Instruction Manual

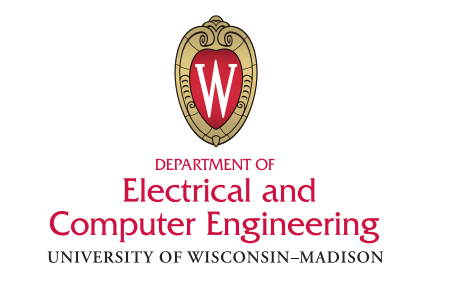
**Carbon Nanotube Mesh Generator**

Version: 1.01

Alexander Gabourie

Amirhossein Davoody

Prof. Irena Knezevic



Contents

[Introduction 2](#_Toc422832280)

[Usage 3](#_Toc422832281)

[Installing 3](#_Toc422832282)

[Initializing 4](#_Toc422832283)

[Configuration File 4](#_Toc422832284)

[Runtime 5](#_Toc422832285)

[Exit 5](#_Toc422832286)

[Simulation Mechanics 5](#_Toc422832287)

[Carbon Nanotube Parameters 5](#_Toc422832288)

[Bullet Physics Integration 5](#_Toc422832289)

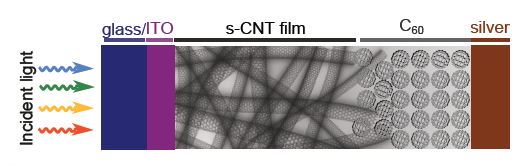
[Limitations 5](#_Toc422832290)

[References 5](#_Toc422832291)

# Introduction

The National Academy of Engineering, concerned about the increasing energy demands as well as the environmental cost of meeting them, has determined that development of an enconomic solar panel is a grand engineering challenge of the 21st century. To address the challenge, much effort has been towards finding a suitable light-absorbing material. In recent years, carbon nanotubes (CNTs) have become a material candidate due to more effective separation techniques. Researchers have the abilty to separate optically absorptive, semiconducting nanotubes from others. With a wide range of possible band gaps due to different nanotube chiralities, CNTs can be used to create widely tunable photovoltaic cells. The optical versatility, ultrafast exciton and charge transport, and chemical stability of CNTs has been the impetus of much recent research.

While research on bilayer donor/acceptor heterojunction photovoltaics has shown that exciton dissociation at the semiconducting CNT (s-CNT) and C60 junction [fig. 1] is very efficient, the diffusion of the exciton to the junction has been determined to be a bottleneck in increasing the overall process efficiency. Amirhossein Davoody, a member of Professor Knezevic’s research group, has undertaken a project to develop the theory of exciton diffusion in CNTs and use his theoretical framework to computationally simulate exciton dynamics in a CNT mesh. Before the start of the CNT Mesh Generator, he had already found success calculating exciton transfer rates between CNTs with arbitrary chirality, angle, and separation.



**Fig. 1** Structure of a CNT-based bilayer donor/acceptor heterojunction photovoltaic device

Since s-CNTs are separated from others through colloidal separation techniques and then deposited on substrates through spin coating, doctor-blade casting, or spray coating, there is no reason to believe that CNTs will have any sort of order when used on a device. For photovoltaics to become economical, the previously mentioned techniques will most likely be used to mass produce devices. If the production devices have unorded CNTs, then ultimately Amirhossein’s simulation should reflect that as accuratly as possible. Unfortunately, no software was found that could create the semi-realistic, random network of CNTs desired for the project.

The purpose of this software is to give a user the ability to choose CNT and device region parameters to create a random mesh of CNTs. Once the mesh is complete, each tube’s physical parameters are output to .csv file and are available for future processing. With respect to Amirhossein’s project’s needs, which was the motivation for this work, the final positions of the CNTs, in conjunction with the exciton tranfer rate tables, allow the modelling of exciton dynamics by using the Monte Carlo method.

This manual provides the imformation necessary to understand how to use the software and how the software works.

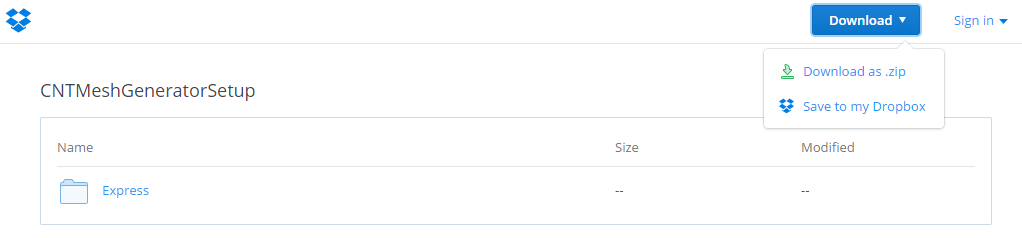
# Usage

## Installing

The current set-up for the software does not have a website for automatic updates. However, it does have a Dropbox link to the most recent deployed version of the software. The link is:

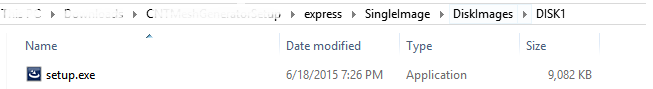
<https://www.dropbox.com/sh/x1aqk4ogpjp9gbe/AAAZMK3uj38gCj4krndCipg_a?dl=0>

After clicking on the link you will come to a screen that looks like the image below. Click on the “Download” button and select “Download as .zip.”



This will download all the necessary program files. Extract the .zip to a directory of your choosing. Open that directory. From that directory navigate to the following path:

.\express\SingleImage\DiskImages\DISK1



Click on “setup.exe” found at the location above and follow the on screen instructions to install. The installer may ask to download and install some Microsoft redistributables. Please do so as they are necessary components to the software.

Once the program is installed, it is important to be aware of the location of the CNT\_Mesh\_Config.xml file. It will initially be located in the program’s directory. This is an example xml file that is necessary to run the simulation. Feel free to move or copy the file. If you are to edit the file, only change the text that are part of the elements, do not change the elements themselves.

## Initializing

One can run the program using either the desktop shortcut or the command line. If running the software from the desktop shortcut, a window will appear and ask for the path of the configuration file. The configuration file is an xml file in the format of CNT\_Mesh\_Config.xml, a file that is included in the program directory. Type the full path to the desired configuration xml file using “/” in the path to denote directory changes. If running from the command line, there is the option to include the file path to the configuration xml as a command line argument. If no argument is provided, the user will get the same prompt as seen when using the desktop shortcut.

### Configuration File

The configuration file itself can be edited to change the simulation parameters and output locations. This section is focused on explaining each of the editable sections.

Acceptable inputs for units in the configuration files are:

-> mm, millimeter

-> um, micrometer

-> nm, nanometer

-> pm, picometer

-> A, angstrom

***outputDirectory:*** The folder in which the output files are to be saved. Enter a full file path

and end it with a “/”.

***numberTubes:*** The number of CNTs in the simulation. Must be an integer.

***friction:*** The amout of friction between surfaces. Due to non-realistic gravity, this is

non-physical, but can be tuned based on person preference. (Default: 1)

***gravity:*** The acceleration of gravity. Due to small device sizes, accurate gravity causes nanotubes

to move too quickly for simulation step sizes and will cause errors. Tune based on person preference, but default of -9.81 works well. (Default: -9.81)

***spacing:*** The minimum spacing between different nanotubes. For <min>, enter any positive

number. For units, refer to top of section.

***Lengths:*** The range of lengths that are acceptable for the simulation. <Lmin> is the minimum

length, <Lmax> is the maximum. <Lmin> must be less than <Lmax>. For units, refer to top of section.

***DeviceDimensions:*** The size of the box-like container that will be considered the device region.

The <xdim> and <ydim> elements are for the base of the region. The <zdim> element is for the height of the region. For units, refer to top of section.

***chirality:*** The chiralities that are acceptable in the simulation. The user is able to enter as many

<cnt> elements as desired as long as <m> and <n> are defined as part of the <cnt> children.

## Runtime

Once the configuration document is accepted by the program, two windows will open. The terminal window will pass along any information about program failures. The other window is the rendering window that shows the user where the simulation is currently at.

Since rendering the objects takes computational resources, the simulation initially starts with the window rendering nothing. This way if the software is called and visuals are not important, the simulation runs anywhere from 25%-50% faster. To turn on the visuals, refer to the top left of the rendering window. It will have instructions to push [shift + d] to toggle the debugDrawer, which will turn rendering on and off.

Overall there are many rendering options during runtime. The controls are displayed in the table below. The user is not required to give any imput to the simulation during runtime. Controls are for additional needs a user may have.

**Rendering Window Controls**

|  |  |
| --- | --- |
| Button | Action |
| a | Toggle axis-aligned boundary box |
| w | Toggle wire frame |
| Shift + d | Toggle rendering |
| Space | Restart simulation |
| l | Step camera left |
| r | Step camera right |
| f | Step camera forwards |
| b | Step camera backwards |
| z | Zoom in |
| x | Zoom out |
| g | Enable shadows |
| u | Enable textures |
| d | Disable deactivation (current program will not end if pressed!) |
| ctrl + left mouse | Pitch and yaw |
| ctrl + middle mouse | Pan |
| ctrl + right mouse | Zoom in/out |
| left mouse | Grab object |
| right mouse | Shoot box |
| o | Orthonormal view |
| +/- | Shoot box speed +/- 10 |

If the user decides to render the simulation they will be able to see colors on the rendered CNTs. The colors of each cylinder help indicate the overall progress of the simulation. The colors mean the following:

|  |  |  |  |
| --- | --- | --- | --- |
| Activation State of Cylinder | Normal | Wire Frame |  |
| Active | Purple | White |  |
| Wants Deactivation | Blue | Light Blue |  |
| Sleeping | Light Blue | Green |  |

These color states are used to determine if the nanotubes have stopped moving. If a pre-selected percentage of each CNT’s cylinders are wanting deactivation or are sleeping, the simulation will end.

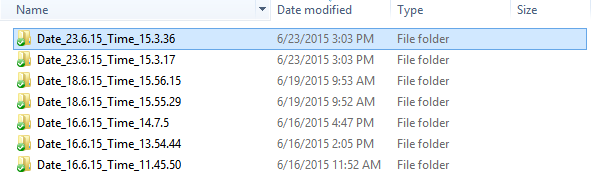
## Exit

- Output file location

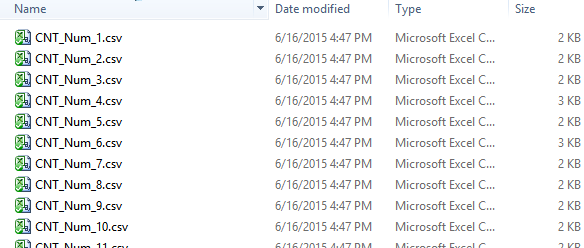
- output file format

- matlab script

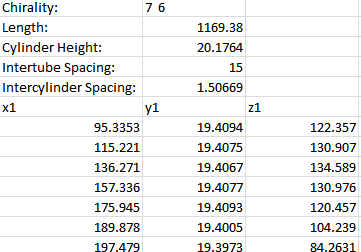
Once the simulation ends, each CNT that was in the simulation will have its own output file. These output files will be found in a time stamped directory found inside your output directory. An example of what your output directory may look like is provided below.



The files in each time stamped directory will look something like the example below.



The output files themselves look something like the following:



The output file format gives details about each tube. The chirality provides enough information for many physical and electrical calculations. The other parameters represent the physical parameters not captured by the CNT’s chirality. Most importantly, the positions of each compositional cylinder are provided. This position data is very important to the Monte Carlo method required for Amirhossein’s project described in the introduction.

In the next version of the software a MATLAB script will be included that can import the .csv files into a MATLAB workspace.

# Simulation Mechanics

- Basic overview of simulation components

## Carbon Nanotube Parameters

## Bullet Physics Integration

# Limitations

# References