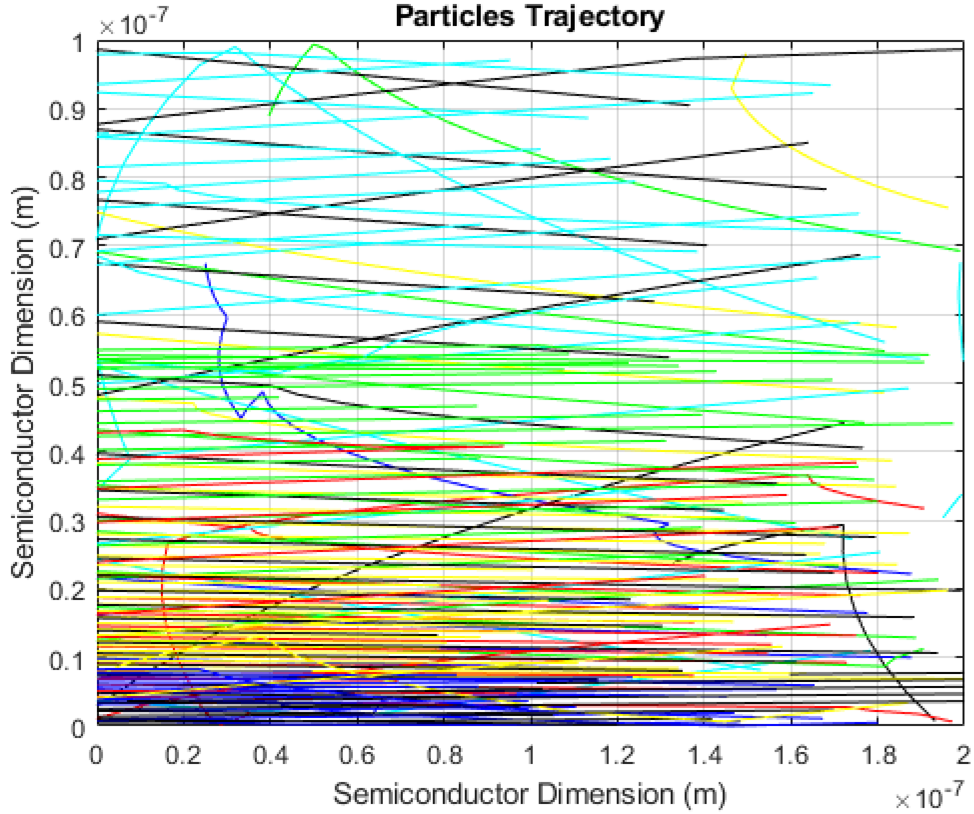
```
% ELEC4700 - Assignment 3 part 1
% Xiaochen Xin 100989338
C.q_0 = 1.60217653e-19;            % electron charge
C.hb = 1.054571596e-34;            % Dirac constant
C.h = C.hb * 2 * pi;                   % Planck constant
C.m_0 = 9.10938215e-31;            % electron mass(kg)
C.kb = 1.3806504e-23;              % Boltzmann constant
C.eps_0 = 8.854187817e-12;         % vacuum permittivity
C.mu_0 = 1.2566370614e-6;          % vacuum permeability
C.c = 299792458;                   % speed of light
C.g = 9.80665; %metres (32.1740 ft) per s²
C.am = 1.66053892e-27;


mn = 0.26*C.m_0; %Effective Mass
l = 200e-9; %Length of area (m)
w = 100e-9; %Width of area (m)
T = 300; %Kelvin

vth = sqrt(C.kb*T/mn);%thermal velocity(velocity at which the
 particles are travelling at)

tmn = 0.2e-12; %mean time between collision (s)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
numOfAtom = 10;
numOfStep = 100;
rvx = randn(numOfAtom,1)*sqrt(C.kb*T/mn); %random vx
rvy = randn(numOfAtom,1)*sqrt(C.kb*T/mn); %random vY

v   = sqrt(rvx.^2+rvy.^2);
% figure (1)
% plot(hist(v,100))
% ylabel("Number of Particles")
% xlabel("Bins")
% title ("Particle Velocity Distribution")

xr = 200e-9.*rand(numOfAtom,1); %x of 100 random locations
yr = 100e-9.*rand(numOfAtom,1); %y of 100 random locations

%Define two arrays store the previous locations
xrp = xr;
yrp = yr;

MFPx = xr;
MFPy = yr;
MFP  = zeros(numOfAtom,1);
MTBC = zeros(numOfAtom,1);
scatter_number = zeros(numOfAtom,1);

t = 1.5e-14; %time interval that captures line
```

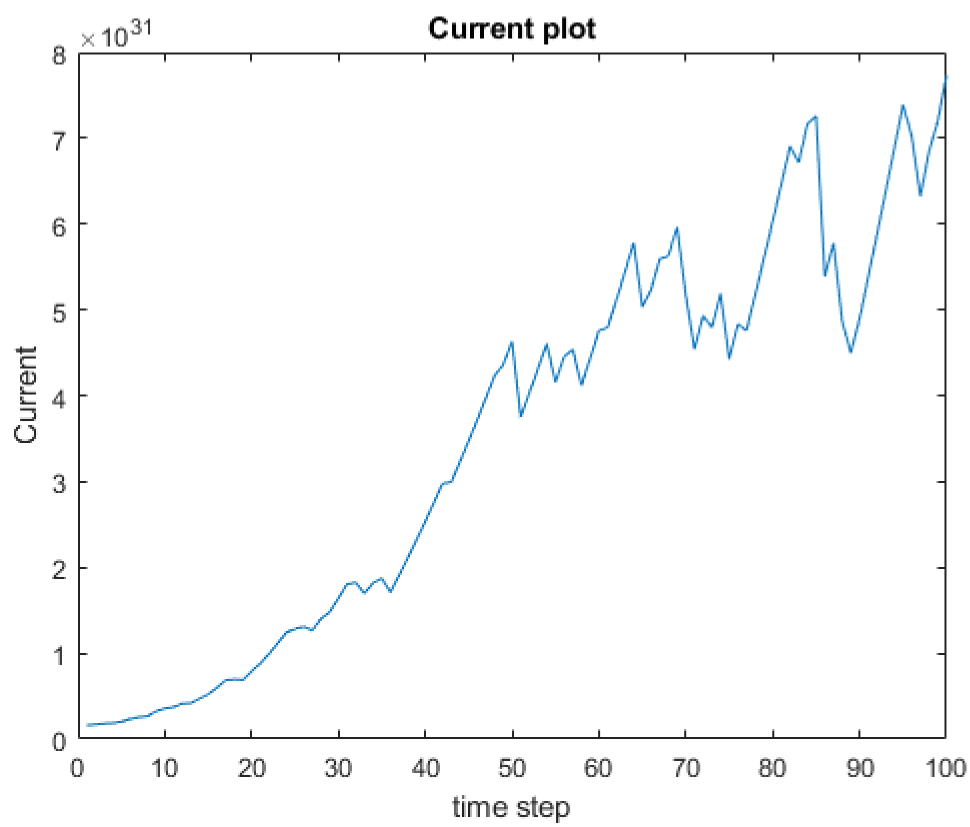
```
xd = rvx*t; %displacement in x during one time interval
yd = rvy*t; %displacement in y during one time interval

Pscat = 1-exp(-t/tmn); % Probability that a particle scatters

%Additional code for Assignment 3

spaceStep = 0.01*w;
dt = spaceStep/vth;
Vleft = 0.1;%voltage of left side
electronConc = 10e15;
Efield = Vleft/l;
force = Efield*C.q_0;
disp('Electric Field:')
disp(Efield)
disp('Force on each electron:')
disp(force)
acceleration = force/mn;
disp('Acceleration')
disp(acceleration)
disp('current = q*n*mu*E/area')
t = 1.5e-14; %time interval that captures line


currentHistory = zeros(1,numOfStep);

for p = 1:1:numOfStep
    scatter_prob = rand(numOfAtom,1);
%%%%%%%%%Calculate Mean Free Path%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    MFP(scatter_prob<Pscat) = MFP(scatter_prob<Pscat) +
 sqrt((xd(scatter_prob<Pscat)-
MFPx(scatter_prob<Pscat)).^2+(yd(scatter_prob<Pscat)-
MFPy(scatter_prob<Pscat)).^2);
    MTBC(scatter_prob<Pscat) =  MTBC(scatter_prob<Pscat) +
 sqrt((xd(scatter_prob<Pscat)-
MFPx(scatter_prob<Pscat)).^2+(yd(scatter_prob<Pscat)-
MFPy(scatter_prob<Pscat)).^2)./v(scatter_prob<Pscat);
    scatter_number(scatter_prob<Pscat) =
 scatter_number(scatter_prob<Pscat) + 1;
    MFPx(scatter_prob<Pscat) = xr(scatter_prob<Pscat);
    MFPy(scatter_prob<Pscat) = yr(scatter_prob<Pscat);
%%%%%%%%%Calculate average temperature of all particles%%%%%%%%%%%%%
%%%%%%%%%%%%%%%
    v = sqrt(rvx.^2+rvy.^2);
    TParticles = (0.5*mn*v.^2)/(C.kb);%Tempearture of individual
 particles
    Tave (p) = sum(TParticles)/numOfAtom;%Average temperature of all
 particles
    figure (3)
    plot (Tave)
    xlabel("Number of steps (1.5e-14s/step)")
    ylabel("Temperature (K)")
    ylim ([0, 500])
    xlim ([0, numOfStep])
```

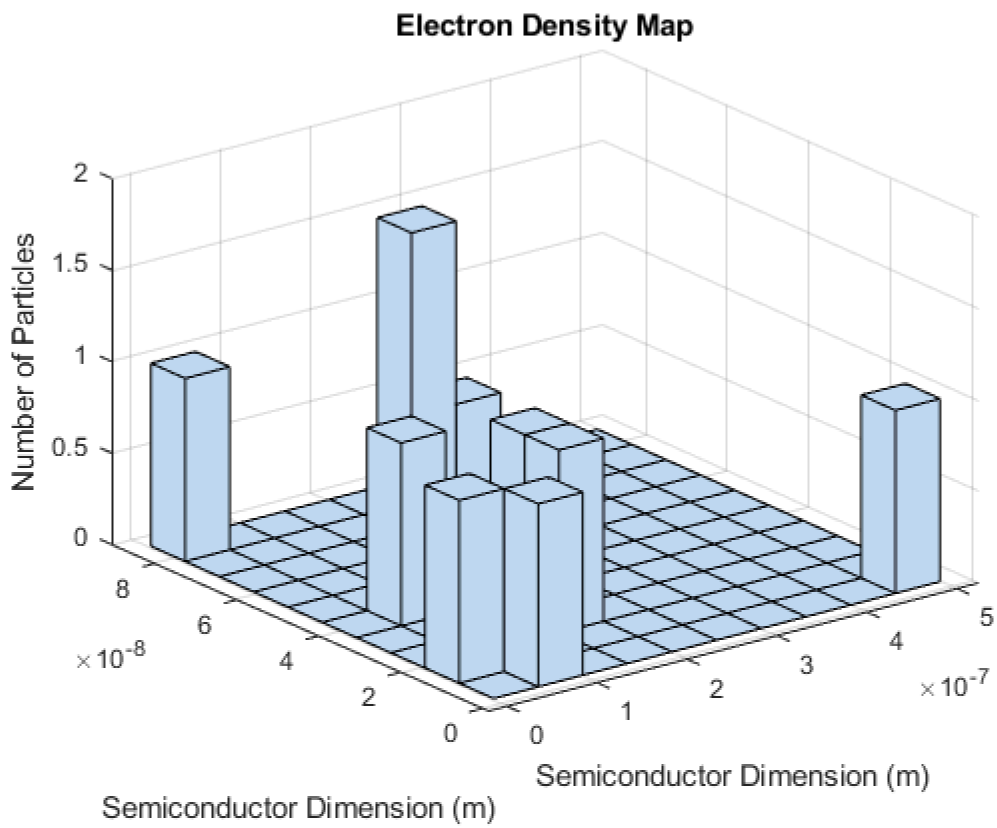
```
    title("average temperature over time")
%%%%%%%%%%Calculate Mean Time Between Collision%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%

    rvx_new = randn(numOfAtom,1)*sqrt(C.kb*T/mn); %new random vx
    rvy_new = randn(numOfAtom,1)*sqrt(C.kb*T/mn); %new random vY
    rvx(scatter_prob<Pscat) = rvx_new(scatter_prob<Pscat);
    rvy(scatter_prob<Pscat) = rvy_new(scatter_prob<Pscat);

    rvx = rvx+p*t*acceleration;
    xd = rvx*t+0.5*acceleration*t^2; %displacement in x during one
 time interval
    yd = rvy*t; %displacement in y during one time interval

    xr = xr+xd;
    yr = yr+yd;

    tvelocity = sqrt((rvx/dt).^2 +(rvy/dt).^2);
    %Define the left&right wrap-around
    xrp(xr >2e-7) = - (2e-7 - xrp(xr >2e-7));% changing previous point
 to prevent line drawn across canvas
    xr(xr > 2e-7) = xr(xr > 2e-7)-(2e-7);
    xrp(xr <0)    = 2e-7 - xrp(xr <0);% changing previous point to
 prevent line drawn across canvas
    xr(xr < 0)    = xr(xr < 0 )+(2e-7);

    %Define the specular top&bottom
    rvy(yr > 1e-7) = - rvy(yr > 1e-7);
    yr(yr > 1e-7) = (1e-7)-(yr(yr > 1e-7)-(1e-7));
    rvy(yr < 0) = -rvy(yr < 0 );
    yr(yr < 0) = -yr(yr < 0 );

    figure (2)
    plot([xrp(1), xr(1)], [yrp(1), yr(1)], 'r')
    plot([xrp(2), xr(2)], [yrp(2), yr(2)], 'b')
    plot([xrp(3), xr(3)], [yrp(3), yr(3)], 'k')
    plot([xrp(4), xr(4)], [yrp(4), yr(4)], 'g')
    plot([xrp(5), xr(5)], [yrp(5), yr(5)], 'y')
    plot([xrp(6), xr(6)], [yrp(6), yr(6)], 'c')
    xlabel("Semiconductor Dimension (m)")
    ylabel("Semiconductor Dimension (m)")
    title ("Particles Trajectory")
    xlim ([0, 2e-7])
    ylim([0,1e-7])
    grid on
    hold on
    pause(0.05)

    %current tracking
    avgVel=sum(tvelocity)/numOfAtom;
    mu = (avgVel)/Efield;
    currentHistory(p) =C.q_0*electronConc*mu*Efield/(w*l);

    xrp = xr;
```

```
    yrp = yr;
end
% %Display Overall MFP
% overallMFP = sum(MFP./scatter_number)/numOfAtom
% overallMTBC = sum(MTBC./scatter_number)/numOfAtom

figure(3)
plot(linspace(1,numOfStep,numOfStep),currentHistory)
title('Current plot')
xlabel('time step')
ylabel('Current')

xy = [xr,yr];
figure (4)
hist3(xy)
title ('Electron Density Map')
xlabel("Semiconductor Dimension (m)")
ylabel("Semiconductor Dimension (m)")
zlabel("Number of Particles")
```

```
Electric Field:
   5.0000e+05

Force on each electron:
   8.0109e-14

Acceleration
   3.3823e+17

current = q*n*mu*E/area
```


Particles Trajectory
×10^-7
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Semiconductor Dimension (m)
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
Semiconductor Dimension (m)
×10^-7


Current plot
×10^31
8
7
6
5
4
3
2
1
0
Current
0
10
20
30
40
50
60
70
80
90
100
time step



Published with MATLAB® R2018b