eicosane founded their higher  $D_z$  under the C type modification; for pentane and undecane, A-type CNT led to their higher  $D_z$ . Except for heptane and eicosane, either of the modifications with  $\varepsilon$  equal to  $0.2 \text{ kcal mol}^{-1}$  showed a higher  $D_z$  than the obtained for nonmodified CNTs. D-type modification under the same  $\varepsilon$  value ( $0.2 \text{ kcal mol}^{-1}$ ) showed smaller

$D_z$  results for a variety of alkanes. All self-diffusivities reported in this work were computed as one-dimensional diffusivities following the direction of the pore axis, even though, calculations of self-diffusivities performed for three-dimensional diffusion (but not reported in this work) resulted in the same general effects.

The effect of  $\varepsilon$  is different for small and large alkane because of the influence of conformational rearrangements over the diffusion process. Then, for small alkanes there is a major effect over adsorption in the CNT walls, causing low  $\varepsilon$  to favor their mobility. Bigger alkanes rotation is constrained by high  $\varepsilon$  values, inducing trans conformers without inhibit the diffusion process. These differences in diffusivities caused by the intercalated modified atoms in SWCNTs can indicate a route to the future engineering of new devices e.g. filters and sensors, based on tunable properties of SWCNTs.





## Appendix A

# Diffusion Coefficients under microcanonical conditions

On this chapter, are listed the calculated self-diffusion coefficients for simulations under microcanonical conditions only for those with a  $\varepsilon$  value equal to  $0.3 \text{ kcal mol}^{-1}$  versus the simulations performed with a nonmodified CNT.

T (K)	NM $D_z(\text{m}^2/\text{s})$	A $D_z(\text{m}^2/\text{s})$	B $D_z(\text{m}^2/\text{s})$	C $D_z(\text{m}^2/\text{s})$
Butane				
10	$1.09E^{-07} \pm 8.80E^{-09}$	$1.66E^{-13} \pm 2.99E^{-13}$	$8.49E^{-13} \pm 4.37E^{-12}$	$2.44E^{-12} \pm 2.46E^{-12}$
273	$3.89E^{-07} \pm 2.13E^{-08}$	$1.92E^{-06} \pm 1.46E^{-07}$	$2.34E^{-07} \pm 6.36E^{-09}$	$2.01E^{-08} \pm 3.52E^{-09}$
288	$4.06E^{-07} \pm 2.40E^{-08}$	$9.24E^{-09} \pm 7.82E^{-10}$	$5.57E^{-08} \pm 6.49E^{-09}$	$2.72E^{-08} \pm 1.00E^{-09}$
298	$2.72E^{-07} \pm 1.90E^{-08}$	$5.35E^{-07} \pm 3.20E^{-08}$	$5.23E^{-08} \pm 2.99E^{-09}$	$1.78E^{-07} \pm 2.90E^{-08}$
318	$5.16E^{-08} \pm 6.78E^{-09}$	$8.29E^{-07} \pm 4.47E^{-08}$	$1.93E^{-06} \pm 1.30E^{-07}$	$5.70E^{-07} \pm 4.51E^{-08}$
338	$2.96E^{-08} \pm 5.78E^{-09}$	$1.00E^{-07} \pm 5.39E^{-09}$	$1.98E^{-06} \pm 1.60E^{-07}$	$2.17E^{-07} \pm 1.55E^{-08}$
358	$7.52E^{-07} \pm 5.93E^{-08}$	$1.97E^{-07} \pm 4.18E^{-09}$	$1.47E^{-07} \pm 1.18E^{-08}$	$8.23E^{-07} \pm 6.66E^{-08}$

<b>T (K)</b>	<b>NM <math>D_z</math> (m<sup>2</sup>/s)</b>	<b>A <math>D_z</math> (m<sup>2</sup>/s)</b>	<b>B <math>D_z</math> (m<sup>2</sup>/s)</b>	<b>C <math>D_z</math> (m<sup>2</sup>/s)</b>
378	$1.42E^{-06} \pm 8.16E^{-08}$	$1.45E^{-06} \pm 1.10E^{-07}$	$5.75E^{-07} \pm 3.78E^{-08}$	$8.29E^{-07} \pm 6.46E^{-08}$
398	$2.80E^{-08} \pm 1.69E^{-09}$	$7.09E^{-07} \pm 4.18E^{-08}$	$5.91E^{-09} \pm 2.05E^{-09}$	$4.10E^{-07} \pm 2.58E^{-08}$
<b>Pentane</b>				
10	$8.48E^{-14} \pm 2.67E^{-13}$	$4.48E^{-12} \pm 7.16E^{-11}$	$2.51E^{-11} \pm 5.36E^{-11}$	$1.14E^{-12} \pm 1.86E^{-12}$
273	$6.99E^{-07} \pm 5.49E^{-08}$	$5.76E^{-07} \pm 3.67E^{-08}$	$3.52E^{-08} \pm 8.57E^{-10}$	$4.94E^{-09} \pm 1.07E^{-09}$
288	$8.45E^{-07} \pm 2.65E^{-08}$	$3.24E^{-08} \pm 2.84E^{-09}$	$1.25E^{-08} \pm 7.49E^{-10}$	$1.09E^{-07} \pm 6.04E^{-09}$
298	$1.10E^{-06} \pm 7.05E^{-08}$	$2.18E^{-07} \pm 7.13E^{-09}$	$4.22E^{-08} \pm 1.36E^{-09}$	$9.09E^{-07} \pm 3.40E^{-08}$
318	$1.00E^{-07} \pm 5.63E^{-09}$	$4.08E^{-07} \pm 2.70E^{-08}$	$5.12E^{-08} \pm 4.44E^{-09}$	$1.00E^{-07} \pm 6.25E^{-09}$
338	$1.64E^{-07} \pm 5.37E^{-09}$	$1.84E^{-08} \pm 2.16E^{-09}$	$1.32E^{-08} \pm 5.72E^{-10}$	$1.05E^{-07} \pm 2.82E^{-09}$
358	$5.09E^{-08} \pm 2.68E^{-09}$	$7.27E^{-08} \pm 3.90E^{-09}$	$3.77E^{-07} \pm 2.54E^{-08}$	$2.10E^{-07} \pm 1.27E^{-08}$
378	$7.14E^{-08} \pm 2.67E^{-09}$	$1.04E^{-07} \pm 2.57E^{-09}$	$1.91E^{-07} \pm 3.22E^{-09}$	$1.61E^{-06} \pm 1.26E^{-07}$
398	$4.34E^{-07} \pm 3.89E^{-08}$	$1.48E^{-07} \pm 1.10E^{-08}$	$6.45E^{-07} \pm 4.70E^{-08}$	$4.43E^{-07} \pm 3.38E^{-08}$
<b>Hexane</b>				
10	$4.13E^{-08} \pm 1.66E^{-09}$	$2.19E^{-12} \pm 1.30E^{-13}$	$6.50E^{-07} \pm 4.96E^{-08}$	$8.15E^{-11} \pm 2.57E^{-11}$
273	$4.37E^{-07} \pm 2.05E^{-08}$	$5.07E^{-09} \pm 3.94E^{-10}$	$2.02E^{-07} \pm 7.60E^{-09}$	$2.18E^{-08} \pm 2.63E^{-09}$
288	$4.70E^{-07} \pm 3.13E^{-08}$	$2.67E^{-07} \pm 1.16E^{-08}$	$7.60E^{-09} \pm 1.36E^{-09}$	$1.33E^{-08} \pm 1.70E^{-09}$
298	$4.47E^{-07} \pm 3.33E^{-08}$	$4.62E^{-07} \pm 1.17E^{-08}$	$1.05E^{-06} \pm 7.09E^{-08}$	$2.66E^{-07} \pm 1.49E^{-08}$
318	$8.00E^{-07} \pm 6.09E^{-08}$	$7.88E^{-08} \pm 5.25E^{-09}$	$6.52E^{-07} \pm 5.50E^{-08}$	$1.85E^{-07} \pm 3.47E^{-09}$
338	$2.57E^{-08} \pm 1.06E^{-09}$	$1.16E^{-06} \pm 9.05E^{-08}$	$4.76E^{-09} \pm 3.41E^{-09}$	$2.03E^{-07} \pm 6.18E^{-09}$
358	$4.00E^{-08} \pm 5.66E^{-09}$	$4.97E^{-07} \pm 3.32E^{-08}$	$2.67E^{-07} \pm 2.96E^{-08}$	$6.92E^{-07} \pm 4.16E^{-08}$
378	$1.88E^{-07} \pm 9.22E^{-09}$	$4.20E^{-08} \pm 3.04E^{-09}$	$3.39E^{-08} \pm 2.11E^{-09}$	$4.69E^{-07} \pm 3.06E^{-08}$
398	$1.33E^{-06} \pm 9.70E^{-08}$	$7.19E^{-07} \pm 4.14E^{-08}$	$4.40E^{-07} \pm 3.10E^{-08}$	$2.47E^{-08} \pm 6.16E^{-09}$
<b>Heptane</b>				

<b>T (K)</b>	<b>NM <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>A <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>B <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>C <math>D_z</math>(m<sup>2</sup>/s)</b>
10	$2.25E^{-10} \pm 1.18E^{-10}$	$1.75E^{-11} \pm 6.41E^{-12}$	$6.24E^{-12} \pm 7.37E^{-12}$	$1.71E^{-12} \pm 4.14E^{-13}$
273	$2.71E^{-08} \pm 3.05E^{-09}$	$6.17E^{-08} \pm 1.45E^{-09}$	$4.53E^{-08} \pm 6.03E^{-10}$	$6.69E^{-08} \pm 3.61E^{-09}$
288	$4.53E^{-09} \pm 2.31E^{-10}$	$1.46E^{-07} \pm 9.01E^{-09}$	$1.11E^{-06} \pm 8.38E^{-08}$	$3.05E^{-09} \pm 8.97E^{-10}$
298	$3.83E^{-07} \pm 2.52E^{-08}$	$3.92E^{-07} \pm 2.98E^{-08}$	$5.16E^{-08} \pm 9.28E^{-09}$	$1.24E^{-08} \pm 3.25E^{-09}$
318	$1.40E^{-07} \pm 6.29E^{-09}$	$3.65E^{-07} \pm 2.73E^{-08}$	$1.27E^{-07} \pm 6.37E^{-09}$	$2.38E^{-09} \pm 3.96E^{-09}$
338	$4.43E^{-07} \pm 3.41E^{-08}$	$9.11E^{-07} \pm 4.38E^{-08}$	$2.83E^{-07} \pm 1.12E^{-08}$	$1.04E^{-09} \pm 2.71E^{-09}$
358	$9.95E^{-07} \pm 7.78E^{-08}$	$2.77E^{-07} \pm 1.89E^{-08}$	$4.83E^{-07} \pm 3.94E^{-08}$	$1.12E^{-08} \pm 2.64E^{-09}$
378	$1.29E^{-06} \pm 1.01E^{-07}$	$2.86E^{-08} \pm 1.00E^{-09}$	$2.40E^{-07} \pm 9.29E^{-09}$	$2.06E^{-07} \pm 7.55E^{-09}$
398	$4.10E^{-07} \pm 3.30E^{-08}$	$1.81E^{-07} \pm 1.63E^{-08}$	$2.97E^{-08} \pm 1.17E^{-09}$	$1.79E^{-07} \pm 1.23E^{-08}$
<b>Octane</b>				
10	$3.97E^{-09} \pm 1.59E^{-10}$	$2.04E^{-13} \pm 2.08E^{-13}$	$4.04E^{-09} \pm 1.75E^{-10}$	$2.73E^{-11} \pm 1.93E^{-11}$
273	$5.76E^{-08} \pm 2.18E^{-09}$	$2.29E^{-08} \pm 1.82E^{-09}$	$4.91E^{-09} \pm 3.51E^{-09}$	$2.87E^{-08} \pm 6.34E^{-10}$
288	$7.95E^{-08} \pm 1.52E^{-09}$	$5.80E^{-07} \pm 3.57E^{-08}$	$1.25E^{-08} \pm 2.80E^{-09}$	$2.48E^{-08} \pm 1.34E^{-08}$
298	$5.89E^{-09} \pm 1.01E^{-09}$	$2.55E^{-07} \pm 7.24E^{-09}$	$1.53E^{-07} \pm 1.05E^{-08}$	$3.80E^{-08} \pm 1.41E^{-09}$
318	$2.15E^{-08} \pm 3.03E^{-09}$	$3.93E^{-07} \pm 2.62E^{-08}$	$1.35E^{-07} \pm 6.23E^{-09}$	$3.33E^{-07} \pm 2.57E^{-08}$
338	$4.84E^{-07} \pm 3.60E^{-08}$	$4.85E^{-08} \pm 1.49E^{-09}$	$1.14E^{-06} \pm 5.37E^{-08}$	$1.05E^{-09} \pm 2.37E^{-10}$
358	$7.62E^{-07} \pm 5.86E^{-08}$	$5.50E^{-08} \pm 3.50E^{-09}$	$8.40E^{-07} \pm 6.39E^{-08}$	$7.23E^{-09} \pm 3.47E^{-09}$
378	$8.66E^{-07} \pm 6.40E^{-08}$	$1.42E^{-08} \pm 9.12E^{-09}$	$4.58E^{-08} \pm 5.60E^{-09}$	$1.31E^{-07} \pm 1.34E^{-08}$
398	$7.90E^{-07} \pm 6.14E^{-08}$	$1.54E^{-07} \pm 1.06E^{-08}$	$2.80E^{-07} \pm 2.25E^{-08}$	$6.94E^{-07} \pm 5.15E^{-08}$
<b>Nonane</b>				
10	$3.56E^{-08} \pm 1.08E^{-09}$	$8.40E^{-13} \pm 1.47E^{-13}$	$1.47E^{-13} \pm 1.36E^{-13}$	$1.88E^{-10} \pm 9.04E^{-12}$
273	$4.74E^{-08} \pm 4.55E^{-09}$	$1.35E^{-07} \pm 6.31E^{-09}$	$4.67E^{-07} \pm 2.73E^{-08}$	$1.16E^{-07} \pm 7.14E^{-09}$
288	$4.32E^{-07} \pm 2.97E^{-08}$	$1.05E^{-06} \pm 8.23E^{-08}$	$1.76E^{-07} \pm 1.39E^{-08}$	$5.00E^{-09} \pm 5.61E^{-10}$

<b>T (K)</b>	<b>NM <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>A <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>B <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>C <math>D_z</math>(m<sup>2</sup>/s)</b>
298	$4.96E^{-07} \pm 3.24E^{-08}$	$6.62E^{-08} \pm 2.37E^{-09}$	$2.93E^{-07} \pm 2.36E^{-08}$	$1.66E^{-09} \pm 1.99E^{-09}$
318	$2.23E^{-07} \pm 1.47E^{-08}$	$4.08E^{-08} \pm 1.97E^{-09}$	$5.00E^{-09} \pm 4.09E^{-09}$	$1.58E^{-08} \pm 5.03E^{-09}$
338	$7.15E^{-07} \pm 5.64E^{-08}$	$4.50E^{-08} \pm 2.79E^{-09}$	$2.76E^{-07} \pm 1.40E^{-08}$	$2.22E^{-07} \pm 1.33E^{-08}$
358	$2.78E^{-07} \pm 2.16E^{-08}$	$1.30E^{-07} \pm 5.19E^{-09}$	$6.75E^{-07} \pm 5.29E^{-08}$	$1.23E^{-08} \pm 1.35E^{-09}$
378	$1.25E^{-07} \pm 7.99E^{-09}$	$2.37E^{-08} \pm 8.25E^{-10}$	$7.91E^{-08} \pm 1.48E^{-09}$	$7.01E^{-09} \pm 6.72E^{-10}$
398	$6.91E^{-08} \pm 4.47E^{-09}$	$2.77E^{-08} \pm 1.76E^{-09}$	$7.06E^{-07} \pm 5.48E^{-08}$	$4.69E^{-08} \pm 1.07E^{-08}$
<b>Decane</b>				
10	$8.51E^{-08} \pm 6.29E^{-09}$	$1.50E^{-12} \pm 2.77E^{-12}$	$2.87E^{-07} \pm 2.16E^{-08}$	$2.48E^{-08} \pm 2.32E^{-09}$
273	$6.27E^{-07} \pm 1.06E^{-07}$	$8.97E^{-08} \pm 4.58E^{-09}$	$5.83E^{-08} \pm 1.64E^{-09}$	$1.83E^{-08} \pm 1.49E^{-09}$
288	$1.93E^{-08} \pm 4.96E^{-10}$	$4.55E^{-08} \pm 1.39E^{-09}$	$2.34E^{-07} \pm 1.84E^{-08}$	$3.34E^{-09} \pm 1.64E^{-10}$
298	$6.04E^{-07} \pm 4.69E^{-08}$	$6.07E^{-08} \pm 4.67E^{-09}$	$6.48E^{-07} \pm 4.42E^{-08}$	$2.66E^{-08} \pm 6.96E^{-10}$
318	$3.48E^{-07} \pm 2.21E^{-08}$	$1.95E^{-09} \pm 2.10E^{-09}$	$2.02E^{-07} \pm 1.22E^{-08}$	$3.63E^{-10} \pm 1.21E^{-09}$
338	$9.89E^{-09} \pm 2.03E^{-10}$	$6.65E^{-09} \pm 3.40E^{-09}$	$2.31E^{-07} \pm 1.67E^{-08}$	$1.81E^{-07} \pm 1.12E^{-08}$
358	$3.29E^{-07} \pm 1.95E^{-08}$	$7.45E^{-08} \pm 6.16E^{-09}$	$8.94E^{-09} \pm 2.65E^{-09}$	$1.34E^{-08} \pm 1.25E^{-09}$
378	$4.25E^{-08} \pm 1.03E^{-09}$	$2.15E^{-07} \pm 8.57E^{-09}$	$2.24E^{-08} \pm 7.18E^{-10}$	$6.98E^{-08} \pm 4.46E^{-09}$
398	$4.16E^{-07} \pm 2.85E^{-08}$	$9.54E^{-08} \pm 7.03E^{-09}$	$1.25E^{-07} \pm 7.75E^{-09}$	$1.11E^{-08} \pm 1.17E^{-09}$
<b>Undecane</b>				
10	$4.43E^{-07} \pm 3.38E^{-08}$	$4.66E^{-14} \pm 4.50E^{-13}$	$3.25E^{-11} \pm 3.50E^{-11}$	$1.10E^{-10} \pm 1.40E^{-11}$
273	$7.96E^{-08} \pm 4.50E^{-09}$	$6.53E^{-08} \pm 4.21E^{-09}$	$4.50E^{-10} \pm 1.69E^{-10}$	$5.46E^{-09} \pm 1.94E^{-09}$
288	$5.10E^{-07} \pm 3.43E^{-08}$	$2.58E^{-07} \pm 2.11E^{-08}$	$1.73E^{-07} \pm 1.25E^{-08}$	$6.16E^{-10} \pm 1.33E^{-10}$
298	$6.42E^{-07} \pm 4.48E^{-08}$	$2.10E^{-12} \pm 1.95E^{-12}$	$8.37E^{-08} \pm 5.72E^{-09}$	$3.96E^{-09} \pm 2.95E^{-09}$
318	$2.91E^{-07} \pm 2.07E^{-08}$	$9.77E^{-08} \pm 2.36E^{-09}$	$8.61E^{-08} \pm 4.78E^{-09}$	$3.20E^{-08} \pm 1.95E^{-09}$
338	$4.26E^{-07} \pm 3.24E^{-08}$	$3.00E^{-08} \pm 2.78E^{-09}$	$5.87E^{-07} \pm 4.12E^{-08}$	$3.28E^{-08} \pm 1.65E^{-09}$

<b>T (K)</b>	<b>NM <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>A <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>B <math>D_z</math>(m<sup>2</sup>/s)</b>	<b>C <math>D_z</math>(m<sup>2</sup>/s)</b>
358	$2.36E^{-07} \pm 1.44E^{-08}$	$7.33E^{-08} \pm 4.82E^{-09}$	$3.64E^{-07} \pm 9.08E^{-09}$	$5.56E^{-07} \pm 4.06E^{-08}$
378	$1.21E^{-08} \pm 9.21E^{-10}$	$3.85E^{-07} \pm 2.61E^{-08}$	$1.14E^{-07} \pm 7.64E^{-09}$	$7.29E^{-08} \pm 6.38E^{-09}$
398	$8.67E^{-08} \pm 5.22E^{-09}$	$9.11E^{-08} \pm 7.31E^{-09}$	$1.40E^{-07} \pm 5.62E^{-09}$	$1.00E^{-08} \pm 3.84E^{-10}$

Table A.1: Obtained  $D_z$  under microcanonical conditions with  $\varepsilon = 0.30 \text{ kcal mol}^{-1}$ .



## Appendix B

# Algorithms

This chapter includes the general algorithms used to modify the PDB and PSF files, and manage the information of the coordinates in order to obtain the folding degree, dihedral angles and centers of mass during the trajectories of the simulations.

The next codes are written in Python 3.7 and were used to modify the PDB files. The module `leerNtPdb.py` is imported when is needed to access to the information from a generated PDB file in order to make it accomplish the NAMD format.

```
1 # -*- coding: utf-8 -*-
2 import math
3 import numpy
4 import matplotlib.pyplot as plt
5 #Class for each atom in the PDB file
6 class Record:
7     def __init__(self, record, number, name, resname, resid, x, y, z, occupancy,
8                 tfactor, symbol, charge):
9         self.record = record
10        self.number = number
```

```
10     self.name = name
11     self.resname = resname
12     self.resid = resid
13     self.x = x
14     self.y = y
15     self.z = z
16     self.occupancy = occupancy
17     self.tfactor = tfactor
18     self.segid = ''
19     self.symbol = symbol
20     self.charge = charge
21 #Class to work only with the coordinates of the atoms
22 class Vector:
23     def __init__(self,x,y,z):
24         self.x = x
25         self.y = y
26         self.z = z
27         self.modulo = math.sqrt((self.x**2)+(self.y**2)+(self.z**2))
28 def contAtomNumber(ntpdb):
29     archivo = open(ntpdb, 'r')
30     i=0
31     for linea in archivo.readlines():
32         if linea[0:2]=='AT':
33             i += 1
34     return i
35 def contTotalAtomNumber(atomList):
36     i=0
37     for atom in atomList():
38         if atom.record=='ATOM':
39             i += 1
```



```
40     return i
41 def modifNtFormat(atomListNt):
42     for atom in atomListNt:
43         if atom.name=='C':
44             atom.name='CA'
45         if atom.symbol=='H':
46             atom.name='HA'
47         atom.resname = 'ARM'
48         atom.occupancy = 1.00
49     return atomListNt
50 def leerNtPdb(ntpdb):
51     atomNumber = contAtomNumber(ntpdb)
52     print(atomNumber)
53     archivo = open(ntpdb, 'r')
54     linea1 = archivo.readline()
55     linea2 = archivo.readline()
56     i=1
57     conectList=[]
58     atomList=[]
59     lista = []
60     for linea in archivo.readlines():
61         if i <= atomNumber:
62             record, number, name, resname, resid, x, y, z, occupancy, tfactor, symbol=
linea.split()
63             number = int(number)
64             x= float(x)
65             y= float(y)
66             z= float(z)
67             resid = int(resid)
68             occupancy = float(occupancy)
```

```
69         tfactor = float(tfactor)
70         if len(symbol) > 1:
71             charge=symbol[-1]
72             charge=symbol[-2]+charge
73             symbol = symbol[0:-2]
74         else:
75             charge=''
76         atomo = Record(record, number, name, resname, resid, x, y, z, occupancy,
77         tfactor, symbol, charge)
78         atomList.append(atomo)
79         if i > atomNumber:
80             lista=linea.split()
81             conectList.append(lista)
82         i += 1
83     archivo.close()
84     conectList.pop(-1)
85     conectList.pop(-1)
86     atomList = modifNtFormat(atomList)
87     return atomList,conectList
```

Listing B.1: Module leerNtPdb.py

The module modPdbNt.py contains the functions used to change the atom type in carbon nanotubes PDB files,

```
1 # -*- coding: utf-8 -*-
2 from math import *
3 import numpy as np
4 import leerNtPdb as lntpdb
5 import shutil, os
6 paratype='CM1'
7 ntpdb='CNT92.pdb'
```

```
8 chiraln=9
9 chirm=9
10 ntlength=40
11 def ntvalues(ntpdb):
12     chiraln=int(ntpdb[1])
13     chirm=int(ntpdb[2])
14     ntlength=int(ntpdb[3:5])
15     return chiraln, chirm, ntlength
16 def contCarbonInList(atomList):
17     nc=0
18     for atom in atomList:
19         if atom.name=='CA':
20             nc+=1
21     return nc
22 def escribirPdb(atomList, nombre):
23     file = open(nombre, 'w')
24     titulo='COMPND '+nombre+'\n'
25     file.write(titulo)
26     for atom in atomList:
27         cadena = ''
28         atom.record = str(atom.record)
29         while len(atom.record) < 6:
30             atom.record += ' '
31         cadena += atom.record
32         atom.number = str(atom.number)
33         while len(atom.number) < 5:
34             atom.number = ' ' + atom.number
35         cadena += atom.number
36         atom.name = str(atom.name)
37         while len(atom.name) < 4:
```

```
38     atom.name = ' ' + atom.name
39     cadena += atom.name
40     atom.resname = str(atom.resname)
41     while len(atom.resname) < 5:
42         atom.resname = ' ' + atom.resname
43     cadena += atom.resname
44     atom.resid = str(atom.resid)
45     while len(atom.resid) < 6:
46         atom.resid = ' ' + atom.resid
47     cadena+= atom.resid
48     atom.x = str(atom.x)
49     while len(atom.x) < 12:
50         atom.x = ' ' + atom.x
51     cadena += atom.x
52     atom.y = str(atom.y)
53     while len(atom.y) < 8:
54         atom.y = ' ' + atom.y
55     cadena += atom.y
56     atom.z = str(atom.z)
57     while len(atom.z) < 8:
58         atom.z = ' ' + atom.z
59     cadena += atom.z
60     atom.occupancy = str(atom.occupancy)
61     while len(atom.occupancy) < 6:
62         atom.occupancy = ' ' + atom.occupancy
63     cadena += atom.occupancy
64     atom.tfactor = str(atom.tfactor)
65     while len(atom.tfactor) < 6:
66         atom.tfactor = ' ' + atom.tfactor
67     cadena += atom.tfactor
```

```
68     atom.symbol = str(atom.symbol)
69     while len(atom.symbol) < 12:
70         atom.symbol = ' ' + atom.symbol
71     cadena += atom.symbol
72     atom.charge = str(atom.charge)
73     while len(atom.charge) < 2:
74         atom.charge += ' '
75     cadena += atom.charge + '\n'
76     file.write(cadena)
77     totAtomNumber=len(atomList)
78     file.write('END')
79     file.close()
80 def single_bencene_seq_mod(paratype, ntlist, ntpdb,ci):
81     l=0
82     nc=len(ntlist)
83     lista=[ci, ci+1, ci+2, ci+3]
84     for c in range (nc):
85         if c%(chiraln*4) == 0:
86             if c>0:
87                 l+=1
88                 for i in range(0, len(lista)):
89                     lista[i]+=(chiraln*4)
90             if c in lista:
91                 ntlist[c].name=paratype
92     return ntlist
93 def double_bencene_seq_mod(paratype, ntlist,ntpdb):
94     if chiraln==chirm:
95         if chiraln % 2 == 0:
96             ci=3
97     else:
```

```
98         ci=1
99     elif chiraln==0:
100         ci=0
101         ntlist=single_bencene_seq_mod(paratype,ntlist,ntpdb,ci)
102         escribirPdb(ntlist, str('sb'+ntpdb))
103         ci=ci+(2*chiraln)
104         ntlist=single_bencene_seq_mod(paratype,ntlist,ntpdb,ci)
105         escribirPdb(ntlist, str('db'+ntpdb))
106     return ntlist
107 def tetra_bencene_seq_mod(paratype,ntlist,ntpdb):
108     ntlist, conectListNt = lntpdb.leerNtPdb(ntpdb)
109     if chiraln==chiraln:
110         if chiraln % 2 == 0:
111             ci=3
112         else:
113             ci=1
114     elif chiraln==0:
115         ci=0
116     ntlistsb=single_bencene_seq_mod(paratype,ntlist,ntpdb,ci)
117     ci=ci+chiraln
118     ntlistb=single_bencene_seq_mod(paratype,ntlistsb,ntpdb,ci)
119     ci=ci+chiraln
120     ntlistdb=single_bencene_seq_mod(paratype,ntlistb,ntpdb,ci)
121     ci=ci+chiraln
122     ntlisttb=single_bencene_seq_mod(paratype,ntlistdb,ntpdb,ci)
123     escribirPdb(ntlisttb, str('tb'+ntpdb))
124     return ntlist
125 def random_seq_mod(paratype, ntpdb):
126     ntlist, conectListNt = lntpdb.leerNtPdb(ntpdb)
127     if chiraln==chiraln:
```

```
128     for atom in ntlist:
129         randomn=np.random()
130         if randomn>0.5:
131             atom.type=paratype
132     escribirPdb(ntlist, str('rd'+ntpdb))
133     return ntlist
134 def transversal_mod(paratype, ntlist, ntpdb):
135     ntlist, conectListNt = lntpdb.leerNtPdb(ntpdb)
136     if chiraln==chiralm:
137         lenring=4*chiraln
138         l=1
139         cont=0
140         for atom in ntlist:
141             cont+=1
142             if(l//10)==0:
143                 atom.type=paratype
144                 if cont//lenring==0:
145                     l+=1
146     escribirPdb(ntlist, str('an'+ntpdb))
147     return ntlist
148 def transversal_mod_prog(paratype, progparatype, ntpdb):
149     ntlist, conectListNt = lntpdb.leerNtPdb(ntpdb)
150     if chiraln==chiralm:
151         lenring=4*chiraln
152         l=1
153         cont=0
154         for atom in ntlist:
155             cont+=1
156             if((l+1)//10)==0:
157                 atom.type==progparatype
```

```
158         elif(l//10)==0:
159             atom.type=paratype
160         elif((l-1)//10)==0:
161             atom.type==progparatype
162         if cont//lenring==0:
163             l+=1
164     escribirPdb(ntlist, str('ap'+ntpdb))
165     return ntlist
166 def trans_prog_tetralinear_mod(firtype,sectype,thirtype,ntlist,ntpdb):
167     ntlist, conectListNt = lntpdb.leerNtPdb(ntpdb)
168     ntlist=tetra_linear_seq_mod(paratype, ntlist,ntpdb)
169     if chiraln==chiralm:
170         lenring=4*chiraln
171         l=1
172         cont=0
173         for atom in ntlist:
174             cont+=1
175             if(cont==1):
176                 atom.type==sectype
177             elif(cont==2):
178                 atom.type==thirtype
179             elif((l+2)//25)==0:
180                 atom.type==thirtype
181             elif((l+1)//25)==0:
182                 atom.type==sectype
183             elif(l//25)==0:
184                 atom.type=firtype
185             elif((l-1)//25)==0:
186                 atom.type==sectype
187             elif((l-2)//25)==0:
```



```
188         atom.type==thirtytype
189         if cont//lenring==0:
190             l+=1
191     escribirPdb(ntlist, str('ac'+ntpdb))
192     return ntlist
```

Listing B.2: Module modPdbNt.py

The module addAlkaneToPdb.py contains the functions used to insert alkane molecules inside the CNTs PDB files.

```
1 # -*- coding: utf-8 -*-
2 import math
3 import numpy as npy
4 import matplotlib.pyplot as plt
5 import leerNtPdb as lntpdb
6 ntbase = 'CNT92.pdb'
7 def contAcNumber(acpdb):
8     archivo = open(acpdb, 'r')
9     i=0
10    for linea in archivo.readlines():
11        if linea[0]=='H':
12            i += 1
13    return i
14 def contCntNumber(acpdb):
15     archivo = open(acpdb, 'r')
16     i=0
17    for linea in archivo.readlines():
18        if linea[0]=='A':
19            i += 1
20    return i
21 def contTotalAtomNumber(pdb):
```

```
22     archivo = open(pdb, 'r')
23     i=0
24     for linea in archivo.readlines():
25         atom = linea[0:4]
26         if atom == 'ATOM':
27             i += 1
28     return i
29 def contCarbonInList(atomList):
30     nc=0
31     for atom in atomList:
32         if atom.name=='C':
33             nc+=1
34     return nc
35 def contHydrogenInList(atomList):
36     nh=0
37     for atom in atomList:
38         if atom.name=='H':
39             nh+=1
40     return nh
41 def sumaVectores(v1, v2):
42     x = v1.x + v2.x
43     y = v1.y + v2.y
44     z = v1.z + v2.z
45     vector = Vector (x,y,z)
46     return vector
47 def restaVectores(vector1, vector2):
48     x = vector1.x - vector2.x
49     y = vector1.y - vector2.y
50     z = vector1.z - vector2.z
51     vector = Vector (x, y, z)
```

```
52     return vector
53 def modifAcFormat(atomListAc):
54     nc = contCarbonInList(atomListAc)
55     n = 0
56     for atom in atomListAc:
57         if atom.name == 'C':
58             n += 1
59             if n == 1 or n == nc:
60                 atom.name = 'CT3'
61             else:
62                 atom.name = 'CT2'
63         if atom.name == 'H':
64             atom.name = 'HA'
65         atom.record = 'ATOM'
66         atom.occupancy = 0.00
67     return atomListAc
68 def modifOrigen(atomListAc, centro):
69     carbon1=Vector(0.00,0.00,0.00)
70     carbon1.x=atomListAc[1].x
71     carbon1.y=atomListAc[1].y
72     carbon1.z=atomListAc[1].z
73     for atom in atomListAc:
74         atom.x = float(atom.x) + float(centro.x)
75         atom.y = float(atom.y) + float(centro.y)
76         atom.z = float(atom.z) + float(centro.z)
77         atom.x = float(atom.x) - float(carbon1.x)
78         atom.y = float(atom.y) - float(carbon1.y)
79         atom.z = float(atom.z) - float(carbon1.z)
80         atom.x = round(atom.x, 4)
81         atom.y = round(atom.y, 4)
```

```
82     atom.z = round(atom.z, 4)
83     return atomListAc
84 def modifAcNumber(atomListAc, ntpdb):
85     ntAtomNumber = contTotalAtomNumber(ntpdb)
86     n=1
87     for atom in atomListAc:
88         atom.number = n + int(ntAtomNumber)
89         n+=1
90     return atomListAc
91 def addAcToNt(atomListAc, atomListNt):
92     for atom in atomListAc:
93         atomListNt.append(atom)
94     return atomListNt
95 def modifAc(atomListAc, ntpdb):
96     atomListNt = lntpdb.leerNtPdb(ntpdb)
97     print('NT ', len(atomListNt))
98     centro = lntpdb.calcCentro(atomListNt)
99     atomListAc = modifOrigen(atomListAc, centro)
100    atomListAc = modifAcFormat(atomListAc)
101    atomListAc= modifAcNumber(atomListAc, ntpdb)
102    atomListAc = addAcToNt(atomListAc, atomListNt)
103    return atomListAc
104 def leerAcPdb(acpdb):
105     acAtomNumber = contTotalAtomNumber(acpdb)
106     archivo = open(acpdb, 'r')
107     lineal = archivo.readline()
108     linea2 = archivo.readline()
109     i=1
110     conectListAc=[]
111     atomListAc=[]
```

```
112 lista=[]
113 for linea in archivo.readlines():
114     if i <= (acAtomNumber):
115         record, number, name, resname, resid, x, y, z, occupancy, tfactor, symbol=
116         linea.split()
117         number = int(number)
118         x= float(x)
119         y= float(y)
120         z= float(z)
121         resid = int(resid)
122         occupancy = float(occupancy)
123         tfactor = float(tfactor)
124         charge=''
125         if resname == 'ARM':
126             resid = 1
127         if resname == 'LIG':
128             resid = 2
129         atomo = Record(record, number, name, resname, resid, x, y, z, occupancy,
130         tfactor, symbol, charge)
131         atomListAc.append(atomo)
132     if i >(acAtomNumber):
133         lista=linea.split()
134         conectListAc.append(lista)
135     i += 1
136 conectListAc.pop(-1)
137 conectListAc.pop(-1)
138 archivo.close()
139 return atomListAc
140 def escribirPdb(atomList, alcanano, ntpdb):
141     nombre = 'filled'+ntpdb
```

```
140 file = open(nombre, 'w')
141 titulo='COMPND '+nombre+'\nGENERATED BY ADRIAN MURIEL\n'
142 file.write(titulo)
143 for atom in atomList:
144     cadena = ''
145     atom.record = str(atom.record)
146     while len(atom.record) < 6:
147         atom.record += ' '
148     cadena += atom.record
149     atom.number = str(atom.number)
150     while len(atom.number) < 5:
151         atom.number = ' ' + atom.number
152     cadena += atom.number
153     atom.name = str(atom.name)
154     while len(atom.name) < 4:
155         atom.name = ' ' + atom.name
156     cadena += atom.name
157     atom.resname = str(atom.resname)
158     while len(atom.resname) < 5:
159         atom.resname = ' ' + atom.resname
160     cadena += atom.resname
161     atom.resid = str(atom.resid)
162     while len(atom.resid) < 6:
163         atom.resid = ' ' + atom.resid
164     cadena+= atom.resid
165     atom.x = str(atom.x)
166     while len(atom.x) < 12:
167         atom.x = ' ' + atom.x
168     cadena += atom.x
169     atom.y = str(atom.y)
```

```
170     while len(atom.y) < 8:
171         atom.y = ' ' + atom.y
172     cadena += atom.y
173     atom.z = str(atom.z)
174     while len(atom.z) < 8:
175         atom.z = ' ' + atom.z
176     cadena += atom.z
177     atom.occupancy = str(atom.occupancy)
178     while len(atom.occupancy) < 6:
179         atom.occupancy = ' ' + atom.occupancy
180     cadena += atom.occupancy
181     atom.tfactor = str(atom.tfactor)
182     while len(atom.tfactor) < 6:
183         atom.tfactor = ' ' + atom.tfactor
184     cadena += atom.tfactor
185     atom.symbol = str(atom.symbol)
186     while len(atom.symbol) < 12:
187         atom.symbol = ' ' + atom.symbol
188     cadena += atom.symbol
189     atom.charge = str(atom.charge)
190     while len(atom.charge) < 2:
191         atom.charge += ' '
192     cadena += atom.charge + '\n'
193     file.write(cadena)
194     totAtomNumber=len(atomList)
195     file.write('END')
196     file.close()
197 z=5.00 #initial coordinate in z-axis
198 incr=28.92 #Depends on the alcane lenght
199 ntpdb='CNT92.pdb'
```

```
200 alcanos = ['butane', 'pentane', 'hexane', 'heptane', 'octane', 'nonane', 'decane', 'undecane', 'eicosane']
201 for alcano in alcanos:
202     acpdb = 'alkanes/'+alcano+'.pdb'
203     lac=leerAcPdb(acpdb)
204     lnt=leerAcPdb(ntpdb)
205     centro=Vector(0.00,0.00,z)
206     lac=modifOrigen(lac,centro)
207     lac=modifAcNumber(lac, ntpdb)
208     lac=modifAcFormat(lac)
209     atomList=addAcToNt(lac, lnt)
210     ntpdb=alcano+ntpdb
211     escribirPdb(atomList, alcano, ntpdb)
```

Listing B.3: Module addAlkaneToPdb

The module modAllPsf.py is used to follow the required NAMD format for the generated PSF files.

```
1 # -*- coding: utf-8 -*-
2 import math as math
3 import numpy as npy
4 import matplotlib.pyplot as plt
5 ntbase='CNT92'
6 alcanos =['butane', 'pentane', 'hexane', 'heptane', 'octane', 'nonane', 'decane', 'undecane', 'eicosane']
7 for alcano in alcanos:
8     acpsf = 'filled'+alcano+ntbase+'.psf'
9     atomNumber = contAtomNumber(acpsf)
10    archivo = open(acpsf, 'r')
11    lista=[]
12    for i in range (6):
13        linea = archivo.readline()
```



```
14     lista.append(linea)
15     for i in range (atomNumber):
16         cadena = ''
17         linea = archivo.readline()
18         number, resid, resname, name, atomtype, charge, mass, cero =linea.split()
19         if resname == 'ARM':
20             segid = ' NT '
21         if resname == 'LIG':
22             segid = ' AC '
23         if resname == 'ARM':
24             resid = '1'
25         if resname == 'LIG':
26             resid = '2'
27         while len(number)< 8:
28             number= ' '+number
29         while len(segid) < 4:
30             segid= segid + ' '
31         while len(resid) < 3:
32             resid= ' ' + resid
33         while len(resname)<7:
34             resname = ' ' + resname
35         while len(name) < 5:
36             name = name + ' '
37         while len(atomtype)<3:
38             atomtype = atomtype+' '
39         while len(charge)<12:
40             charge = ' ' +charge
41         while len(mass) < 14:
42             mass = ' ' + mass
43         while len(cero)<12:
```

```
44         cero = ' ' + cero
45         cadena=number+segid+resid+resname+' '+name+atomtype+charge+mass+cero+ '\n'
46         lista.append(cadena)
47     for linea in archivo.readlines():
48         lista.append(linea)
49     archivo.close()
50     file = open(acpsf, 'w')
51     for line in lista:
52         file.write(line)
53     file.close()
```

Listing B.4: Module modAllPsf.py

The module procData.py contains the functions used to follow the trajectory generated by NAMD, in order to calculate the end-to-end distance, centers of mass, squared displacements and dihedral angles at each recorded time and save those values in text files,

```
1 # -*- coding: utf-8 -*-
2 import math as math
3 import numpy as np
4 import matplotlib
5 import matplotlib.pyplot as plt
6 import leerNtPdb as lntpdb
7 ntpdb = 'CNT92.pdb'
8 class Atomo:
9     def __init__(self, name, x, y, z):
10         self.name = name
11         self.x = x
12         self.y = y
13         self.z = z
14         if self.name == 'CT2' or self.name == 'CT3':
15             self.masa = 12.0107
```

```
16         if self.name == 'HA':
17             self.masa = 1.0079
18 class Punto:
19     def __init__(self,x,y,z):
20         self.x = x
21         self.y = y
22         self.z = z
23 def extraer_coord(ruta,N):
24     archivo = open(ruta+"coord1.xyz", 'r')
25     linea1 = archivo.readline()
26     nalcano = int(linea1)
27     i = 1
28     n = 0
29     iteraciones = []
30     alcanano = []
31     subalcano=[]
32     for linea in archivo.readlines():
33         if linea == linea1:
34             i = 1
35             n = 0
36             alcanano = []
37             subalcano = []
38         if i > 2:
39             nombre, x, y, z= linea.split()
40             x = float(x)
41             y = float(y)
42             z = float(z)
43             atomo = Atomo(nombre, x, y, z)
44             subalcano.append(atomo)
45             n+=1
```

```
46         if n == N:
47             alcano.append(subalcano)
48             n = 0
49             subalcano=[]
50         if i == nalcano+2:
51             iteraciones.append(alcano)
52         i += 1
53     archivo.close()
54     return iteraciones
55 def calc_masa_total(alcano):
56     masa = 0.0
57     for atomo in alcano:
58         masa += atomo.masa
59     return masa
60 def calc_centro_masa(alcano):
61     cm = Punto(0.0, 0.0, 0.0)
62     mt = calc_masa_total(alcano)
63     for atomo in alcano:
64         cm.x += (atomo.masa*atomo.x)
65         cm.y += (atomo.masa*atomo.y)
66         cm.z += (atomo.masa*atomo.z)
67     cm.x = cm.x/mt
68     cm.y = cm.y/mt
69     cm.z = cm.z/mt
70     return cm
71 def calc_distancia(atomo1, atomo2):
72     distancia = math.sqrt((atomo2.x-atomo1.x)**2+(atomo2.y-atomo1.y)**2+(atomo2.z-
73     atomo1.z)**2)
74     return distancia
75 def solo_carbonos(alcano):
```

```
75     lista_carbonos = []
76     for atomo in alcanos:
77         if atomo.name == 'CT3' or atomo.name == 'CT2':
78             lista_carbonos.append(atomo)
79     return lista_carbonos
80 def distancia_end_to_end(alcanos):
81     lista_carbonos = solo_carbonos(alcanos)
82     distancia = calc_distancia(lista_carbonos[0], lista_carbonos[-1])
83     return distancia
84 def producto_escalar(atomo1, atomo2):
85     producto = atomo1.x*atomo2.x+atomo1.y*atomo2.y+atomo1.z*atomo2.z
86     return producto
87 def sumar_vectores(vector1, vector2):
88     sumax = vector1.x + vector2.x
89     sumay = vector1.y + vector2.y
90     sumaz = vector1.z + vector2.z
91     vector = Punto(sumax, sumay, sumaz)
92     return vector
93 def restar_vectores(vector1, vector2):
94     sumax = vector1.x - vector2.x
95     sumay = vector1.y - vector2.y
96     sumaz = vector1.z - vector2.z
97     vector = Punto(sumax, sumay, sumaz)
98     return vector
99 def cambiar_origen(alcanos, posicion):
100     for c in alcanos:
101         c.x -= alcanos[posicion].x
102         c.y -= alcanos[posicion].y
103         c.z -= alcanos[posicion].z
104     return alcanos
```

```
105 def alc_toNumpy(alcane_list):
106     acnp_list = np.zeros((len(alcane_list),3))
107     cont = 0
108     for alcane in alcane_list:
109         acnp_list[cont] = [alcane.x, alcane.y, alcane.z]
110         cont += 1
111     return acnp_list
112 def torsion(atom1, atom2, atom3, atom4):
113     import numpy as np
114     import math
115     vector1 = np.subtract(atom2, atom1)
116     vector2 = np.subtract(atom3, atom2)
117     vector3 = np.subtract(atom4, atom3)
118     cross12 = np.cross(vector1, vector2)
119     cross23 = np.cross(vector2, vector3)
120     unit12 = cross12/np.sqrt(np.dot(cross12, cross12))
121     unit23 = cross23/np.sqrt(np.dot(cross23, cross23))
122     unit1 = unit23
123     unit2 = vector2/(np.sqrt(np.dot(vector2, vector2)))
124     unit3 = np.cross(unit2, unit1)
125     cosine = np.dot(unit12, unit1)
126     sine = np.dot(unit12, unit3)
127     torsion = -math.atan2(sine, cosine)
128     torsiongrados = np.degrees(theta)
129     return torsiongrados
130 def calc_phipsi(time, alcanos):
131     lista_carbonos = solo_carbonos(alcanos)
132     np_carbonos = alc_toNumpy(lista_carbonos)
133     phi = np.zeros(len(lista_carbonos)-4)
134     psi = np.zeros(len(lista_carbonos)-4)
```

```
135 lista_angulos = []
136 parada = len(lista_carbonos)-4
137 lista_r=[]
138 for c in range(0, parada):
139     phi[c] = torsion(np_carbonos[c],np_carbonos[c+1],np_carbonos[c+2],np_carbonos
140     [c+3])
141     psi[c] = torsion(np_carbonos[c+1],np_carbonos[c+2],np_carbonos[c+3],
142     np_carbonos[c+4])
143     resultado = str(time)+'\t\t'+str(c+1)+'-C'+str(c+2)+'\t'+str(phi[c])+'\t'
144     +str(c+2)+'-C'+str(c+3)+'\t'+str(psi[c])+'\n'
145     lista_angulos.append(resultado)
146     r=str(phi[c])+'\t'+str(psi[c])+'\tC'+str(c+2)+'\n'
147     lista_r.append(r)
148 return phi, psi, lista_angulos, lista_r
149 def atomo_a_string(time, atomo):
150     cadena = str(time)+'\t\t'+str(atomo.x)+'\t\t'+str(atomo.y)+'\t\t'+str(atomo.z)+'\n'
151     return cadena
152 def ntvalues(ntpdb):
153     chiraln=int(ntpdb[1])
154     chirm=ntpdb[2]
155     ntlength=int(ntpdb[3:5])
156     return chiraln, chirm, ntlength
157 def serieMinima(atomList, ntpdb):
158     chiraln, chirm, ntlength=ntvalues(ntpdb)
159     if chiraln == chirm :
160         n=chiraln*4
161     else:
162         n=chiraln*chirm
163     pointList=[]
```

```
161     for i in range (0,n):
162         punto=atomList[i]
163         pointList.append(punto)
164     return pointList
165 def calcDistanciaXY(atomo1, atomo2):
166     distancia = math.sqrt((atomo2.x-atomo1.x)**2+(atomo2.y-atomo1.y)**2)
167     return distancia
168 def calcMaxDistancia(atomList, ntpdb):
169     atomList= serieMinima(atomList, ntpdb)
170     i = 0
171     maxid = 0
172     maxdist = 0.00
173     for atom in atomList:
174         dist = calcDistanciaXY(atom, atomList[0])
175         if dist >= maxdist:
176             maxdist = dist
177             maxid = i
178         i += 1
179     return maxid
180 def open_outs(nametxt):
181     f = open (nametxt,'w')
182     f.close()
183 def open_out_header(nametxt):
184     f = open (nametxt,'w')
185     f.write("Phi\tPsi\tType\n")
186     f.close()
187 def escribir_outs(lista,nametxt):
188     f = open (nametxt,'a')
189     f.writelines(lista)
190     f.close()
```



```
191 plt.ion()
192 nanotube_list, conectListNt = lntpdb.leerNtPdb('CNT92.pdb')
193 centro = lntpdb.calcCentro(nanotube_list)
194 radio = calcDistanciaXY(nanotube_list[0], centro)
195 distanciascerolist=[3.852, 5.013, 6.325, 7.523, 8.819, 10.033, 11.32,
196     12.544000039859695, 24.433645839292996]
197 alnumberlist=[11,10,8,7,6,6,5,5,3]
198 alclenlist=[14,17,20,23,26,29,32,35,62]
199 ymax_list=[]
200 ymin_list=[]
201 alnumber=0
202 alcanos = ['butane', 'pentane', 'hexane', 'heptane', 'octane', 'nonane', 'decane', 'undecane', 'eicosane']
203 for alc in alcanos:
204     ruta = alc+'/'
205     coord = extraer_coord(ruta,alclenlist[alnumber])
206     centro_masa_list = []
207     porc_list = []
208     phi_list = []
209     psi_list = []
210     open_outs(ruta+'centros.txt')
211     open_outs(ruta+'distancias.txt')
212     open_outs(ruta+'angulos_phipsi.txt')
213     open_out_header(ruta+'phi_psi.tsv')
214     timecont = 500
215     do = distanciascerolist[alnumber]
216     N=alnumberlist[alnumber]
217     for alcanano in range(len(coord)):
218         dist_sum=0
219         for n in range(N):
```

```
219     timeinps = round(timecont*0.001,3)
220     centro_masa = calc_centro_masa(coord[alcano][n])
221     centro_masa_list.append(centro_masa)
222     centro_masa = atomo_a_string(timeinps, centro_masa)
223     escribir_outs(centro_masa, ruta+'centros.txt')
224     distancia = distancia_end_to_end(coord[alcano][n])
225     dist_sum += distancia
226     phi, psi, lista_phipsi, lista_r = calc_phipsi(timeinps, coord[alcano][n])
227     escribir_outs(lista_phipsi, ruta+'angulos_phipsi.txt')
228     escribir_outs(lista_r, ruta+'phi_psi.tsv')
229     phi_list.append(phi)
230     psi_list.append(psi)
231     dist_avg = dist_sum/N
232     porc = (do-dist_avg)*100/do
233     porc_list.append([timeinps, porc])
234     distancia = str(timeinps) + '\t\t' + str(dist_avg) + '\n'
235     escribir_outs(distancia, ruta+'distancias.txt')
236     timecont+=500
```

Listing B.5: Module procData.py

# Bibliography

- [1] Lippert, R. A.; Bowers, K. J.; Dror, R. O.; Eastwood, M. P.; Gregersen, B. A.; Klepeis, J. L.; Kolossvary, I.; Shaw, D. E. A common, avoidable source of error in molecular dynamics integrators. *The Journal of chemical physics* **2007**, *126*, 046101.
- [2] Slatt, R. M. *Developments in Petroleum Science*; Elsevier, 2013; Vol. 61; pp 39–93.
- [3] Berrios, R.; Marak, A.; Morgenstern, S. Explaining hydrocarbon nationalization in Latin America: Economics and political ideology. *Review of International Political Economy* **2011**, *18*, 673–697.
- [4] Lin, F.-Y.; MacKerell, A. D. *Biomolecular Simulations*; Springer, 2019; pp 21–54.
- [5] Edberg, R. A. Molecular dynamics studies of n-alkanes. **1987**,
- [6] Anzar, N.; Hasan, R.; Tyagi, M.; Yadav, N.; Narang, J. Carbon nanotube-A review on Synthesis, Properties and plethora of applications in the field of biomedical science. *Sensors International* **2020**, *1*, 100003.
- [7] Niyogi, S.; Hamon, M.; Hu, H.; Zhao, B.; Bhowmik, P.; Sen, R.; Itkis, M.; Haddon, R. Chemistry of single-walled carbon nanotubes. *Accounts of Chemical Research* **2002**, *35*, 1105–1113.
- [8] Duclaux, L. Review of the doping of carbon nanotubes (multiwalled and single-walled). *Carbon* **2002**, *40*, 1751–1764.

- [9] Pandurangappa, M.; Raghu, G. K. *Carbon Nanotubes Applications on Electron Devices*; IntechOpen, 2011.
- [10] Vairavapandian, D.; Vichchulada, P.; Lay, M. D. Preparation and modification of carbon nanotubes: Review of recent advances and applications in catalysis and sensing. *Analytica chimica acta* **2008**, *626*, 119–129.
- [11] Sun, Y.; He, K.; Zhang, Z.; Zhou, A.; Duan, H. Real-time electrochemical detection of hydrogen peroxide secretion in live cells by Pt nanoparticles decorated graphene–carbon nanotube hybrid paper electrode. *Biosensors and Bioelectronics* **2015**, *68*, 358–364.
- [12] Aradhya, S. V.; Frei, M.; Hybertsen, M. S.; Venkataraman, L. Van der Waals interactions at metal/organic interfaces at the single-molecule level. *Nature materials* **2012**, *11*, 872–876.
- [13] Rapaport, D. C.; Rapaport, D. C. R. *The art of molecular dynamics simulation*; Cambridge university press, 2004.
- [14] Allen, M. P.; Tildesley, D. J. *Computer simulation in chemical physics*; Springer Science & Business Media, 2012; Vol. 397.
- [15] Goldstein, H.; Poole, C.; Safko, J. *Classical mechanics*. 3rd. 2002.
- [16] Coveney, P. V.; Wan, S. On the calculation of equilibrium thermodynamic properties from molecular dynamics. *Physical Chemistry Chemical Physics* **2016**, *18*, 30236–30240.
- [17] Oliveira, C. R. d.; Werlang, T. Ergodic hypothesis in classical statistical mechanics. *Revista Brasileira de Ensino de Física* **2007**, *29*, 189–201.
- [18] Schlick, T. *Molecular modeling and simulation: an interdisciplinary guide*; Springer, 2010; pp 425–461.

- [19] Jensen, F. *Introduction to computational chemistry*; John Wiley & Sons, 2017.
- [20] Choe, J.-I.; Kim, B. Determination of proper time step for molecular dynamics simulation. *BULLETIN-KOREAN CHEMICAL SOCIETY* **2000**, *21*, 419–424.
- [21] Verlet, L. Computer" experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Physical review* **1967**, *159*, 98.
- [22] Kutteh, R.; Straatsma, T. P. *Reviews in Computational Chemistry*; John Wiley & Sons, Inc., 2007; pp 75–136.
- [23] Andersen, H. C. Rattle: A “velocity” version of the shake algorithm for molecular dynamics calculations. *Journal of Computational Physics* **1983**, *52*, 24–34.
- [24] Miyamoto, S.; Kollman, P. A. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *Journal of computational chemistry* **1992**, *13*, 952–962.
- [25] Jiang, W.; Chipot, C.; Roux, B. Computing relative binding affinity of ligands to receptor: An effective hybrid single-dual-topology free-energy perturbation approach in NAMD. *Journal of chemical information and modeling* **2019**, *59*, 3794–3802.
- [26] Ohashi, Y. *Reactivity in molecular crystals*; John Wiley & Sons, 2008.
- [27] Siepmann, J. I.; Karaborni, S.; Smit, B. Simulating the critical behaviour of complex fluids. *Nature* **1993**, *365*, 330–332.
- [28] Nath, S. K.; Escobedo, F. A.; de Pablo, J. J.; Patramai, I. Simulation of Vapor- Liquid Equilibria for Alkane Mixtures. *Industrial & engineering chemistry research* **1998**, *37*, 3195–3202.
- [29] Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. Development and testing of the OPLS all-atom

- force field on conformational energetics and properties of organic liquids. *Journal of the American Chemical Society* **1996**, *118*, 11225–11236.
- [30] Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; De Vries, A. H. The MARTINI force field: coarse grained model for biomolecular simulations. *The journal of physical chemistry B* **2007**, *111*, 7812–7824.
- [31] Martin, M. G.; Siepmann, J. I. Transferable potentials for phase equilibria. 1. United-atom description of n-alkanes. *The Journal of Physical Chemistry B* **1998**, *102*, 2569–2577.
- [32] Maerzke, K. A.; Siepmann, J. I. Transferable potentials for phase equilibria- coarse-grain description for linear alkanes. *The Journal of Physical Chemistry B* **2011**, *115*, 3452–3465.
- [33] Eggimann, B. L.; Sunnarborg, A. J.; Stern, H. D.; Bliss, A. P.; Siepmann, J. I. An online parameter and property database for the TraPPE force field. *Molecular Simulation* **2014**, *40*, 101–105.
- [34] Brooks, B.; Brooks III, C.; Mackerell Jr, A.; Nilsson, L.; Petrella, R.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S. CHARMM: the biomolecular simulation program. *Journal of computational chemistry* **2009**, *30*, 1545–1614.
- [35] Weiner, P. K.; Kollman, P. A. AMBER: Assisted model building with energy refinement. A general program for modeling molecules and their interactions. *Journal of Computational Chemistry* **1981**, *2*, 287–303.
- [36] Young, D. *Computational chemistry: a practical guide for applying techniques to real world problems*; John Wiley & Sons, 2004.
- [37] Knaap, H.; Beenakker, J. The Lennard-Jones 6–12 potential parameters of H<sub>2</sub> and D<sub>2</sub>. *Physica* **1961**, *27*, 523–530.

- [38] Trockhymchuk, A.; Alejandre, J. ÅComputer simulations of liqM uid/vapor interface in Lennard-MJones fluids: Some questions and anM swersG. *J. Chem. Phys* **1999**, *111*, 8510.
- [39] Adams, J. Bonding energy models. **2001**,
- [40] Padua, D. *Encyclopedia of parallel computing*; Springer Science & Business Media, 2011.
- [41] Phillips, J. C.; Braun, R.; Wang, W.; Gumbart, J.; Tajkhorshid, E.; Villa, E.; Chipot, C.; Skeel, R. D.; Kale, L.; Schulten, K. Scalable molecular dynamics with NAMD. *Journal of computational chemistry* **2005**, *26*, 1781–1802.
- [42] Phillips, J. C. *Proceedings of the Practice and Experience on Advanced Research Computing*; 2018; pp 1–6.
- [43] Aluru, S. *Handbook of computational molecular biology*; CRC Press, 2005.
- [44] Bernardi, R. *et al.* NAMD User’s Guide. version Version 2.13, University of Illinois and Beckman Institute.
- [45] Zhou, K. *Carbon nanomaterials: modeling, design, and applications*; CRC Press, 2019.
- [46] Vanommeslaeghe, K.; Hatcher, E.; Acharya, C.; Kundu, S. CHARMM General Force Field: A Force Field for Drug-Like Molecules Compatible with the CHARMM All-Atom Additive Biological Force Fields. *Journal of computational chemistry* **2010**, *31*, 671–690.
- [47] Makov, G.; Payne, M. Periodic boundary conditions in ab initio calculations. *Physical Review B* **1995**, *51*, 4014.
- [48] Cisneros, G. A.; Karttunen, M.; Ren, P.; Sagui, C. Classical electrostatics for biomolecular simulations. *Chemical reviews* **2014**, *114*, 779–814.

- [49] Holden, Z. C.; Richard, R. M.; Herbert, J. M. Periodic boundary conditions for QM/MM calculations: Ewald summation for extended Gaussian basis sets. *The Journal of chemical physics* **2013**, *139*, 244108.
- [50] Pan, C.; Yi, S.; Hu, Z. The effect of electrostatic boundaries in molecular simulations: symmetry matters. *Physical Chemistry Chemical Physics* **2017**, *19*, 4861–4876.
- [51] He, X. Q.; Kitipornchai, S.; Wang, C.; Liew, K. Modeling of van der Waals force for infinitesimal deformation of multi-walled carbon nanotubes treated as cylindrical shells. *International Journal of Solids and Structures* **2005**, *42*, 6032–6047.
- [52] Iijima, S. Helical microtubules of graphitic carbon. *nature* **1991**, *354*, 56–58.
- [53] Collins, P. G.; Avouris, P. Nanotubes for electronics. *Scientific american* **2000**, *283*, 62–69.
- [54] Odom, T. W.; Huang, J.-L.; Kim, P.; Lieber, C. M. Atomic structure and electronic properties of single-walled carbon nanotubes. *Nature* **1998**, *391*, 62–64.
- [55] Grumezescu, A. M., Ed. *Surface Chemistry of Nanobiomaterials*; William Andrew Publishing, 2016; pp 301–327.
- [56] Muhulet, A.; Miculescu, F.; Voicu, S. I.; Schütt, F.; Thakur, V. K.; Mishra, Y. K. Fundamentals and scopes of doped carbon nanotubes towards energy and biosensing applications. *Materials today energy* **2018**, *9*, 154–186.
- [57] Eatemadi, A.; Daraee, H.; Karimkhanloo, H.; Kouhi, M.; Zarghami, N.; Akbarzadeh, A.; Abasi, M.; Hanifehpour, Y.; Joo, S. W. Carbon nanotubes: properties, synthesis, purification, and medical applications. *Nanoscale research letters* **2014**, *9*, 393.
- [58] Budyka, M.; Zyubina, T.; Ryabenko, A.; Lin, S.; Mebel, A. Bond lengths and diameters of armchair single wall carbon nanotubes. *Chemical physics letters* **2005**, *407*, 266–271.



- [59] Sun, C.-H.; Yin, L.-C.; Li, F.; Lu, G.-Q.; Cheng, H.-M. Van der Waals interactions between two parallel infinitely long single-walled nanotubes. *Chemical physics letters* **2005**, *403*, 343–346.
- [60] Sun, C.-H.; Lu, G.-Q.; Cheng, H.-M. Simple approach to estimating the van der Waals interaction between carbon nanotubes. *Physical Review B* **2006**, *73*, 195414.
- [61] Rajter, R. F.; French, R. H.; Ching, W.-Y.; Podgornik, R.; Parsegian, V. A. Chirality-dependent properties of carbon nanotubes: electronic structure, optical dispersion properties, Hamaker coefficients and van der Waals–London dispersion interactions. *RSC advances* **2013**, *3*, 823–842.
- [62] Dappe, Y. J.; Scipioni, R. Role of the van der Waals forces in the ability of a double-walled carbon nanotube to accommodate a C<sub>60</sub> molecule: The example of C<sub>60</sub>@(15, 0)@(24, 0). *Physical Review B* **2011**, *84*, 193409.
- [63] Rawn, J. D.; Ouellette, R. J. *Organic Chemistry: Structure, Mechanism, Synthesis*; Academic Press, 2018.
- [64] Smith, P. W. G.; Tatchell, A. R. *Organic Chemistry for General Degree Students: Fundamental Aliphatic Chemistry*; Pergamon Press, 1965; Vol. 1.
- [65] Verma, N.; Khanna, S.; Kapila, B. *Comprehensive chemistry XI*; Laxmi publications, 2010.
- [66] Bettelheim, F. A.; Brown, W. H.; Campbell, M. K.; Farrell, S. O.; Torres, O. *Introduction to general, organic and biochemistry*; Nelson Education, 2012.
- [67] Bartell, L. S.; Kohl, D. Structure and rotational isomerization of free hydrocarbon chains. *The Journal of Chemical Physics* **1963**, *39*, 3097–3105.
- [68] Thomas, L. L.; Christakis, T. J.; Jorgensen, W. L. Conformation of alkanes in the gas phase and pure liquids. *The Journal of Physical Chemistry B* **2006**, *110*, 21198–21204.

- [69] Yin, D.; MacKerell Jr, A. D. Combined ab initio/empirical approach for optimization of Lennard–Jones parameters. *Journal of computational chemistry* **1998**, *19*, 334–348.
- [70] Vermaas, J. V.; Hardy, D. J.; Stone, J. E.; Tajkhorshid, E.; Kohlmeyer, A. TopoGromacs: Automated topology conversion from CHARMM to GROMACS within VMD. 2016.
- [71] Davidchack, R. L.; Handel, R.; Tretyakov, M. Langevin thermostat for rigid body dynamics. *The Journal of chemical physics* **2009**, *130*, 234101.
- [72] Ramachandran, G. N. Stereochemistry of polypeptide chain configurations. *J. Mol. Biol.* **1963**, *7*, 95–99.
- [73] Lovell, S. C.; Davis, I. W.; Arendall III, W. B.; De Bakker, P. I.; Word, J. M.; Prisant, M. G.; Richardson, J. S.; Richardson, D. C. Structure validation by  $C\alpha$  geometry:  $\phi$ ,  $\psi$  and  $C\beta$  deviation. *Proteins: Structure, Function, and Bioinformatics* **2003**, *50*, 437–450.
- [74] Philibert, J. One and a half century of diffusion: Fick, Einstein before and beyond. **2006**,
- [75] Jakobtorweihen, S.; Keil, F.; Smit, B. Temperature and size effects on diffusion in carbon nanotubes. *The Journal of Physical Chemistry B* **2006**, *110*, 16332–16336.
- [76] Ajami, D.; Rebek, J. Compressed alkanes in reversible encapsulation complexes. *Nature chemistry* **2009**, *1*, 87–90.
- [77] Wei, C.; Srivastava, D. Theory of transport of long polymer molecules through carbon nanotube channels. *Physical Review Letters* **2003**, *91*, 235901.
- [78] Wanjari, P. P.; Sangwai, A. V.; Ashbaugh, H. S. Confinement induced conformational changes in n-alkanes sequestered within a narrow carbon nanotube. *Phys. Chem. Chem. Phys.* **2012**, *14*, 2702–2709.

- [79] Padilla, P.; Toxværd, S. Self-diffusion in n-alkane fluid models. *The Journal of chemical physics* **1991**, *94*, 5650–5654.
- [80] Yu, Y.-X.; Gao, G.-H. Lennard–Jones chain model for self-diffusion of n-alkanes. *International journal of thermophysics* **2000**, *21*, 57–70.
- [81] Chen, Q.; Moore, J. D.; Liu, Y.-C.; Roussel, T. J.; Wang, Q.; Wu, T.; Gubbins, K. E. Transition from single-file to Fickian diffusion for binary mixtures in single-walled carbon nanotubes. *The Journal of chemical physics* **2010**, *133*, 094501.
- [82] Bhatia, S. K.; Chen, H.; Sholl, D. S. Comparisons of diffusive and viscous contributions to transport coefficients of light gases in single-walled carbon nanotubes. *Molecular Simulation* **2005**, *31*, 643–649.
- [83] Yashonath, S.; ; Chitra Rajappa, b. Temperature dependence of the levitation effect Implications for separation of multicomponent mixtures. *Faraday Discuss.* **1997**, *106*, 105–118.
- [84] Yashonath, S.; Santikary, P. Diffusion in zeolites: Anomalous dependence on sorbate diameter. *The Journal of chemical physics* **1994**, *100*, 4013–4016.