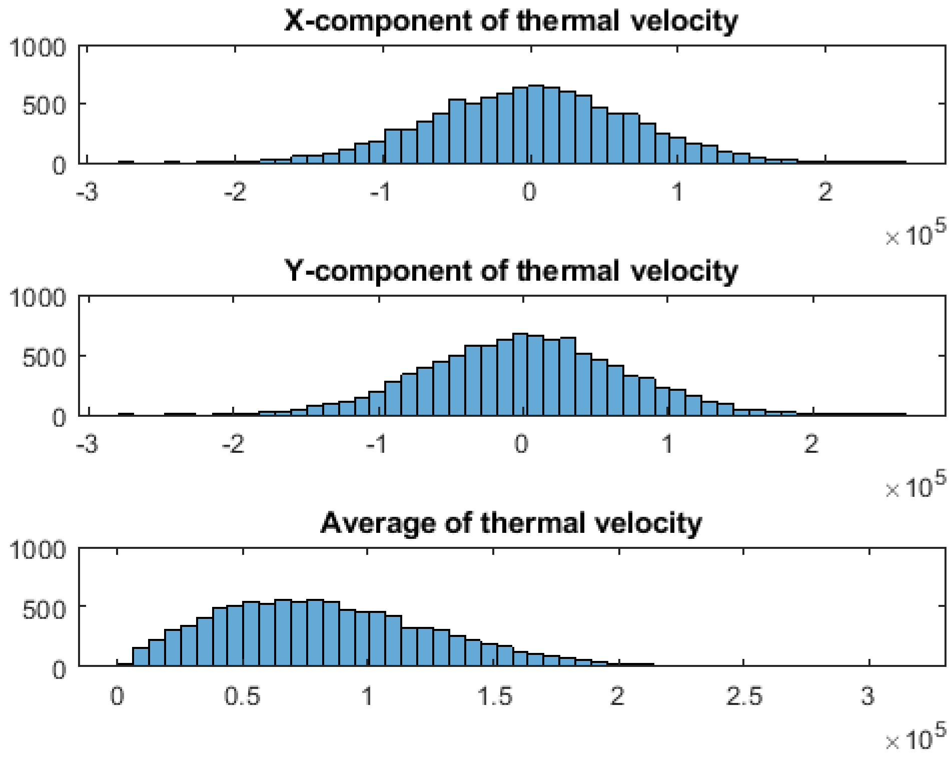
Part 2

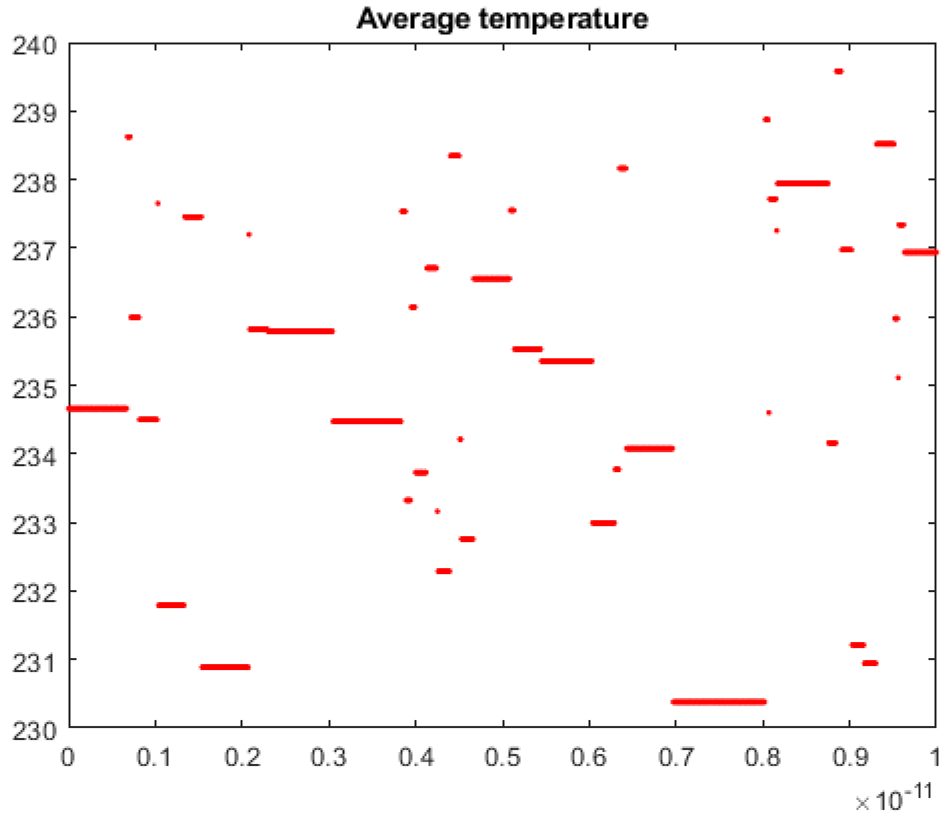
This part of the assignment required us to create a scatter probability for the electrons by using the equation $Psat = 1 - e^(-dt/tmn)$ with all the previous boundaries.

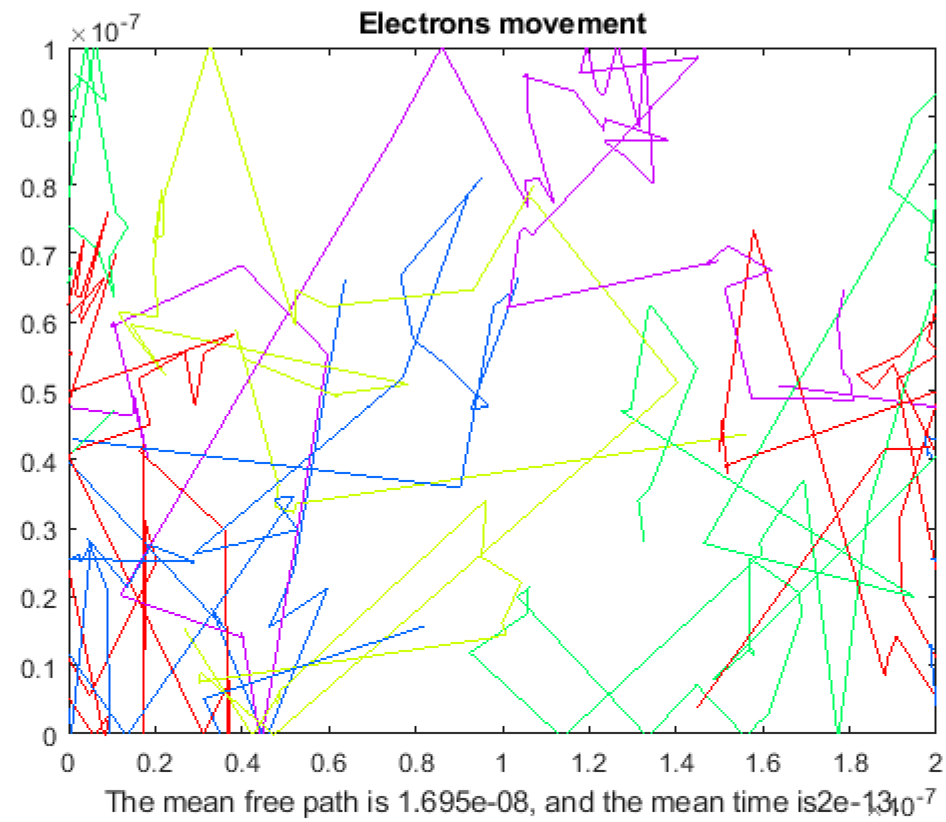
```
%Reset Everything
close all
clear

% Constant
q_0 = 1.60217653e-19;                    % electron charge
m_0 = 9.10938215e-31;                    % electron mass
kb = 1.3806504e-23;                      % Boltzmann constant
tmn = 0.2e-12;                           % mean time between collisions

% Region Defining
L = 200e-9;
W = 100e-9;

% Current Condition and variables
num = 1e4;                                  % Number of electrons
T = 300;                                    % Temperature (Kelvin)
vth_e = sqrt((2*kb*T)/(m_0));               % Thermal velocity of an
 electron
vth_ex = (vth_e/sqrt(2))*randn(num, 1);     % X-component of thermal
 velocity
vth_ey = (vth_e/sqrt(2))*randn(num, 1);     % Y-component of thermal
 velocity
vthdis = sqrt(vth_ex.^2+vth_ey.^2);         % Distribution of electron
 thermal velocity
vthav = mean(sqrt(vth_ex.^2+vth_ey.^2));    % Average of thermal
 velocity of all electrons
MFP = vthav*tmn;                            % Mean free path of
 electrons

% Histogram for thermal velocity
figure(1);
hold on
subplot(3, 1, 1);
histogram(vth_ex, 50)
title('X-component of thermal velocity');
subplot(3, 1, 2);
histogram(vth_ey, 50)
title('Y-component of thermal velocity');
subplot(3, 1, 3);
histogram(vthdis, 50)
title('Average of thermal velocity');

% Electrons Defining
Elec = zeros(num, 4);
Elec(:, 1) = L*rand(num, 1);
Elec(:, 2) = W*rand(num, 1);
```

```
Elec(:, 3) = vth_ex;
Elec(:, 4) = vth_ey;
previous = zeros(num, 4);
previous = Elec;

% Electron simulation
t = 1e-11;                          % Total Time
dt = 1e-14;                         % Time Step
Psat = 1 - exp(-dt/tmn);            % Exponential Scattering
 Probability
numplot = 5;                        % Number of electron plotted
color = hsv(numplot);               % Colour Setup

for n = 0:dt:t

    % Part 2 Simulation
    if Psat > rand()
        vth_ex = (vth_e/sqrt(2))*randn(num, 1);
        vth_ey = (vth_e/sqrt(2))*randn(num, 1);
        Elec(:, 3) = vth_ex;
        Elec(:, 4) = vth_ey;
    end

    for p = 1:1:num
        previous(p, 1) = Elec(p, 1);
        previous(p, 2) = Elec(p, 2);
        Elec(p, 1) = Elec(p, 1)+ Elec(p, 3)*dt;
        Elec(p, 2) = Elec(p, 2)+ Elec(p, 4)*dt;
    end

    % Plotting limited amount of electrons
    figure(2)
    for q = 1:1:numplot
        title('Electrons movement');
        plot([previous(q, 1), Elec(q, 1)], [previous(q, 2),
 Elec(q,2)], 'color', color(q, :))
        xlim([0 L])
        ylim([0 W])
        hold on
    end

    % Setting up the boundaries
    for o = 1:1:num
        % Looping on x-axis
        if Elec(o, 1) > L
            Elec(o, 1) = Elec(o, 1) - L;
            previous = Elec;
        end
        if Elec(o, 1) < 0
            Elec(o, 1) = Elec(o, 1) + L;
            previous = Elec;
        end
        % Reflecting on y-axis
        if Elec(o, 2) > W || Elec(o, 2) < 0
```

```
            Elec(o, 4) = -1*Elec(o, 4);
        end
    end

    % Plotting average temperature
    vthav = mean(sqrt(vth_ex.^2+vth_ey.^2));   % Average of thermal
 velocity of all electrons
    aveT = (0.5*m_0*vthav^2)/kb;               % Average temperature
    figure(3)
    plot(n, aveT, 'r.')
    title('Average temperature');
    hold on
end

vthav = mean(sqrt(vth_ex.^2+vth_ey.^2));    % Average of thermal
 velocity of all electrons
MFP = vthav*tmn;                            % Mean free path of
 electrons
figure(2)
xlabel(['The mean free path is ' num2str(MFP) ', and the mean time is'
 num2str(tmn) '.']);
```




Average temperature
240
239
238
237
236
235
234
233
232
231
230
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
×10^-11


Electrons movement
×10^-7
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
×10^-7
The mean free path is 1.695e-08, and the mean time is2e-13

Published with MATLAB® R2017b