My Awesome Adventures at Berkeley Lab

Diego Juarez

December 11, 2024

Abstract

Get ready to dive into my notes, updates, and all the most significant highlights from my journey through the prestigious halls of the Berkeley Lawrence National Laboratory.

Week 1: Downloading and Installation AMReX

AMReX a software framework designed to accelerate scientific discovery for applications solving partial differential equations on block-structured meshes.

Building Using CMake

```
git clone https://github.com/AMReX-Codes/amrex.git
cd amrex
mkdir build
cd build
cmake ..
cmake –build . -j8
```

Heat Equation (Tutorial Example)

In mathematics and physics, the heat equation is a certain **partial differential equation**. Solutions of the heat equation are sometimes known as caloric functions.
The compact form of the heat equation is expressed as:

$$\frac{\partial u}{\partial t} = \Delta u \tag{1}$$

Using AMReX we can construct a executable to solve numerically this equation.

Building Executable with GNU Make

```
git clone https://github.com/AMReX-Codes/amrex-tutorials.git
cd /workspace/amrex-tutorials/GuidedTutorials/HeatEquation
make
```

After you type all those commands in your terminal, you can use your executable to **compile** your **input file** typing:

./HeatEquation **input file**

Then if you followed these steps you will get your **output files** which are not human-readable so you can use the package for analyzing and visualizing **yt** in Python or also you can download the Visualization application Vislt.
For this problem the visualization is shown below:

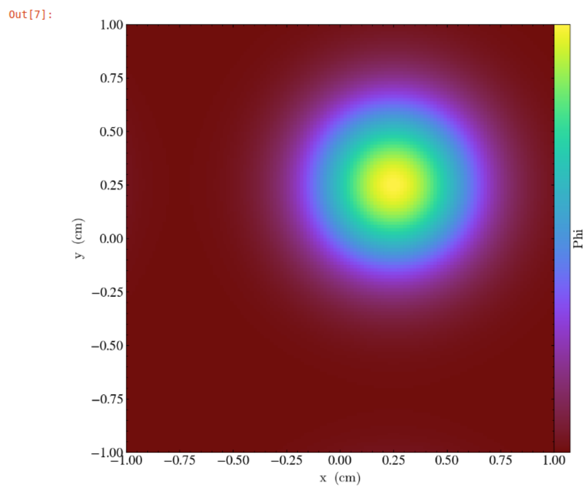


Figure 1: Heat Equation Plot Visualization (using yt)

yt-visualization (python-code)

```
import yt from yt.frontends.boxlib.data_structures
import AMReXDataset
ds = AMReXDataset("plt01000")
sl = yt.SlicePlot(ds, 2, ('boxlib', 'phi'))
sl
```

FHDeX

Inside of AMReX you can find a variety of codes developed each one with different task some of them are **FerroX**, **ARTEMIS**, **FHDeX** ando more that you can find reading the documentation: (https://amrex-codes.github.io), the last one mentioned FHDeX is an AMReX-enabled code libraries for stochastic hybrid models and algorithms for fluids, inside of it you can find one... working

Week 2-3: Phonon Detector

After we installed FHDeX we find a directory called PhononDetector where there is available the compiler to run and construct the executable to use Phonon-Detector. After compile it you will get a executable called main3d.gnu.x86-milan.MPI.ex, in the same directory you can find a Jupyter Notebook, which contain the assigned program to create the first of two **input files**, after run it you create a file called paramplanes.dat, the next input file is inside of the directory **test inputs** and is called input_detector. Finally to use PERLMUTTER to run the simulation you will need to create a Job script (run.sbatch). One suggestion of my professor is to create a linker to the executable to copy it in different directories when you want to run more than one simulation at the same time.

Creating the parameters of the plane

Inside of the Jupyter notebook there are some functions which at the end of the day create a file with some parameters which help to create the plane where will be the phonons created.
You can specify the **Temperature**, **Surface Roughness** and **Porosity** and also you can create figures inside of each block for example Cylinder as Figure 3.

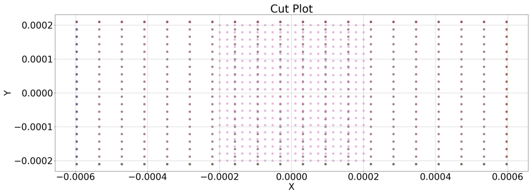


Figure 2: Plane

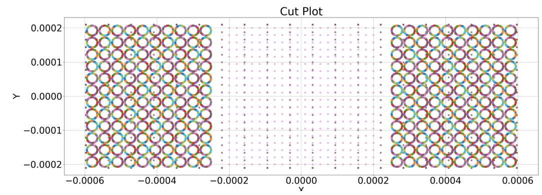


Figure 3: Plane with cylinders inside

Input Detector

Here you will specify the maximum numbers of time steps, scale of time steps, some constants and how often generate a plt file.

```
# Problem specification
 prob_lo = -0.000597375 -0.00020999999999999998 0.0        # physical lo coordinate
 prob_hi = 0.000597375 0.00020999999999999998 145e-7  # physical hi coordinate (cm)

 n_cells = 16 8 2 # keep as powers of two
 max_grid_size = 8 4 2
 max_particle_tile_size = 256 256 256

 # Time-step control
 fixed_dt = 1e-9

 # Controls for number of steps between actions
 max_step =     5000
 plot_int =     100
 n_steps_skip = -100

 reset_stats = 1
 restart  = -1
 chk_int  = -1

 particle_neff = 1

 #Species info
 #--------------
 nspecies     =  2
 phonon_sound_speed = 600000

 #Internal scattering parameters
 #-------------------
 tau_i = 2.95e45
 tau_ta = 9.3e13
 tau_la = 2.0e24
 toggleTimeFrac = 1


 # Stochastic parameters
 seed    = 1
 k_B     = 1.38064852e-16
 T_init  = 3
 h_bar   = 1.0546e-27
 variance_coef_mom = 1
```

Figure 4: Input Detector

Job script

Job script File

```
#!/bin/bash
#SBATCH -C gpu
#SBATCH -q shared
#SBATCH -t 00:30:00
#SBATCH -c 32
#SBATCH --gpus-per-task=1
#SBATCH -n 1
export SLURM_CPU_BIND="cores"
srun main3d.gnu.x86-milan.MPI.ex input_detector > output.txt
```

Simulations

To run a simulation we need 4 files:

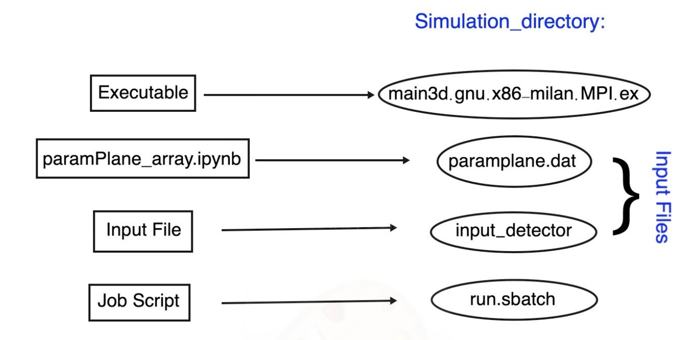


Figure 5: Diagram of the Files inside of each directory

You will receive the next three output files once the run is complete:

- Pltcu Files
- output.txt
- 1_particle_right_or_left_number_ts File

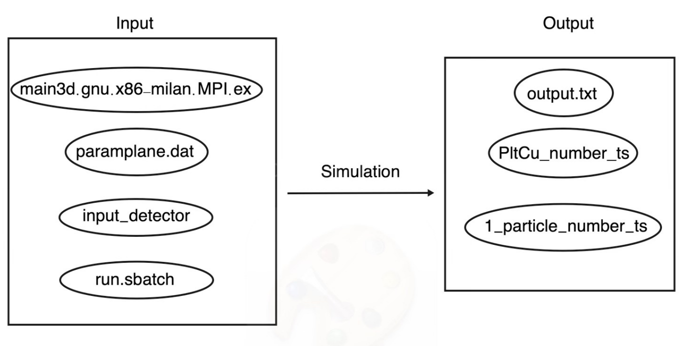


Figure 6: Procces of simulations

Let's see which physics properties can we obtain from these files
↘

Output.txt

Here we can find information about the system, for example how many phonons are inside of the walls (**Total Particles**), the specular fraction (**Fraction of boundary interactions which were specular**) and **Scattering Events**

```
Initializing AMReX (24.06-3-g0a691f6f4804-dirty)...
MPI initialized with 1 MPI processes
MPI initialized with thread support level 0
AMReX (24.06-3-g0a691f6f4804-dirty) initialized
Creating 7 parametric surfaces
Species 0 rhop 0
Species 0 Yk0 -nan
Species 0 rho0 1
Species 0 neff 1
Species 0 total -9223372036854775808
Species 0 count -9223372036854775808
Species 0 n0 -1.267638425e+30
Species 0 rho0 -0, from -nan
Species 1 rhop 0
Species 1 Yk0 -nan
Species 1 rho0 1
Species 1 neff 1
Species 1 total -9223372036854775808
Species 1 count -9223372036854775808
Species 1 n0 -1.267638425e+30
Species 1 rho0 -0, from -nan
Rho0: 1
Total n0: -1.267638425e+30
Initialization time = 0.003561987 seconds
Total particles: 6412
Internal scattering events: 12
Fraction of boundary interactions which were specular: 0.182745098
Advanced step 100 in 0.262502475 seconds
Total particles: 6396
Internal scattering events: 4
Fraction of boundary interactions which were specular: 0.1784708374
Advanced step 200 in 0.315252094 seconds
Total particles: 6225
Internal scattering events: 5
Fraction of boundary interactions which were specular: 0.1844393689
Advanced step 300 in 0.337100493 seconds
Total particles: 6393
Internal scattering events: 11
Fraction of boundary interactions which were specular: 0.1799418987
Advanced step 400 in 0.742793632 seconds
Total particles: 6436
Internal scattering events: 6
Fraction of boundary interactions which were specular: 0.1774963252
Advanced step 500 in 0.758425366 seconds
Total particles: 6320
Internal scattering events: 2
Fraction of boundary interactions which were specular: 0.1829527904
Advanced step 600 in 0.560165362 seconds
Total particles: 6358
Internal scattering events: 6
Fraction of boundary interactions which were specular: 0.1851939476
Advanced step 700 in 0.505713627 seconds
Total particles: 6234
Internal scattering events: 6
Fraction of boundary interactions which were specular: 0.1843924028
Advanced step 800 in 0.332100964 seconds
Total particles: 6383
Internal scattering events: 9
Fraction of boundary interactions which were specular: 0.1800059537
Advanced step 900 in 0.166334063 seconds
Total particles: 6411
```

Figure 7: Output File: Properties of the system

1_particle_right_or_left_number_ts File

In these files we have information about the Phonons that came out of the walls per time step.

Advanced Step (ts) = 100							
x	**y**	**z**	V_x	V_y	V_z	ω	τ
x_0	y_0	z_0	V_{x_0}	V_{y_0}	V_{z_0}	ω_0	τ_0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	y_n	z_n	V_{x_n}	V_{y_n}	V_{z_n}	ω_n	τ_n

Table 1: time step: 100

$\vdots$

Advanced Step (ts) = Max Step							
x	**y**	**z**	V_x	V_y	V_z	ω	τ
x_0	y_0	z_0	V_{x_0}	V_{y_0}	V_{z_0}	ω_0	τ_0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	y_n	z_n	V_{x_n}	V_{y_n}	V_{z_n}	ω_n	τ_n

Table 2: time step Max: Step

Finding the Steady State

One of my first job was find the steady state for the system varying properties as temperature and Roughness from the input file (paramplanes.dat), so i write a python code to run n Simulations increasing the temperature, deciding start with a small temperatures from $10 - 100K$ and analyzing with graphs when they reach the steady state.

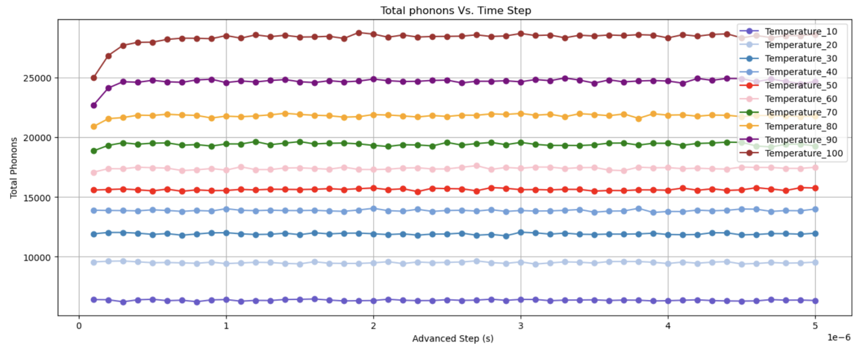


Figure 8: Total Phonons Vs. Time Step, Surface roughness 2×10^{-7}, Porosity 0.

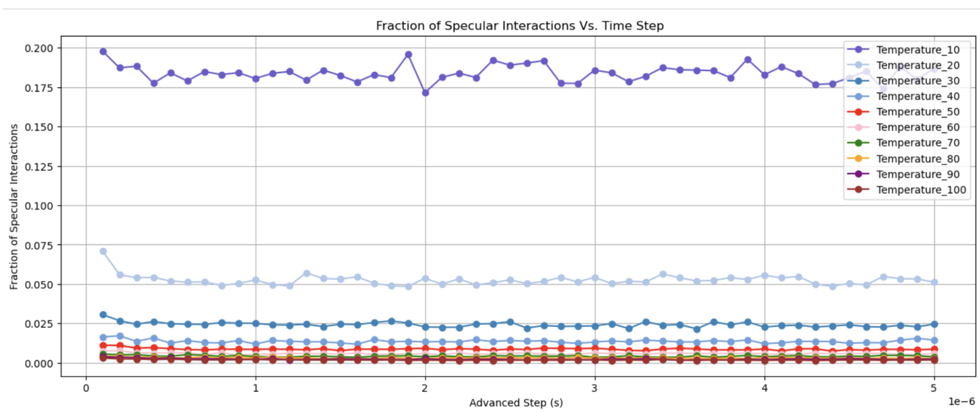


Figure 9: Fraction of Specular Interactions Vs. Time Step, Surface roughness 2×10^{-7}, Porosity 0.

Exit Angle (Angular Distribution)

When one of the phonons leaves the plane, the wall detects it and save their component velocities, with those values and using some trigonometry we can calculate the exit angle using:

$$\theta_{ex} = \tan^{-1}\left(\frac{V_y}{V_x}\right)$$

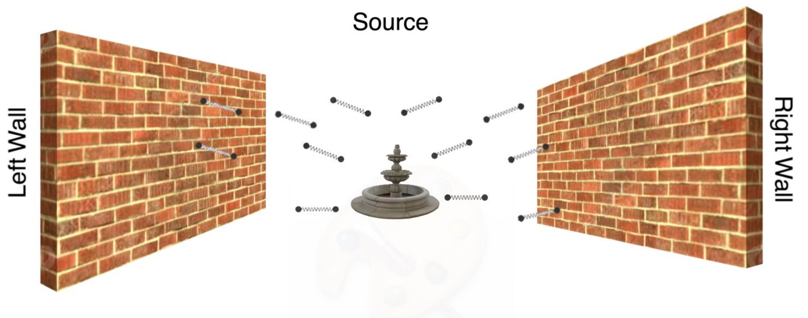


Figure 10: Mexican Cartoon of Phonon Detection; phonon detector represented as left and right walls, phonons as Little springs and Source as Mexican Fountain.

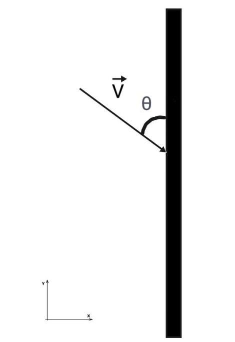


Figure 11: Phonon Hitting right wall

Simulation with Temperature $10K$, Roughness 2×10^{-7} and Porosity 0 (right wall):

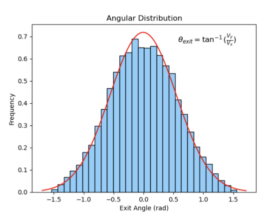


Figure 12: $ts = 0$

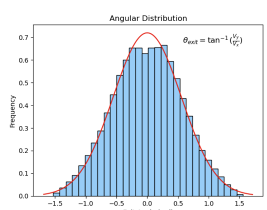


Figure 13: $ts = 500$

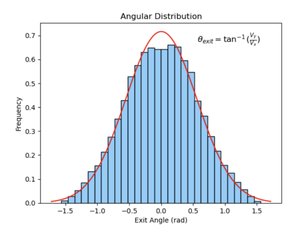


Figure 14: $ts = 5000$

Simulation with Temperature $100K$, Roughness 2×10^{-7} and Porosity 0 (right wall):

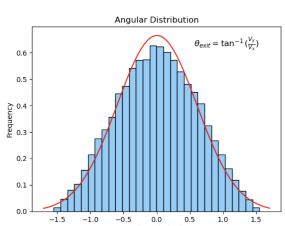


Figure 15: $ts = 0$

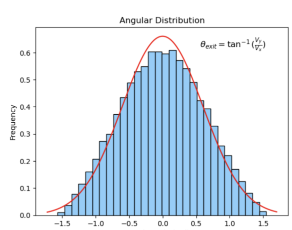


Figure 16: $ts = 500$

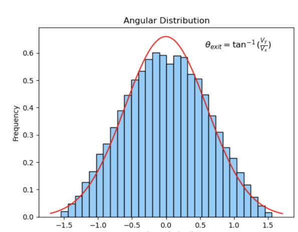


Figure 17: $ts = 5000$

Lifetime and Phonon Distribution

From the output file we can extract the **lifetime** and **phonons frequency**, which the lifetime correspond to the time that the phonons were inside of the walls, meanwhile the phonon frequency is the frequency of the phonons when they hit the wall. Simulation with Temperature $10K$, Roughness 2×10^{-7} and Porosity 0 (right wall):

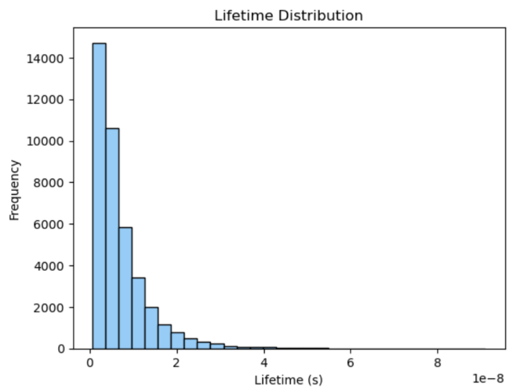


Figure 18: Lifetime Distribution Simulation 1

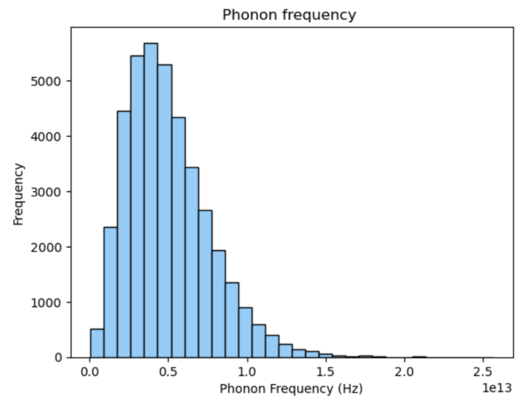


Figure 19: Phonon Frequency ω Simulation 1

Temperature $100K$, Roughness 2×10^{-7} and Porosity 0 (right wall):

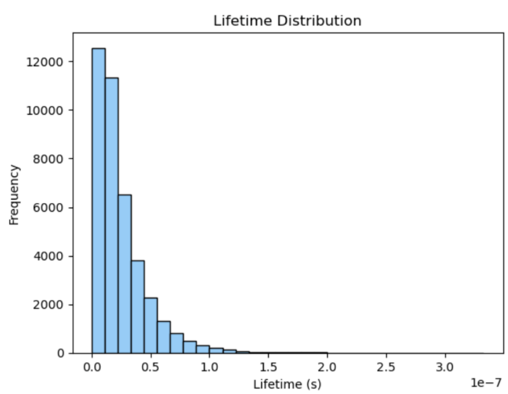


Figure 20: Lifetime Distribution Simulation 10

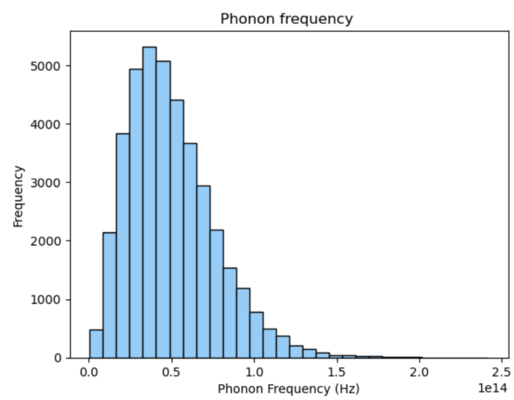


Figure 21: Phonon Frequency ω Simulation 10

Week 4

Overview of the experiment

'PhonoeXpose' (Name proposed by me c:): This is a Program designed to simulate the behavior of phonons and detection of them.

Phonon Generation

- **Source Plane**: The phonons are generated in a plane situated just above the system.
- **Phonons**: Can be thought of as quantized sound waves [1]

Phonon Transport (Travel)

- Once generated, phonons travel **ballistically**, meaning they move in straight lines without scattering.
- They can interact with walls, either getting reflected **diffusively** (scattering in many directions) or **specularly** (like a mirror reflection).

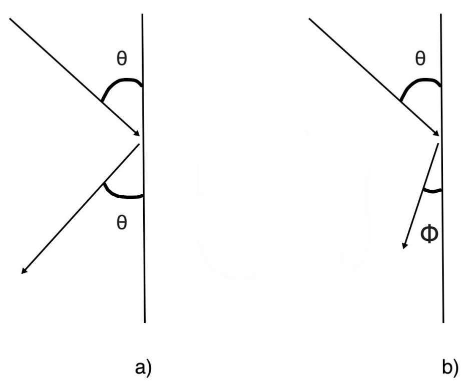


Figure 22: Reflection Example: **a)** Specular reflection, same incidence angle and reflected angle
b) Diffusive reflection, different incidence angle and reflected angle

Phonon Detection

- **Left and Right Walls**: They are acting as Detector and when a phonon hit them they record their frequencies, lifetime, positions and velocities, as you can see in table 1 and 2

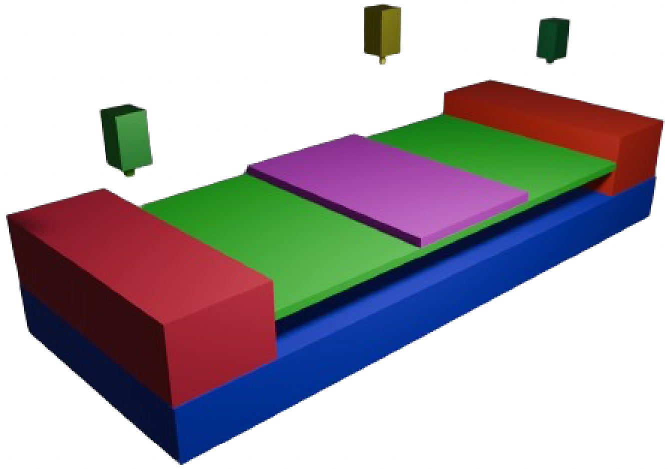


Figure 23: Ilustration of Phonon Detector Experiment

From this figure 26 we can identify three lasers: two metallic ones—green and yellow. The yellow laser emits light onto the pink pad, generating phonons that travel ballistically through the green plane. As these phonons move, they interact with the walls, either diffusively or specularly, before finally reaching one of the two red walls. The red walls, which are also targeted by the green metallic lasers, are responsible for detecting the impact of these phonons.

The probability of specular scattering

To record if a phonon is specular or not, the program define a conditional using the **The probability of specular scattering** which depends on the phonon wavelength (λ), root mean square surface roughness (η) and normal incidence angle (α) as:

$$P = \exp\left(-16\pi\eta^2 \frac{\cos^2\alpha}{\lambda^2}\right) \tag{2}$$

So from these expression we can extract important information and even verify the reliability of the program by changing certain parameters, such as roughness, to check if the program is running correctly. For instance easily we can see if we increase the surface roughness (η), we decrease the probability, i.e

$$\eta \longrightarrow \infty$$

$$p \longrightarrow 0$$

and vice-versa:

$$\eta \longrightarrow 0$$

$$p \longrightarrow 1$$

On the other hand as we are working with phonons as ballistic particles, if we increase the temperature we increase the wavelenght $T \propto \lambda$, therefore from the equation 2 we can see that:

$$T \longrightarrow \infty$$

$$\lambda \longrightarrow 0$$

$$p \longrightarrow 0$$

and vice-versa:

$$T \longrightarrow 0$$

$$\lambda \longrightarrow \infty$$

$$p \longrightarrow 1$$

Then we can arrange in a table an average of the specular fraction and compare with different simulations where i will change the Temperature and Roughness Surface.

Specular Fraction Average	
Simulation 1	$\langle p_1 \rangle$
Simulation 2	$\langle p_2 \rangle$
$\vdots$	$\vdots$
Simulation n	$\langle p_n \rangle$

Table 3: comparison specular fraction average

Temperature

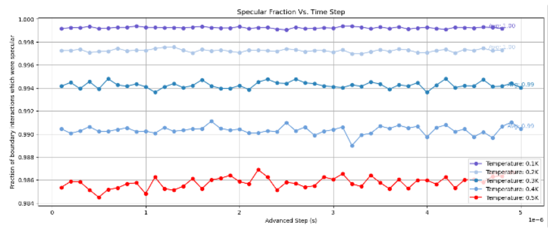


Figure 24: Specular Fractions Vs Time Step Temperature in range $(0.1K - 1K)$

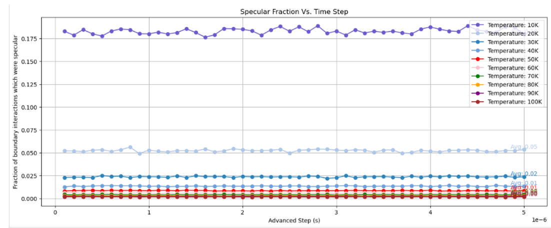


Figure 25: Specular Fractions Vs Time Step Temperature in range $(10K - 100K)$

Temperature K	**Specular Fraction Average**
0.1	0.99924
0.2	0.99725
0.3	0.99425
0.4	0.99033
0.5	0.98575
1	0.95214
10	0.1828
20	0.05225
30	0.02364
40	0.01343
50	0.00844
60	0.00574
70	0.00404
80	0.00292
90	0.00216
100	0.00162

Table 4: Temperature at Roughness $2 \times 10^{-7}m$

Roughness

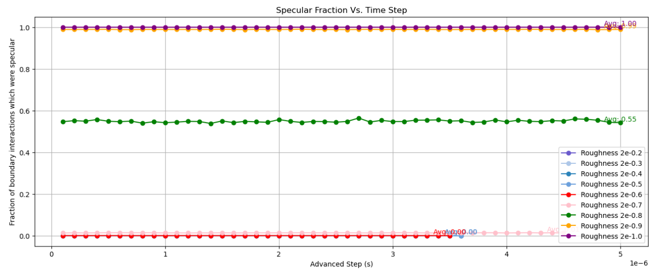


Figure 26: varying the Roughness

Roughness m	**Specular Fraction Average**
2×10^{-02}	0.00000
2×10^{-03}	0.00000
2×10^{-04}	0.00000
2×10^{-05}	0.00000
2×10^{-06}	0.00016
2×10^{-07}	0.01344
2×10^{-08}	0.54879
2×10^{-09}	0.98913
2×10^{-10}	0.99989

Table 5: Roughness at temperature 40 K

Source Plane Size

I decided to do a test and change the size of the source to see what effect it had on phonon generation. decreasing its area as you can see below:

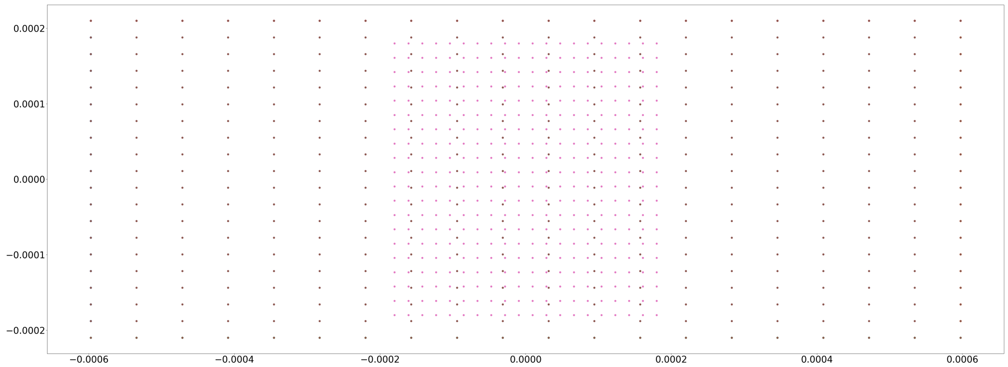


Figure 27: Decreasing 10%

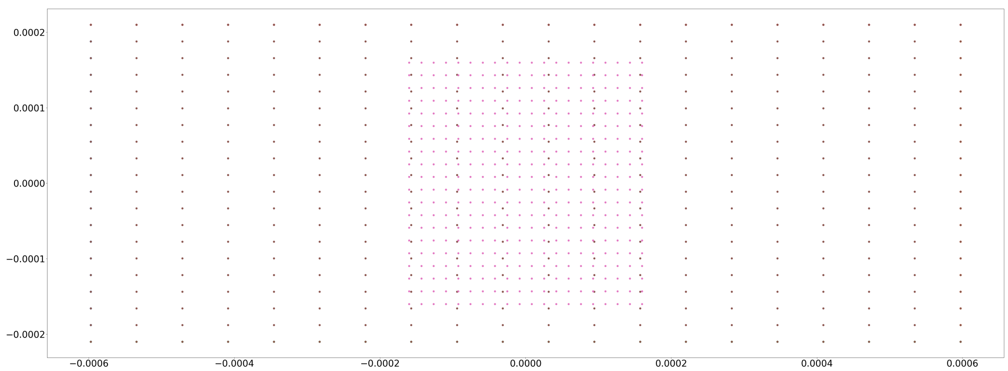


Figure 28: Decreasing 20%

⋮

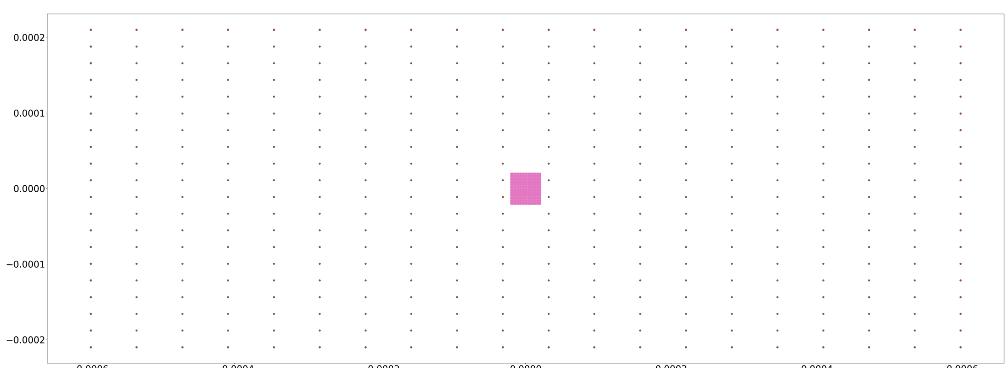


Figure 29: Decreasing 90%

And After this, plotting the graph of Phonons Vs. time step and Specular fraction Vs. time step:

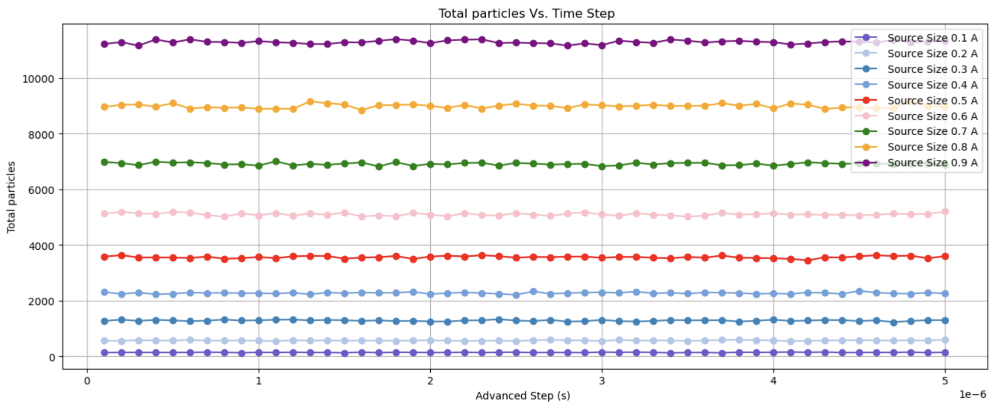


Figure 30: Phonon Vs. time step, Temperature $40K$ and Roughness Surface 2×10^{-7}

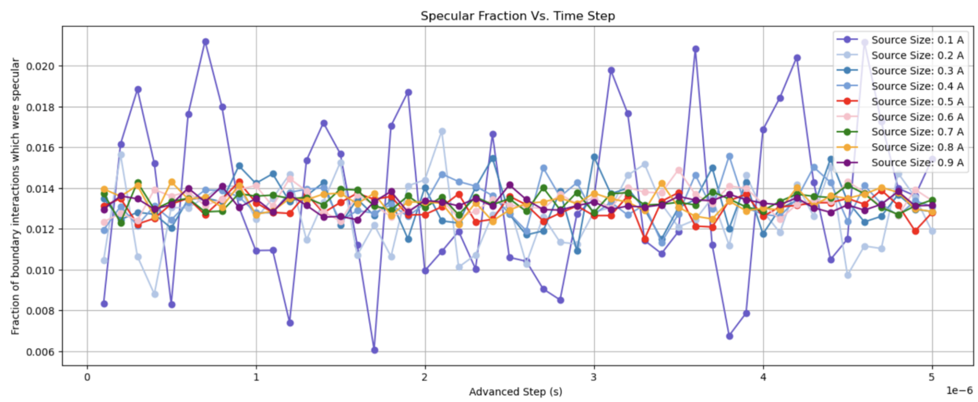


Figure 31: Specular fraction Vs. time step

Phonon Frequency Vs. Lifetime

From the ouput file labeled as **1_particle right_ or_left_number_ts** we can extract the frequency of the phonons (Hz) and lifetimes (time the phonon lasted before being detected).
If we plot the Frequency Vs. Lifetime of one simulation, we got something like this:

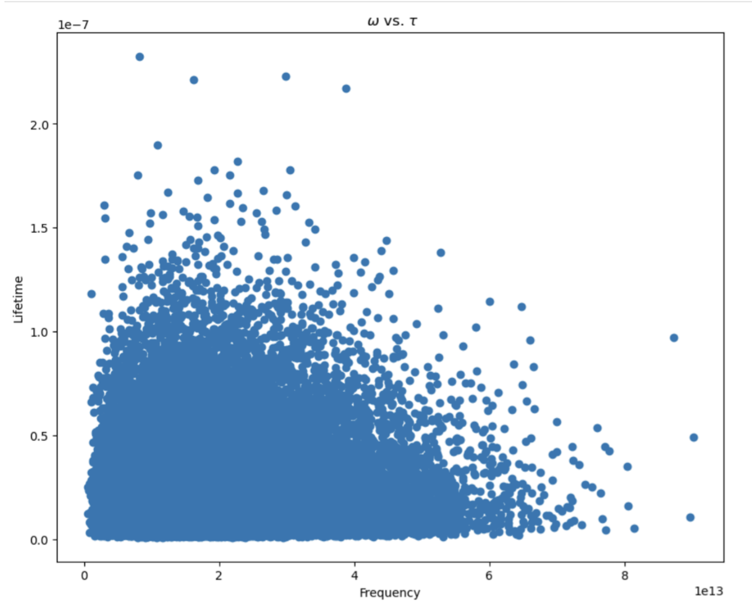


Figure 32: Frequency Vs. Lifetime, Temperature $40K$ and Roughness Surface 2×10^{-02}

We can calculate the average of a selected region of this graph

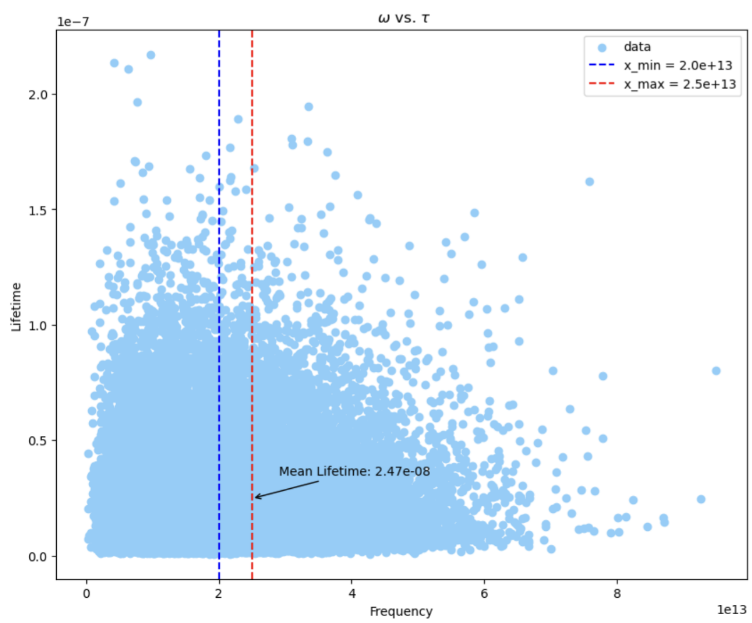


Figure 33: Frequency Vs. Lifetime, Temperature $40K$ and Roughness Surface 2×10^{-02}

With these stored values we want to calculate the standard error, which is expressed as:

$$E_{std} = \frac{\sigma}{\sqrt{n}}$$

And the standard deviation is expressed as:

$$\sigma = \sqrt{\frac{\Sigma(x_i - \mu)^2}{n}}$$

So with these equation and using some code we can plot the mean lifetime Vs Frequency

Comparison between two cases where the fraction of specular reflections is high and one where it is low

Surface Roughness: $2 \times 10^{-06} m$, Temperature: $40K$ and Specular fraction average: 0.00016 (**Low Specular fraction**)

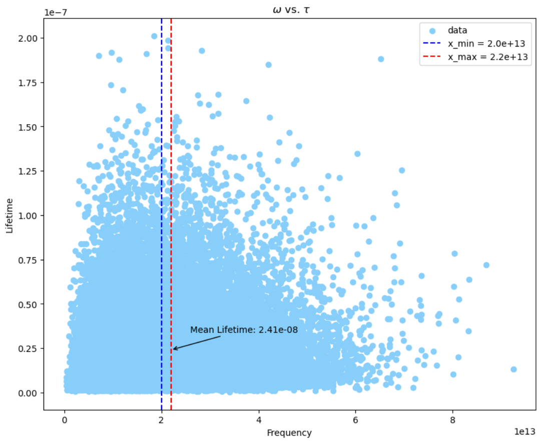


Figure 34: Range of 0.2×10^{14}

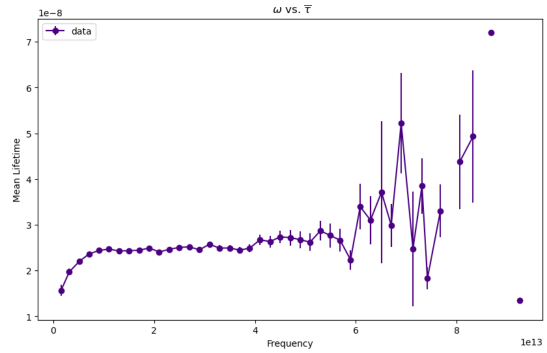


Figure 35: Frequency Vs Mean Lifetime, Surface Roughness: $2 \times 10^{-06} m$, Temperature: $40K$ and Specular fraction average: 0.00016

Surface Roughness: $2 \times 10^{-10} m$, Temperature: $40K$ and Specular fraction average: 0.99989 (**High Specular fraction**)

Temperature

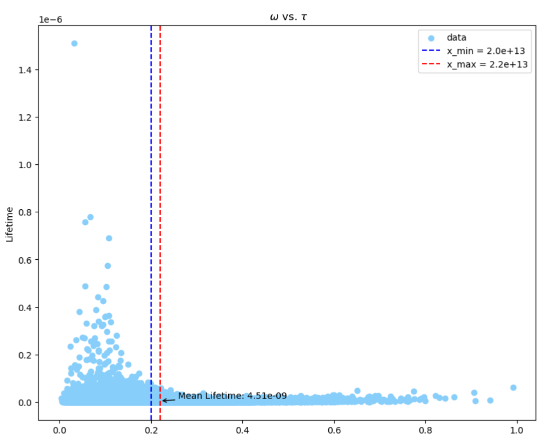


Figure 36: Range of 0.2×10^{14}

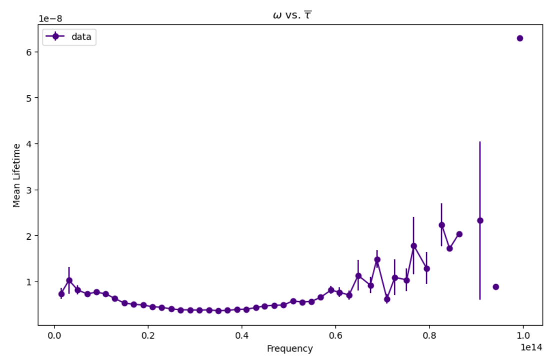


Figure 37: Frequency Vs Mean Lifetime, Surface Roughness: $2 \times 10^{-10} m$, Temperature: $40K$ and Specular fraction average: 0.99989

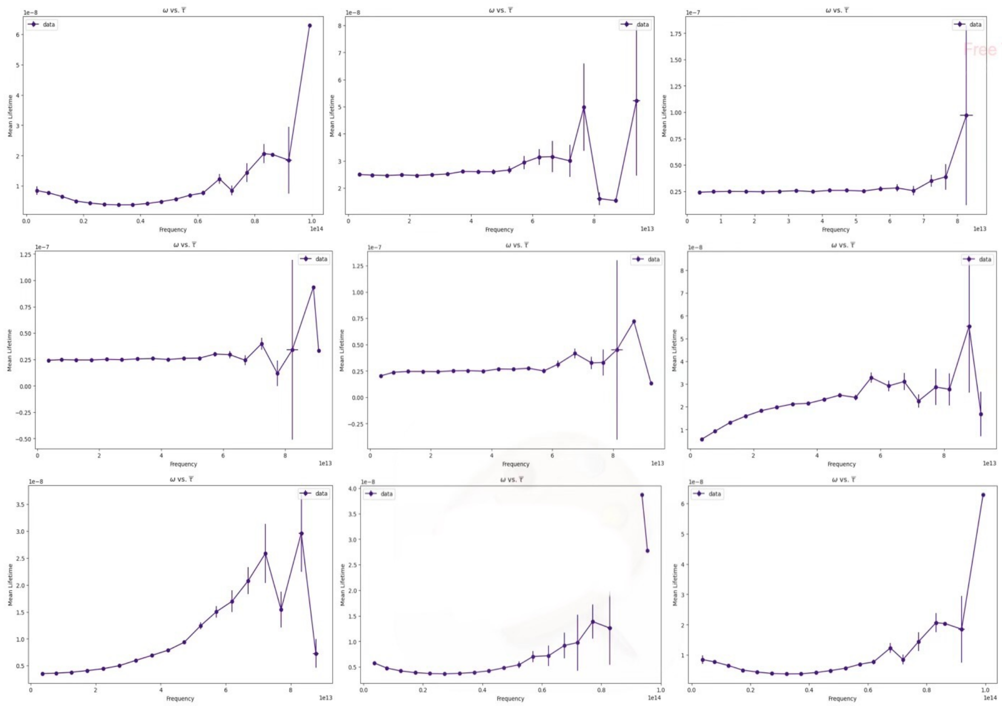


Figure 38: Frequency Vs. Mean Lifetime, Temperature $40K$ and Roughness Surface 2×10^{-02} All Simulation decreasing area of Source from $10\% - 90\%$

Week 5

Moving Source

The generation of phonons by dark matter interactions can occur at various positions, not just at the top of the plane. By altering the location of the phonon source, we can observe how these changes affect our plots.

Moving Source in Z-Axis

So you can see in the figure below how it was moved:

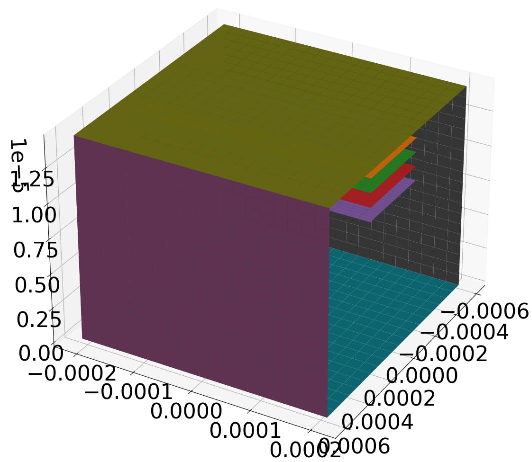


Figure 39: Along Z-Axis

Moving it at the center we can see how our plot for Exit angle change and Frequency Vs. Mean Lifetime, this simulations was at 40 K and Roughnes of 2×10^{-7};

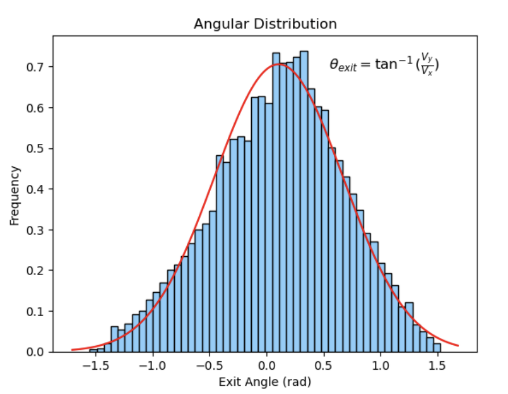


Figure 40: Exit Angle with Source of Phonons in the middle

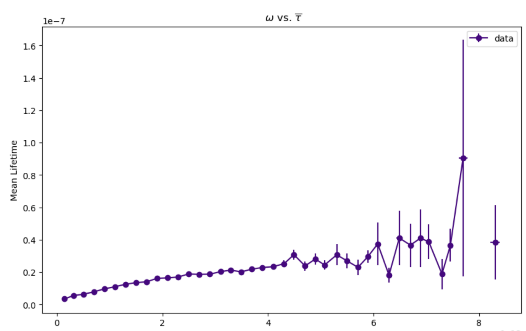


Figure 41: Frequency Vs. Mean Lifetime at 40 K and Roughness of 2×10^{-7}

Week 6

Moving Source Around Plane

Now that we are able to move the source wherever we want and run few simulation in distinct position of it, we can investigate how will be the behavior of the phonon detection.

Simulations

This simulations were performed at temperature 40 K, Roughness Surface 2×10^{-7}, plane source with an area of $0.05A_S$ which make it look as a puntual source source, 40 simulations where we map the diagonal from left to right (x-y plane) from up to down (x-z plane), as you can see in figure 42, More cores used (which make the simulations faster than before), 40 simulations were random generated and the other 40 with cosine Law Hemisphere which will be explained in next section **??**.

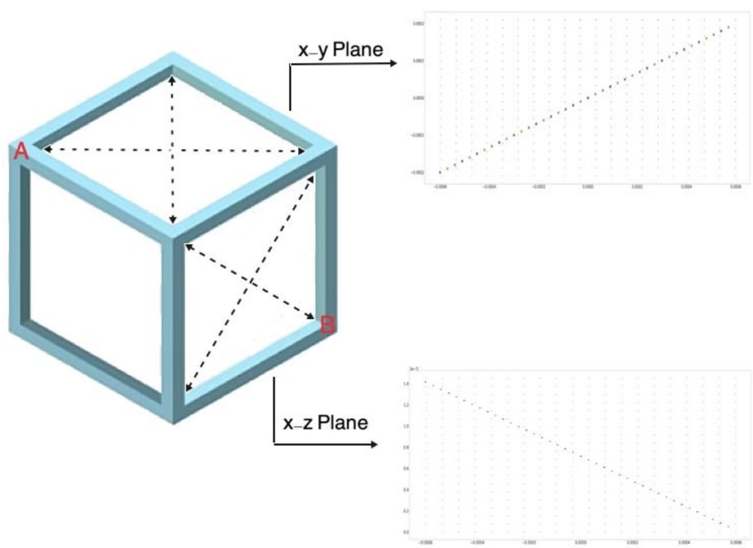


Figure 42: Moving Source Around the Diagonals of the plane diagram

Week 7

Phonon Generation

Simulations Parameters: 2000 time steps, Roughness Surface 2×10^{-7}, Temperature 40K

Simulation	Phonons Created	Phonons Exited	Percent Phonons Exit	Spec Frac Ave
1	264292	39000	14.75 %	0.031
2	291948	36000	12.33 %	0.030
3	225517	25000	11.08 %	0.029
4	217532	23000	10.57 %	0.031
5	61958	6000	9.68 %	0.033
6	129564	12000	9.26 %	0.033
7	167155	17000	10.17 %	0.033
8	102646	11000	10.71 %	0.034
9	33834	6000	17.73 %	0.028

Table 6:

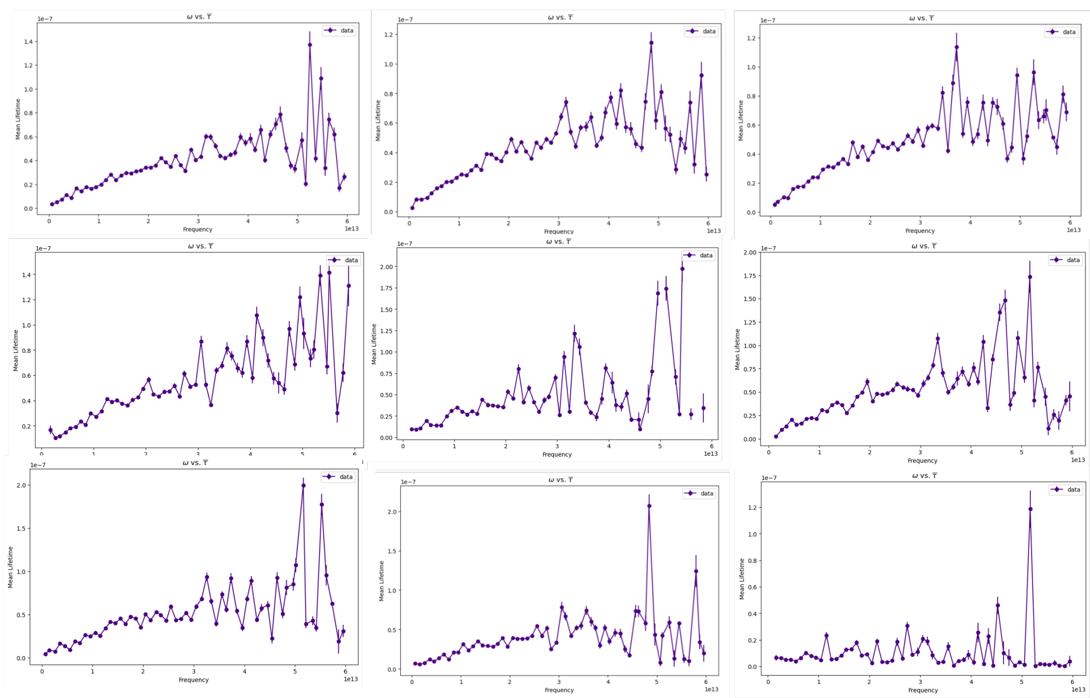
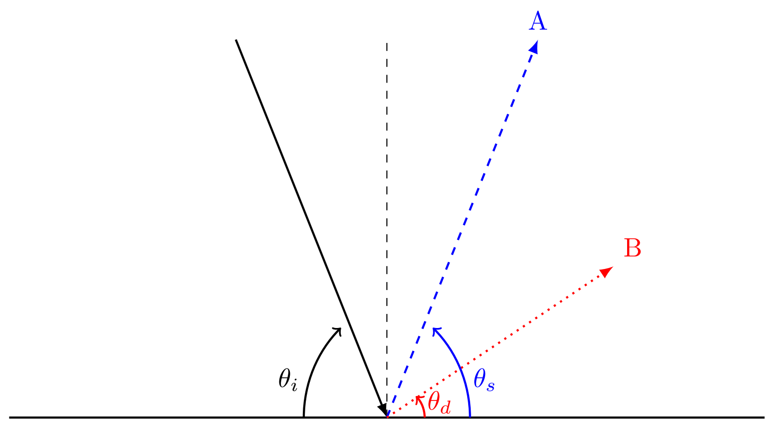


Figure 43: Moving Source from right to left, with wall detector in left


A
B
θ_i
θ_s
θ_d

References

[1] Girvin, Steven M.; Yang, Kun (2019). Modern Condensed Matter Physics. Cambridge University Press. pp. 78–96. ISBN 978-1-107-13739-4