Study of Water Transport Through Carbon Nanotubes

- Final Project on Nanoscale Systems Simulations -

Daniel Bedmar Romero

Studying water transport at the molecular scale is of great scientific interest, as water plays a key role as a solvent in many chemical processes, especially in biological ones. As science progresses at the molecular scale, understanding how water diffuses through small channels becomes more important. In this project, we will study the saturation of water transport through carbon nanotubes.

1 Introduction

During the 21st century, carbon nanotubes have emerged as the star molecule of nanoscience due to the great interest they, as well as their progenitor, graphene, have aroused in the scientific community. This interest is justified because these materials exhibit impressive properties.

Carbon nanotubes have demonstrated incredible electronic properties, as depending on their structure they can be conductors or semiconductors, impressive strength along with impressive flexibility, high heat dissipation capacity, and many more properties that have placed carbon nanotubes in the spotlight of different fields of research, but in our case, they are not the ones that interest us the most.

The study of carbon nanotubes in the field of fluidics or nanofluidics is justified because carbon nanotubes have a very high aspect ratio, i.e., their length is much greater compared to their diameter, they have extremely smooth and, moreover, very hydrophobic surfaces, these properties, together with diameters below or around the nanometer (pardon the redundancy) cause water transport through carbon nanotubes to be extremely efficient.

Despite these properties, the structure of a carbon nanotube, as can be seen in Figure (1), is very simple, as a carbon nanotube can be imagined as the wrapping of a graphene sheet, and, depending on how this hypothetical wrapping occurs, we can define a "chirality vector" (n,m) which will give us both the chirality and the diameter of the nanotube, defined as:

$$d_{in} = \frac{a}{\pi}\sqrt{n^2 + m^2 + nm} - 2r_c \qquad (1)$$

2 Methodology

2.1 Structure Preparation

To simulate water transport through carbon nanotubes, I have used the NAMD program [2] to employ molecular dynamics methods, as these methods provide the precision required for my studies, but we can use systems with quite a few atoms (3732 in my case) without the computational cost being outrageous. For molecular dynamics simulations, we need different input files.

First of all, we need a .pdb file and a .psf file in which the coordinates of all the atoms are defined, as well as the bonds

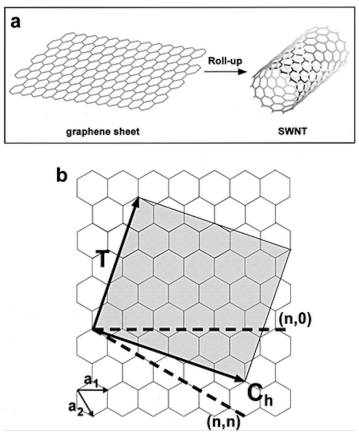

Figure 1: Diagram of the structure of a carbon nanotube [1]

they form, the angles of these bonds, and the different dihedrals. With these files, we are defining the structure to be simulated and some basic properties of it, such as the characteristics of the atoms.

To build my structure 2, I have used materials that I had already prepared in a previous session of this course. In this session, a system consisting of a (6,6) nanotube with a radius of 8.1 Å and a length of 13.5 Å with two water reservoirs at each end was constructed. In my case, I duplicated this structure to obtain more significant results. The generation of this structure,

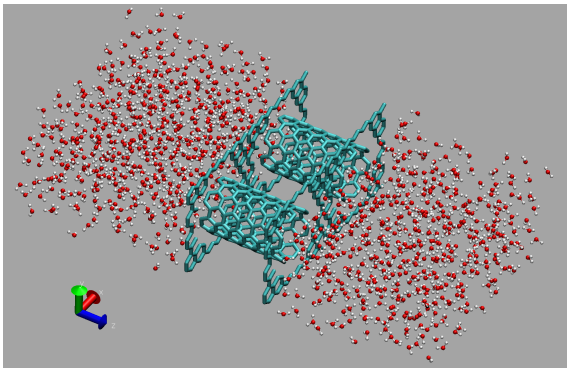

Figure 2: Structure used in the simulations

and its visualization, was performed with the VMD program [3], as this program comes with very useful plugins for modeling structures and a Tk console to manipulate the system with ease.

The first step is to use the "Nanotube Builder" plugin [4] to generate a carbon nanotube. Then, with the same plugin, create a graphene sheet of 15x15 Å area. Next, we will make a circular hole in the center of the graphene sheet and move the sheet to align the hole with the mouth of the carbon nanotube. Then we repeat the process with another graphene sheet but aligning it with the other end of the carbon nanotube. Next, we will use the "Merge Structures" plugin [5]. With this plugin, we unify the three structures (the two sheets and the nanotube) into one. To duplicate the structure, we can move the entire new structure so that if we load the structure again, they do not overlap and use the "Merge Structures" plugin again. Finally, we will use the "Add solvatation Box" plugin [6] to add water molecules and simulate the water reservoirs, this plugin allows us to create a space where water molecules will be added to solvate

the structures found in that space. In our case, once the structures are solvated, we will have to use the TkConsole to select the water molecules between the graphene sheets and delete them.

One last detail, crucial for this simulation, is to define the beta factor of all the Carbon atoms as 1.00, as later we will need to make a distinction between the atoms we want to move and the ones we want to keep fixed.

2.2 Force Fields

The other necessary files for molecular dynamics simulations are the force field files. These files are extremely important for these simulations since they define how each element of the simulation will interact with the rest of the elements. In our case, we will use two force fields.

For water molecules, we will use the TIP3P water model. This water model defines water as a rigid structure in which the bonds do not vibrate or change their angles and gives a certain negative charge to the oxygen atom and a certain positive charge to the hydrogen. Although this may seem like too coarse a simulation to study the behavior of a few water molecules, in our case, where we want to study the behavior of many water molecules together, it is quite good. Furthermore, the charges of the atoms allow us to study water transport modulated by hydrogen bonds between water molecules.

For the Carbon atoms, we will use a reduced version of the CHARM22 force field in which we only define the behavior of the Carbon atoms, modeling the potentials as Lennard-Jones potentials. This reduced version is sufficient since we don't really want to study how water molecules (or other molecules) interact and react with Carbon or how the structure would behave over time, we want to do a more reduced simulation.

2.3 Simulations

We will perform two types of simulations, the first type will be an equilibration, and the other will be a simulation under non-equilibrium conditions where we will add pressure to the water molecules.

In the equilibration simulations, the objective is to simulate how the molecules of the system would behave under equilibrium conditions. The equilibrium conditions will be defined by each one and the program will seek to simulate in such a way that it finds the states of minimum energy. In this way, if the structure we have prepared has any kind of bonds or placements of the molecules that are very unlikely, these would evolve towards a state that would be more likely to occur spontaneously under the fixed conditions. In our case, these conditions are 300K temperature, atmospheric pressure, no force applied to the atoms, with the Carbon atoms fixed, and a simulation box of 34.4x17.01x80 Å in x,y, and z respectively. Additionally, we will keep pressure, temperature, and the number of particles constant during the simulation.

The other type of simulation, and the one that will really interest us, will be a simulation in which we will apply a force to all water molecules (actually only to the Oxygen atoms but, thanks to the TIP3P water model, this will be sufficient) in the z+ direction, so that it has the sense parallel to the carbon nanotubes and the molecules are pushed towards the mouth of the nanotube.

This simulation will use the coordinates of the atoms obtained in the equilibration simulation as the original coordinates, and, thanks to a tcl script that can be found in a tutorial to learn how to use namd [7] we will be able to modulate the pressure we want to apply to the system. In this script, we can define what force is applied to each particle and on which region this force will be applied. It should be noted that the force applied is defined in units of Kcal/mol·Å, so to obtain the pressure exerted on the system we will need to make the corresponding conversion factors, which, in this case, would be to multiply the force by 4.18410^{12} and the amount of water molecules on which the force will be exerted and then divide by the area perpendicular to the region where this force will be applied.

2.4 Analysis

To perform the analysis of these simulations, I have used the tcl scripts provided in the NAMD tutorials [7] with the modification made by Dr. Jordi Faraudo [8] on his GitHub where the permeation and water flux evolution is stored in each step of the simulation. After using these scripts in each of the simulations, I have written a Python code to represent the evolution of water permeation as a function of time elapsed and another one to represent the total water flux in each of the simulations. All these codes can be found in the GitHub repository where this work was found [9].

3 Results

The results I have obtained are quite surprising, actually. When analyzing the data, I expected to see how the water flux increased linearly, following the Hagen–Poiseuille equation [10], which establishes that the flux increases linearly with pressure.

Instead, what I have found, as we can see in figures (3) and (4), indicates that no matter how much we increase the pressure, both the water flux and the permeation of water molecules through the nanotubes do not seem to increase significantly, but rather they stabilize. As if we had reached a saturation limit.

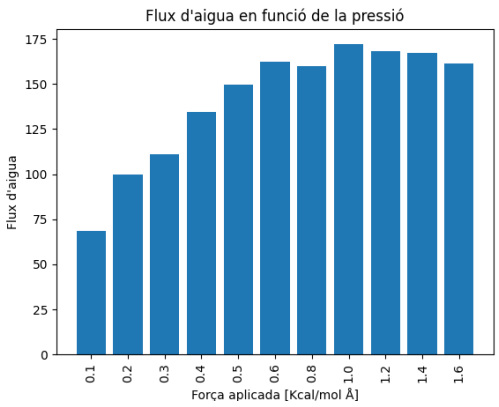

Figure 3: Water flux evolution

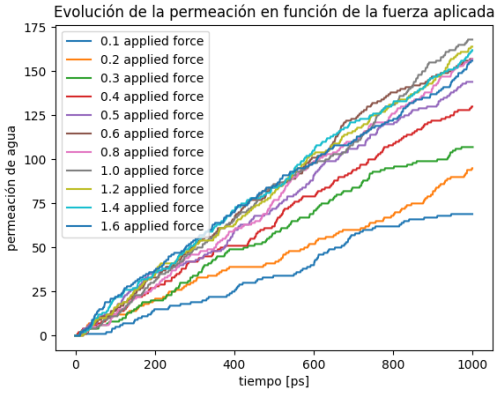

Figure 4: Water permeation evolution

4 Conclusions

My conclusion is that what I have found as results is not the saturation of the carbon nanotube channels, but I have found the maximum amount of water flux that can pass through the nanotube in the time elapsed during the simulation, since all of them have been executed for the same amount of time.

References

[1] T. W. Odom, J.-L. Huang, P. Kim, and C. M. Lieber, "Structure and electronic properties of carbon nanotubes," *The Journal of Physical Chemistry B*, vol. 104, pp. 2794–2809, April 2000.

[2] Theoretical and Computational Biophysics Group, "Namd." https://www.ks.uiuc.edu/Research/namd/, Fecha de acceso.

[3] Theoretical and Computational Biophysics Group, "Vmd." https://www.ks.uiuc.edu/Research/vmd/, Fecha de acceso.

[4] Theoretical and Computational Biophysics Group, "Nanotube plugin for vmd." http://www.ks.uiuc.edu/Research/vmd/plugins/nanotube/, Fecha de acceso.

[5] Theoretical and Computational Biophysics Group, "Mergestructs plugin for vmd." https://www.ks.uiuc.edu/Research/vmd/plugins/mergestructs/, Fecha de acceso.

[6] Theoretical and Computational Biophysics Group, "Solvate plugin for vmd." https://www.ks.uiuc.edu/Research/vmd/plugins/solvate/, Fecha de acceso.

[7] Theoretical and Computational Biophysics Group, "Namd tutorials." http://www.ks.uiuc.edu/Training/Tutorials/#namd, Fecha de acceso.

[8] jfaraudo, "Examples_namd repository." https://github.com/jfaraudo/Examples_NAMD.git, Fecha de acceso.

[9] DaniBedmar, "Treball_final repository." https://github.com/DaniBedmar/Treball_Final.git, Fecha de acceso.

[10] S. P. Sutera and R. Skalak, "The history of poiseuille's law," *Annual Review of Fluid Mechanics*, vol. 25, no. 1, pp. 1–20, 1993.