Waves and Optics

Mini Project | Computational Physics (PHY4602) | Semester 1 2023/2024

Chapter: Waves and Optics

Group Members

1. Muhammad Azren Shah Bin Mohd Halim Shah (206069)
2. Amin Sufi Bin Yus Mohd Al Munnawar (206473)
3. Alif Azraie Rizal Bin Mohd Nasudin (208190)
4. Wan Muhammad Muizzuddin bin Wan Mazlan (207737)

Table of Contents

Introduction

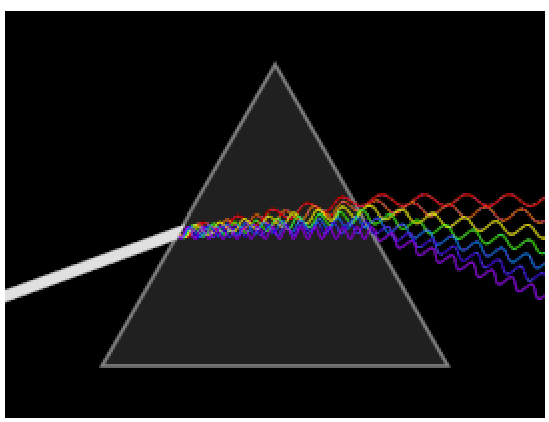


Figure 1: Schematic of light being dispersed by a prism.. Source: Wikipedia

In the realm of **waves** and **optics**, as a branch of physics, focuses on unraveling the behavior and characteristics of light. Understanding light as an electromagnetic wave connects optics to the broader study of waves. Optics delves into how light interacts with matter and the creation of instruments for its use or detection. This exploration expands to encompass visible, ultraviolet, and infrared light. Importantly, acknowledging light as an electromagnetic wave establishes a link with other forms of electromagnetic radiation, such as X-rays, microwaves, and radio waves, showcasing shared properties across this spectrum of waves

To optimize the reading experience, the following toolboxes are required:

1. Statistical and Machine Learning Toolbox
2. Signal Processing Toolbox

Types of Waves and Optics

Types of Waves:

1. Electromagnetic Waves:

- Include visible light, ultraviolet, infrared, X-rays, microwaves, and radio waves.
- Characterized by oscillating electric and magnetic fields.

2. Sound Waves:

- Mechanical waves that propagate through a medium (solid, liquid, or gas).
- Result from compressions and rarefactions of molecules in the medium.

3. Water Waves:

- Mechanical waves traveling on the surface of a body of water.
- Can be transverse or longitudinal, depending on the motion of water particles.

Optical Phenomena:

1. Refraction:

- Bending of light as it passes from one medium to another with a different optical density.
- Commonly observed when light passes through lenses.

2. Reflection:

- Bouncing back of light when it encounters a reflective surface.
- Essential for mirrors and various optical devices.

3. Diffraction:

- Bending of light around obstacles or through openings.
- Leads to the spreading of light waves.

4. Interference:

- Combination of two or more light waves to produce a resultant wave.
- Can be constructive or destructive, influencing light patterns.

$$\oint E \cdot \mathrm{dS} = \int \rho \, \mathrm{dV} = Q$$

$$\oint B \cdot d\ell - \frac{1}{C} \frac{\partial}{\partial t} \int E \cdot \mathrm{dS}$$

Spherical Mirror

Spherical mirrors are like shiny, curved pieces of glass. They can be shaped like the inside of a ball (concave) or the outside of a ball (convex).

Concave Mirrors:

- Shape: Curved inward, like a cave.
- What they do: Concave mirrors can make parallel light rays come together at a point. It's like they bring the sunlight or light from a flashlight to a focus, making things look brighter.

Convex Mirrors:

- Shape: Curved outward.
- What they do: Convex mirrors spread out parallel light rays. They're like security mirrors in stores or car side mirrors, making things look smaller and allowing us to see a wider area.

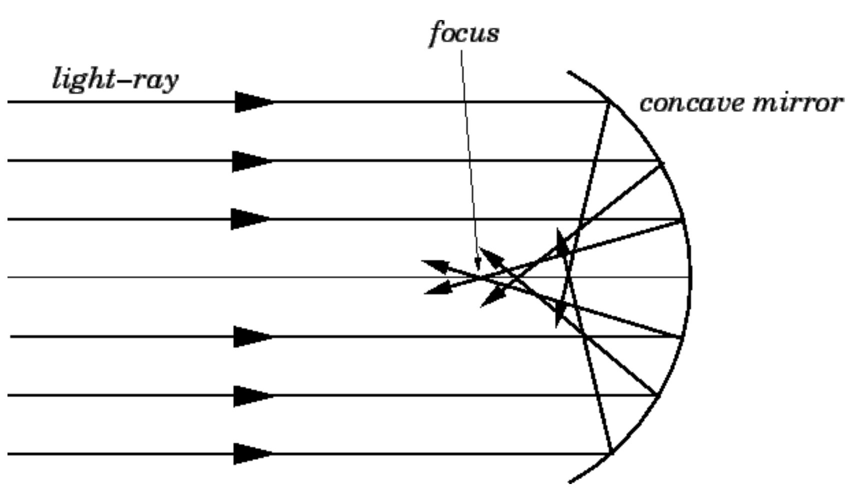


Figure 2: Four electromagnetic radiation interaction processe with matter. Source: Tomal et al. (2023)

```
%
% Program to Ray Trace the Focus of a Spherical Mirror and Look at
```

```
% Aberration
%
clear all;
help Spherical_Mirror;  % Clear memory and print header
```

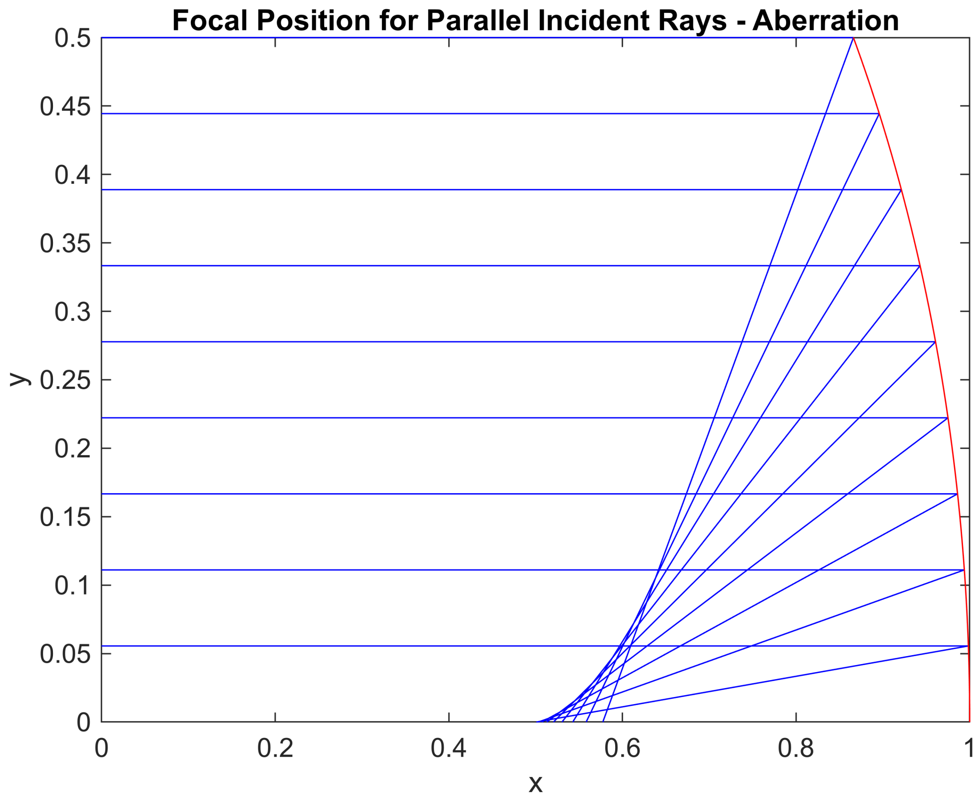
```
  Program to Ray Trace the Focus of a Spherical Mirror and Look at
  Aberration
```

```
%
fprintf(' Focal Point of a Spherical Mirror, Radius = 1 \n');
```

```
 Focal Point of a Spherical Mirror, Radius = 1
```

```
%
% unit radius, with incident parallel rays from the left (x < 0)
%
iloop = 0;
irun = 1;
%
while irun > 0
    kk = menu('Pick Another Maximum Incident Ray for the Mirror?','Yes','No');
    if kk == 2
        irun = -1;
        break
    end
    if kk == 1
        %
        ymax = input('Enter The Maximum Value of the Incident Parallel Ray y : ');
        yray = linspace(0,ymax, 10);
        xhit = cos(asin(yray));   % the incident rays
        theta = asin(yray);
        xr(1,:) = [-1 -1 -1 -1 -1 -1 -1 -1 -1 -1];
        xr(2,:) = xhit;
        yr(1,:) = yray(:);
        yr(2,:) = yray(:);
        yr(3,:) = 0.0;
        xr(3,:) = (xhit .*tan(2.0 .*theta) - yray) ./(tan(2.0 .*theta));
        %
        iloop = iloop + 1;
        figure(iloop)
        for i = 1:10
            plot(xr(:,i),yr(:,i),'b-')
            plot(xr(2,i),yr(2,i),'b-',xr(3,i),yr(3,i),'b-')
            hold on
        end
        ym = linspace(0,1);   % the mirror
        xm = sqrt(1 - ym .^2);
        plot(xm,ym,'r-')
        title('Focal Position for Parallel Incident Rays - Aberration')
        xlabel('x')
        ylabel('y')
```

```
        axis([-0.0 1.0 0 ymax]);
        %
        hold off
    end
end
```




```
%
```

Spherical Lenses

Shape : A spherical lens has a curved surface, much like a ball cut in half.

Light Interaction: It bends (refracts) light that passes through it.

Types Spherical Lense :

1) Convex Lens:

- This lens converges a straight beam of light
- This lens is thicker at the centre and gets thinner as we move towards the edges.

2) Concave Lens:

- This lens diverges a straight beam of light.
- This lens is thinner at the centre and gets thicker as we move towards the edges.

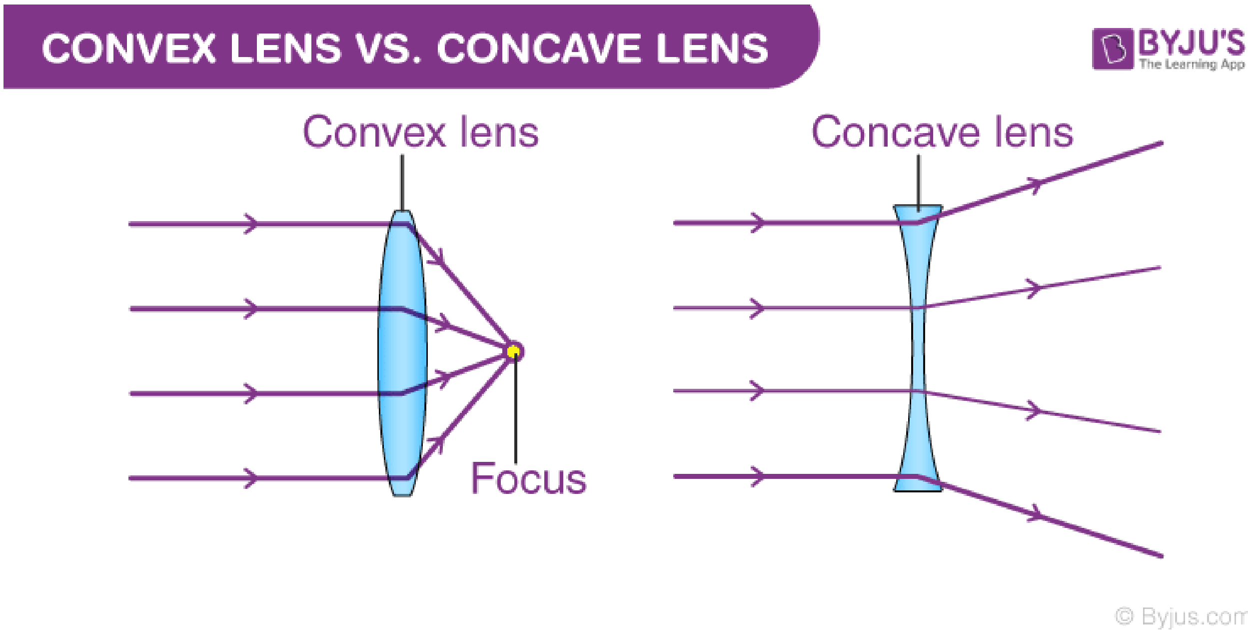


Figure 3 : Convex Lens and Concave Lens.Source ConvexConcave

Lense Formula

- The lens formula is a mathematical expression that relates the focal length (f), object distance (u), and image distance (v) for a thin lens. The formula is based on the principles of geometric optics. The lens formula is given by:
- The lens formula is given by,

$$\frac{1}{f} = \frac{1}{v} - \frac{1}{u}$$

- Where v distance of the image formed from the optical center of the lens,u is the distance of the object from the optical centre of the lens and f is the focal length of the lens.�

Spherical Lenses Ray Tracing Simulation

This MATLAB script simulates ray tracing through a thick spherical lens.It visualizes the behavior of parallel rays as they pass through the lens,demonstrating the impact of lens geometry on focal aberrations.

```
%
% Ray tracing for a thick shperical lense
%
```

```
clear all;
help Spherical_Lense      % Clear the memory and print header
```

```
  Ray tracing for a thick shperical lense
```

```
%
% Initialize
%
fprintf('Lens Radius = 10, Index of Refraction = 1.5 \n ')
```

```
Lens Radius = 10, Index of Refraction = 1.5
```

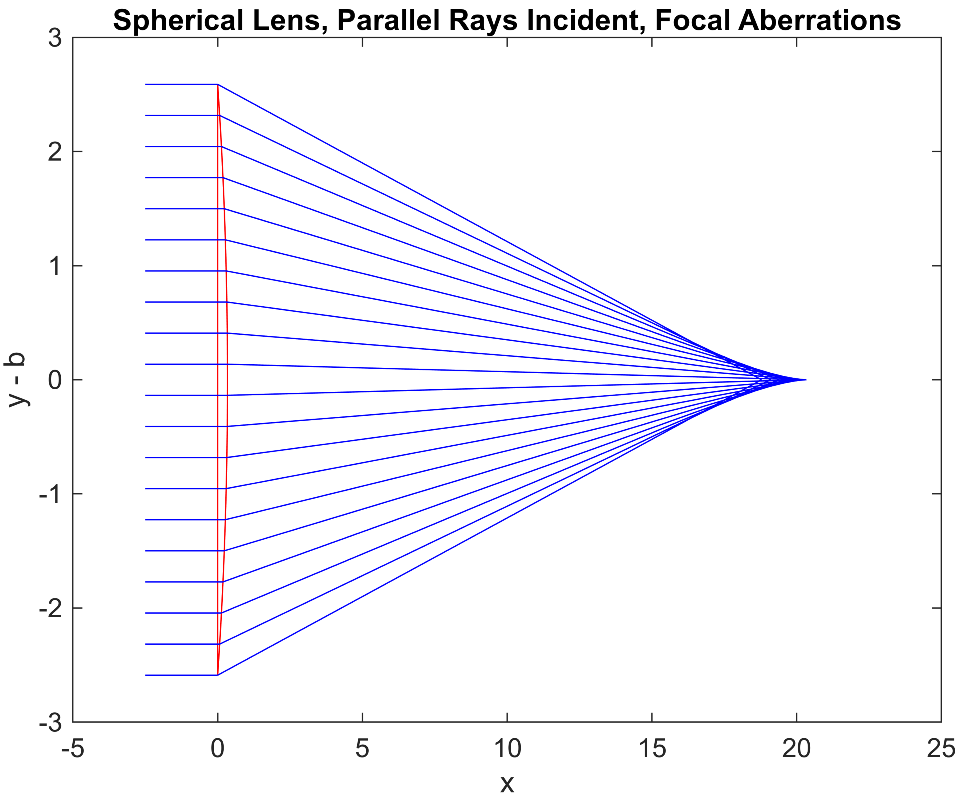
```
fprintf('Lens Makers Equation - 1/f = (n-1)/R \n ')
```

```
Lens Makers Equation - 1/f = (n-1)/R
```

```
%
R = 10;
n = 1.5;
%
iloop = 0;
irun = 1;
%
while irun > 0
    kk = menu('Pick Another Shape of the Lense?','Yes','No');
    if kk == 2
        irun = -1;
        break
    end
    if kk == 1
        %
        theta = input('Enter The Angular Size of the Lense in Degrees, < 60: ');
        theta = pi .*theta ./360.0;
        %
        % sagitta and center of the circle forming the lense, center of lense at
(0,0)
        % flat face at x=0, wave incident from the left
        %
        s = R .*(1.0 - cos(theta)); % sagitta
        yl = R .*sin(theta); % y boundary of lense
        %
        fprintf('Lense Thickness = %g and 1/2 Height = %g \n ', s , yl)
        %
        % draw the lens
        yd = linspace(-yl,yl,25);
        for i = 1:length(yd)
            xdm(i) = 0;
        end
        xdp = sqrt(R .^2 - yd .^2) -R + s;
        %
```

```
        iloop = iloop + 1;
        figure(iloop)
        plot(xdm,yd,'r-',xdp,yd,'r-')
        hold on
        %
        % parallel rays in - impact height b
        %
        b = linspace(-yl,yl,20);
        xdpb = sqrt(R .^2 - b .^2) -R + s;
        for i = 1:length(b)
            x(1,i) = -R/4;
            y(1,i) = b(i);
            x(2,i) = xdpb(i) ;
            y(2,i) = b(i);
            plot(x(:,i),y(:,i),'b-');  % plot incoming parallel rays
        end
        %
        sinth = b ./R;    % angle of ray w.r.t. normal
        sinthr = n .*sinth;  % refracted angle w.r.t. normal
        phir = asin(sinthr) - asin(sinth);   % angle of refracted ray
        phir = -phir;  % converging
        c = b - tan(phir) .*xdpb;
        fray = (-b + tan(phir) .*xdpb) ./tan(phir);
         for i = 1:length(b)
            x(1,i) = xdpb(i);
            y(1,i) = b(i);
            x(2,i) = fray(i);
            y(2,i) = 0;
            plot(x(:,i),y(:,i),'b-');  % plot refrated rays to focus @ y = 0
            title('Spherical Lens, Parallel Rays Incident, Focal Aberrations')
            xlabel('x')
            ylabel('y - b')
        end
        %
       hold off
    %
    end
end
```

```
Lense Thickness = 0.340742 and 1/2 Height = 2.58819
```




```
%
```

Diffraction Phenomena

- Diffraction is what happens when waves encounter an obstacle or opening. They bend and spread out.
- It occurs with various waves like light, sound, or even particles like electrons.
- When waves meet something (like a slit or edge), they don't just stop; they bend around it.
- In light, it's like when sunlight bends around the edge of a building.
- This bending creates patterns of light and dark areas – some areas where waves reinforce, others where they cancel out.

Applications:

- Important in optics, acoustics, and quantum mechanics.

Type of diffraction :

1) Single Slit Diffraction:

- One narrow slit in a barrier.
- Waves spread out, creating bright and dark regions.
- Waves bend as they pass through the slit, causing interference.

2) Double Slit Diffraction:

- Two closely spaced slits in a barrier.
- Interference pattern with bright and dark bands.
- Waves from the two slits interfere, reinforcing or canceling each other.

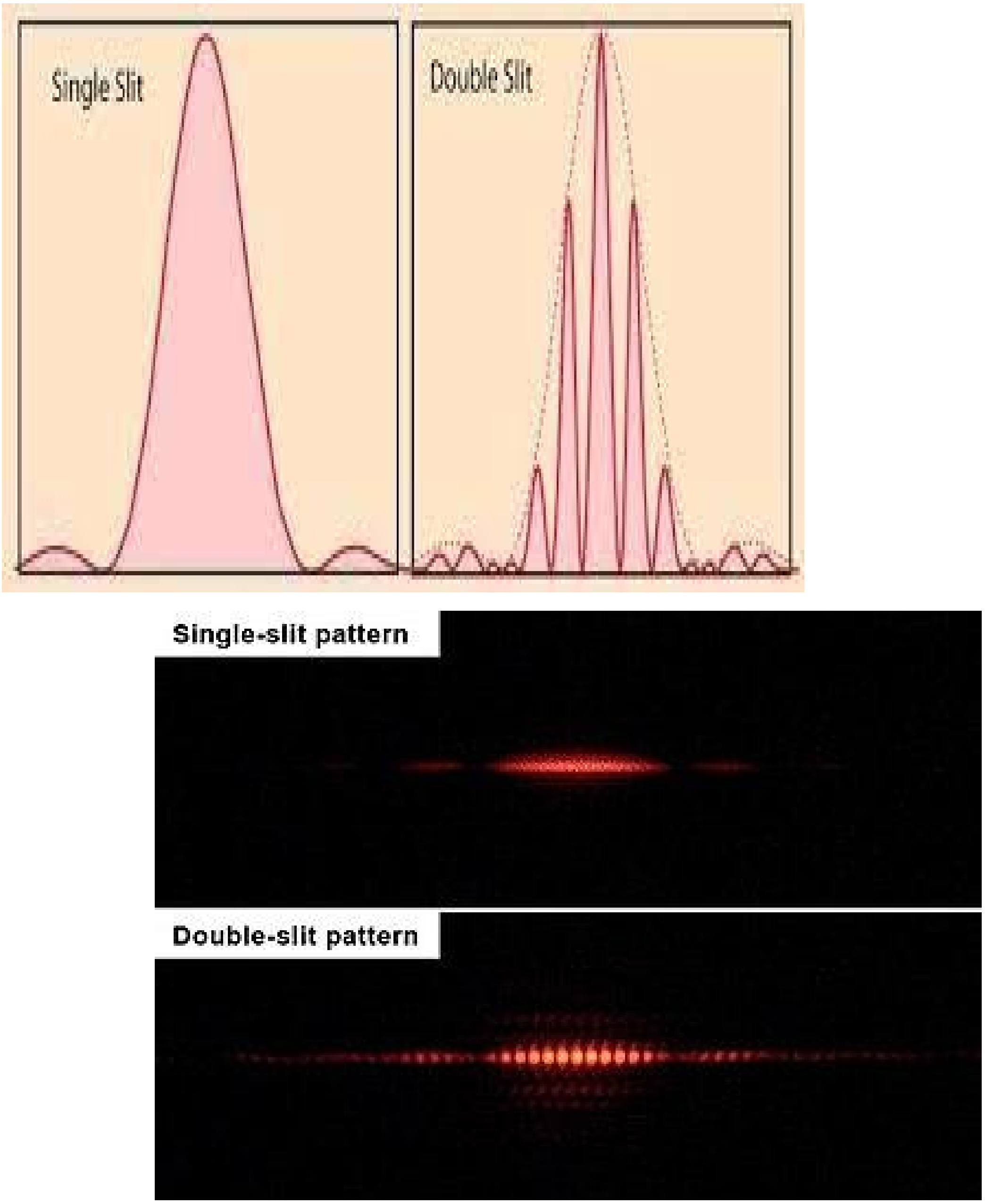


Figure 4 : Single slit and double slit pattern of diffraction

```
%
% Program to Illustrate Diffraction at a Single Slit and Double Slit
% use MATLAB bessel function for circular aperture
%
clear all;
help Diffract;  % Clear memory and print header
```

```
  Program to Illustrate Diffraction at a Single Slit and Double Slit
  use MATLAB bessel function for circular aperture
```

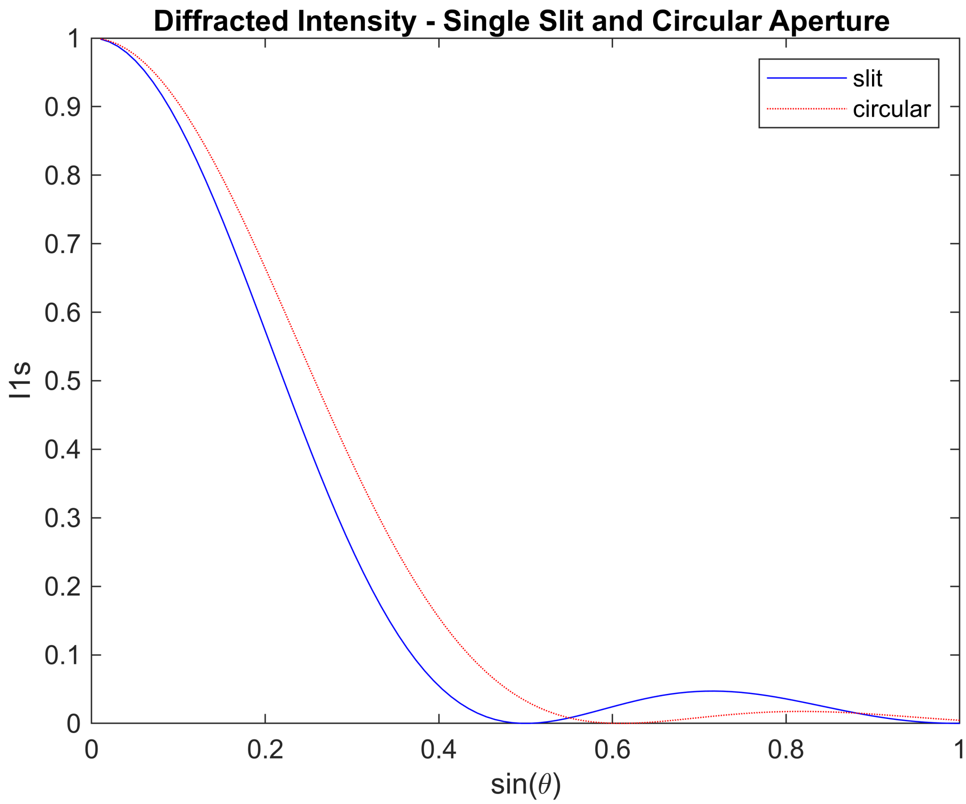
```
%
fprintf(' Diffraction at a Single and Double Slit \n');
```

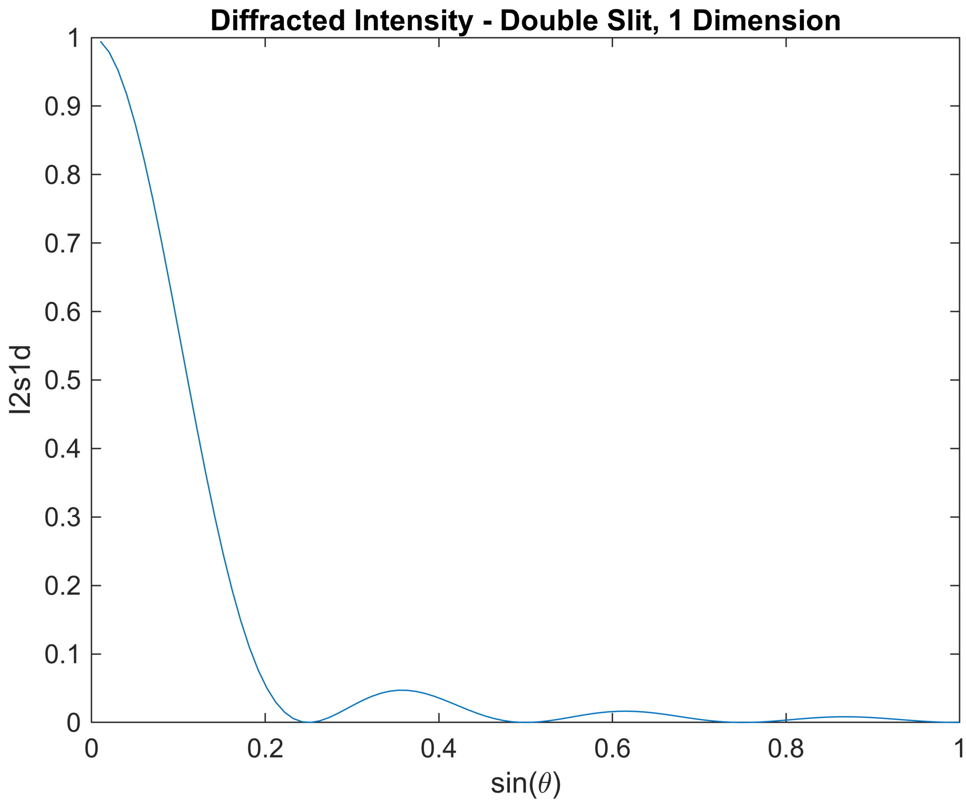
```
 Diffraction at a Single and Double Slit
```

```
%
iloop = 0;
irun = 1;
%
while irun > 0
    kk = menu('Pick Another Single Slit Size?','Yes','No');
    if kk == 2
        irun = -1;
        break
    end
    if kk == 1
        %
        dslit = input('Input Ratio of Slit Width the Incident Wavelength ~
Resolving Power: ');
        %
        fprintf(' Path Difference = d*sintheta, Phase Difference = pi*d*sintheta/
lambda ')
        %
        sinth = linspace(0,1.0); % angle of observation
        dphi = pi .*sinth .*dslit;
        I1d = (sin(dphi) ./(dphi)) .^2;
        %
        fprintf(' Phase Difference for Circular Aperture, k*D*sintheta/lambda, D =
diameter \n')
        %
        Icirc = (2.0 .*besselj(1,dphi) ./dphi) .^2 ;
        %
        iloop = iloop + 1;
        figure(iloop)
        plot(sinth, I1d,'b-',sinth,Icirc,'r:')
        title('Diffracted Intensity - Single Slit and Circular Aperture ')
        xlabel('sin(\theta)')
        ylabel('I1s')
        legend('slit','circular')
        %
```

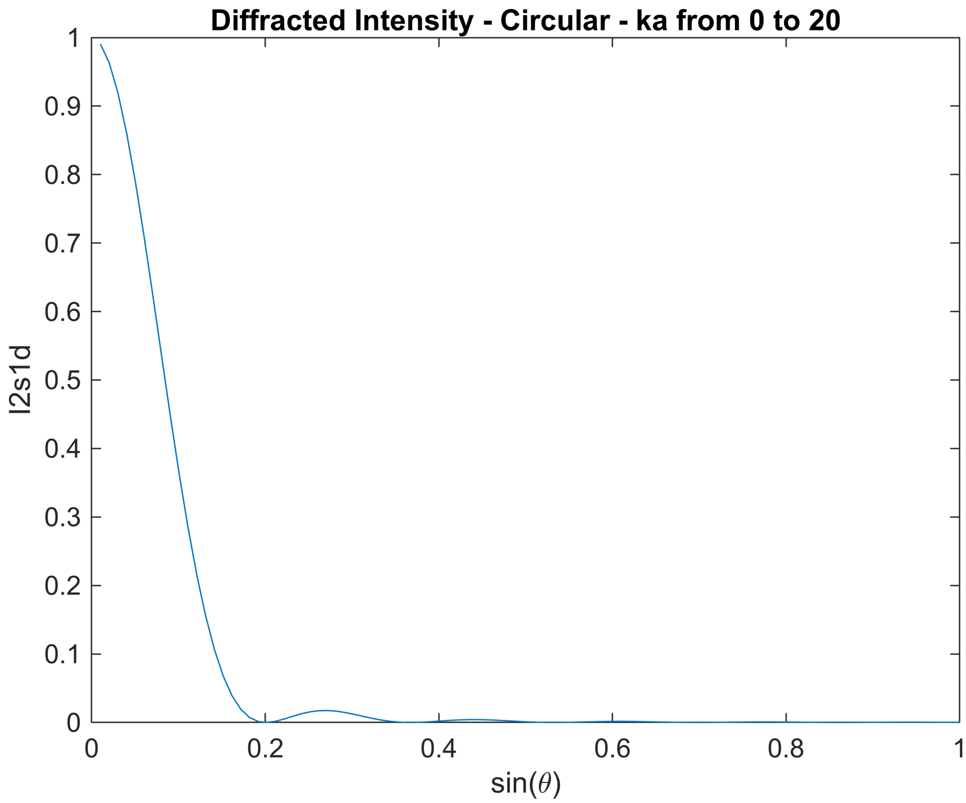
```
        al = input('Input Ratio of Two Slit Seperation to the Incident Wavelength:
');
        %
        phase = pi .*sinth .*al;
        I2s1d = I1d .*((cos(phase)) .^2);
        iloop = iloop + 1;
        figure(iloop)
        plot(sinth, I2s1d, '-')
        title('Diffracted Intensity - Double Slit, 1 Dimension')
        xlabel('sin(\theta)')
        ylabel('I2s1d')
        %
    end
end
```

```
Path Difference = d*sintheta, Phase Difference = pi*d*sintheta/lambda
Phase Difference for Circular Aperture, k*D*sintheta/lambda, D = diameter
```






```
%
% movie for circular aperture - vary ka
%
iloop = iloop + 1;
figure(iloop)
for i = 1:20
    ka =  0.1  + 1 .*(i - 1);
    sinth = linspace(0,1.0); % angle of observation
    dphi = ka .*sinth;
    Icirc = (2.0 .*besselj(1,dphi) ./dphi) .^2 ;
    %
    plot(sinth, Icirc, '-')
    title('Diffracted Intensity - Circular - ka from 0 to 20')
    xlabel('sin(\theta)')
    ylabel('I2s1d')
    axis([0 1.0 0 1])
    pause(0.5)
end
```




```
%
```

Doppler Cherenkov effect

Definition: The Cherenkov effect occurs when a charged particle moves through a medium at a speed greater than the phase velocity of light in that medium, resulting in the emission of characteristic blue light.

Definition: The Doppler Cherenkov effect refers to changes in the emitted light frequency due to the motion of the Cherenkov emitter.

Applications:

- Particle Physics: Used to detect high-energy particles in experiments.
- Medical Imaging: Applied in techniques such as Cherenkov luminescence imaging.

```
% Program to illustrate || and T Doppler and Cerenkov Mach cone
%
clear all;
help Doppler_Cerenkov;  % Clear memory and print header
```

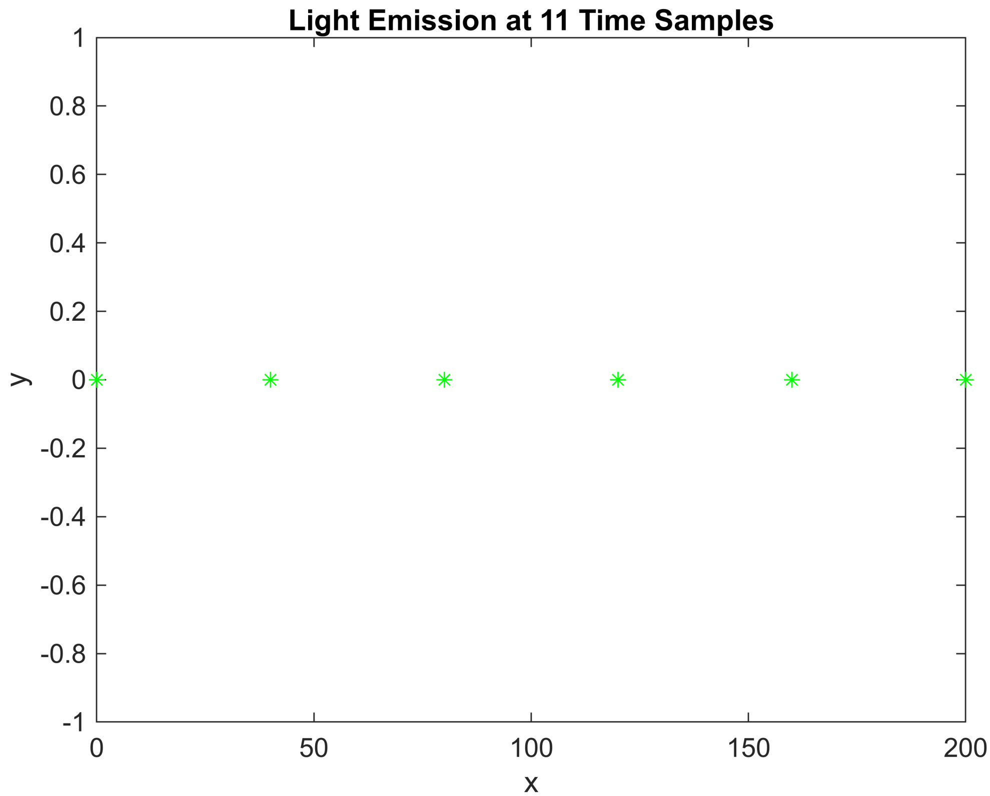
```
Program to illustrate || and T Doppler and Cerenkov Mach cone
```

```
%
fprintf(' Doppler Wavelength Shift - Red, Blue and Transverse \n');
```

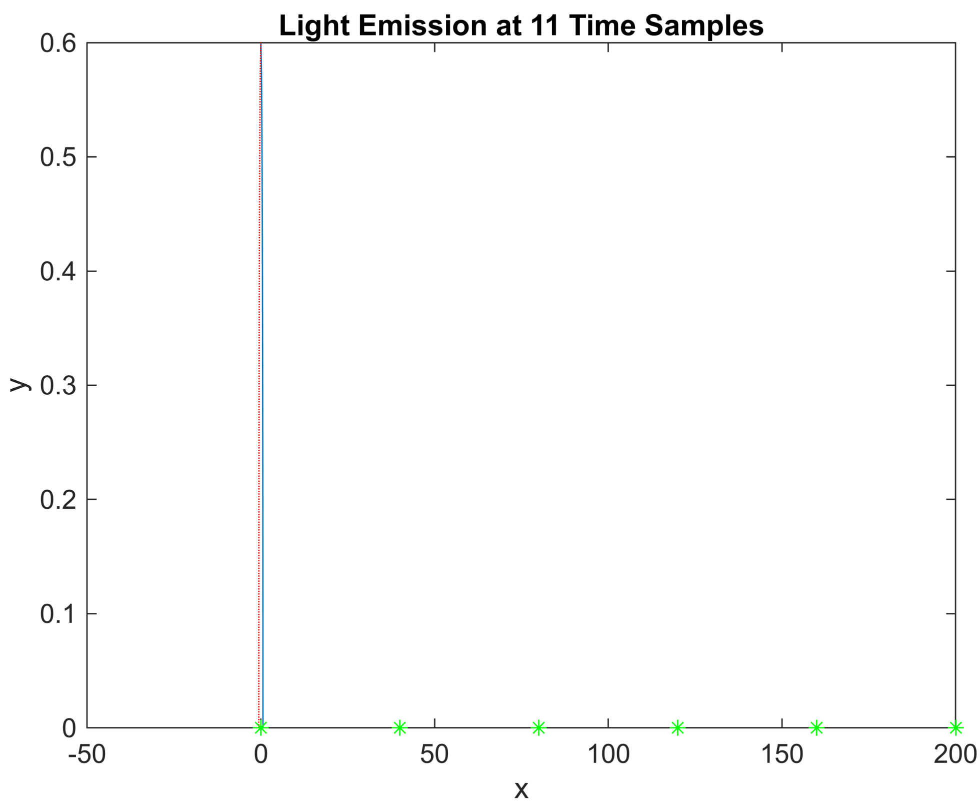
```
 Doppler Wavelength Shift - Red, Blue and Transverse
```

```
%
iloop = 0;
irun = 1;
%
while irun > 0
    kk = menu('Pick Another Velocity of the Source?','Yes','No');
    if kk == 2
        irun = -1;
        break
    end
    if kk == 1
        %
        v = input('Input Velocity of Source w.r.t. Light, sin(thetac) = 1/n: ');
        %
        % make for 6 emission times and points
        %
        te = linspace(0,5,6);
        %
        xe = v .*te;  % x position at emission
        ye = 0.0 .*te;
        %
        % sample at 10 times
        %
        t = linspace(0,6,11);
        %
        for i = 1:11 ; % time of snapshot
            if t(i) > te(1);   % after emission at point 1
                r = (t(i) - te(1));
                yr = linspace(0,r,25);
                y1(i,:) = yr;
                x1(i,:) =sqrt(r .^2 - y1(i,:) .^2) + xe(1);
                xx1(i,:) = -sqrt(r .^2 - y1(i,:) .^2) + xe(1);
            end
            if t(i) > te(2);   % after emission at point 2
                r = (t(i) - te(2));
                yr = linspace(0,r,25);
                y2(i,:) = yr;
                x2(i,:) =sqrt(r .^2 - y2(i,:) .^2) + xe(2);
                xx2(i,:) = -sqrt(r .^2 - y2(i,:) .^2) + xe(2);
            end
            if t(i) > te(3);   % after emission at point 3
                r = (t(i) - te(3));
                yr = linspace(0,r,25);
                y3(i,:) = yr;
                x3(i,:) =sqrt(r .^2 - y3(i,:) .^2) + xe(3);
```

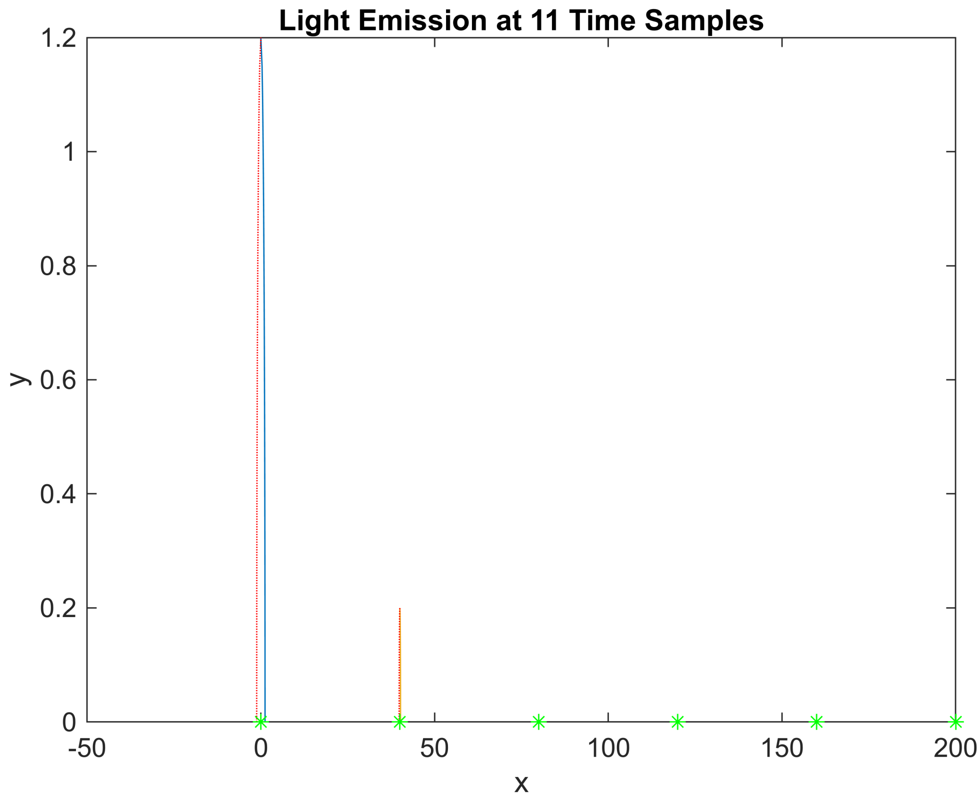
```
                xx3(i,:) = -sqrt(r .^2 - y3(i,:) .^2) + xe(3);
            end
            if t(i) > te(4);   % after emission at point 4
                r = (t(i) - te(4));
                yr = linspace(0,r,25);
                y4(i,:) = yr;
                x4(i,:) =sqrt(r .^2 - y4(i,:) .^2) + xe(4);
                xx4(i,:) = -sqrt(r .^2 - y4(i,:) .^2) + xe(4);
            end
            if t(i) > te(5);   % after emission at point 5
                r = (t(i) - te(5));
                yr = linspace(0,r,25);
                y5(i,:) = yr;
                x5(i,:) =sqrt(r .^2 - y5(i,:) .^2) + xe(5);
                xx5(i,:) = -sqrt(r .^2 - y5(i,:) .^2) + xe(5);
            end
            if t(i) > te(6)   % after emission at point 6
                r = (t(i) - te(6));
                yr = linspace(0,r,25);
                y6(i,:) = yr;
                x6(i,:) =sqrt(r .^2 - y6(i,:) .^2) + xe(6);
                xx6(i,:) = -sqrt(r .^2 - y6(i,:) .^2) + xe(6);
            end
        end
        %
        iloop = iloop + 1;
        for i = 1:11
            figure(11 .*(iloop -1) +i)
            plot(x1(i,:), y1(i,:),'-',xx1(i,:),y1(i,:),':r')
            hold on
            plot(x2(i,:), y2(i,:),'-',xx2(i,:),y2(i,:),':r')
            plot(x3(i,:), y3(i,:),'-',xx3(i,:),y3(i,:),':r')
            plot(x4(i,:), y4(i,:),'-',xx4(i,:),y4(i,:),':r')
            plot(x5(i,:), y5(i,:),'-',xx5(i,:),y5(i,:),':r')
            plot(x6(i,:), y6(i,:),'-',xx6(i,:),y6(i,:),':r')
            title(' Light Emission at 11 Time Samples ')
            xlabel('x')
            ylabel('y')
            plot(xe,ye,'g*')
            pause(1);
            hold off
        end
        %
    end
end
```



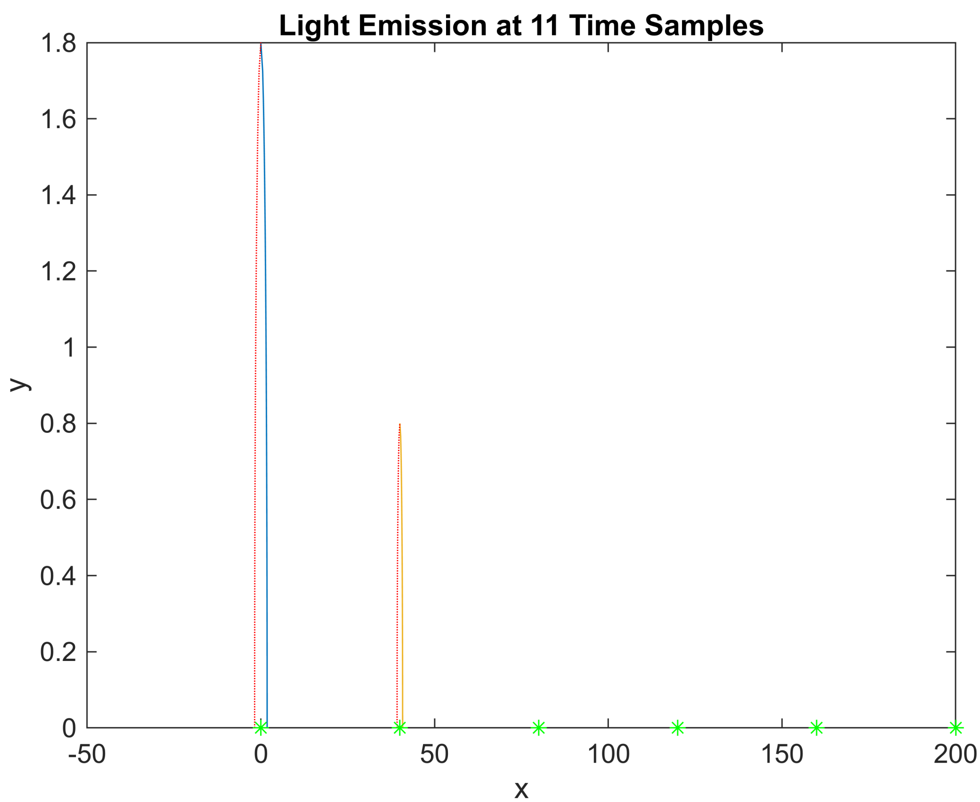
Light Emission at 11 Time Samples
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
y
0
50
100
150
200
x



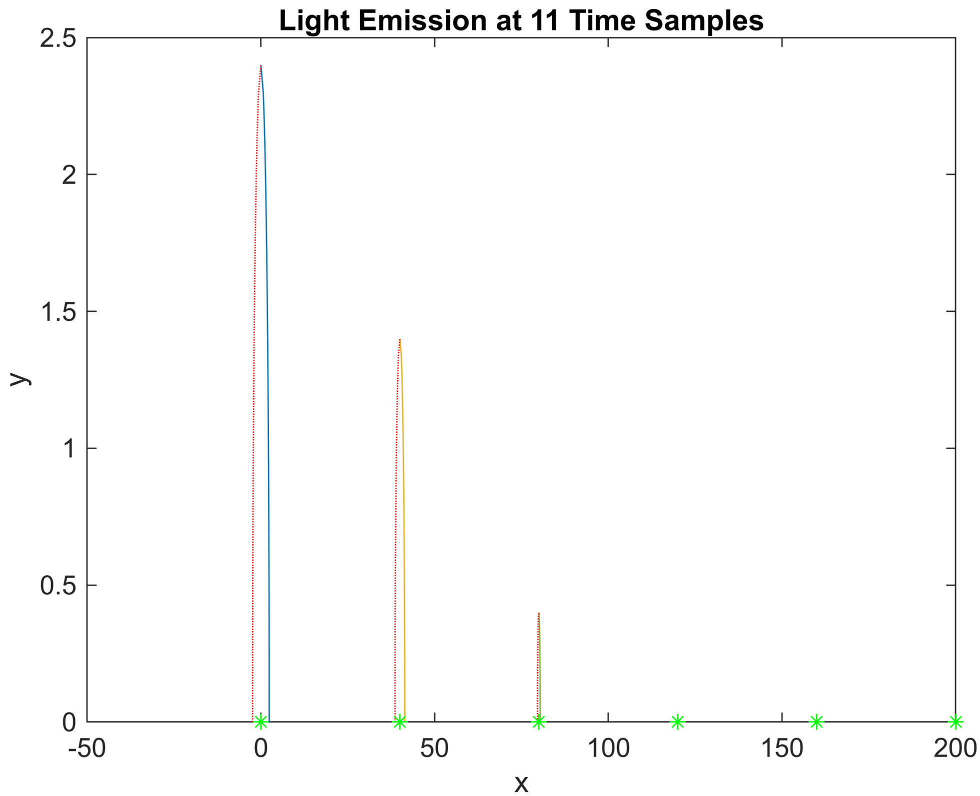
Light Emission at 11 Time Samples
0.6
0.5
0.4
0.3
0.2
0.1
0
y
-50
0
50
100
150
200
x

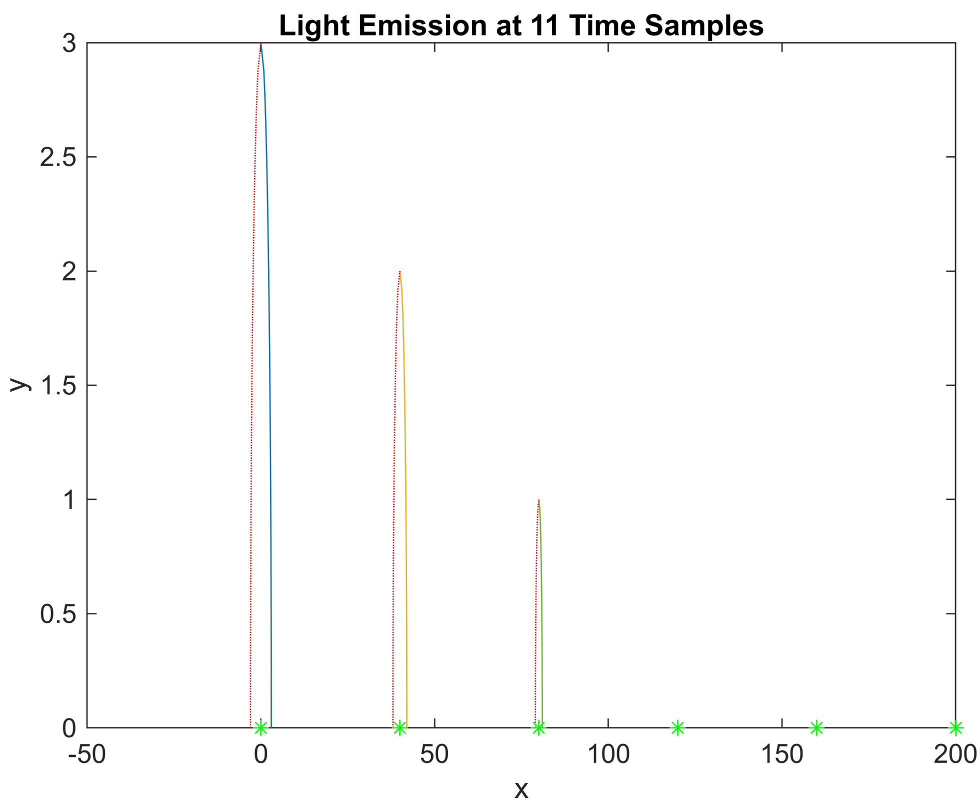


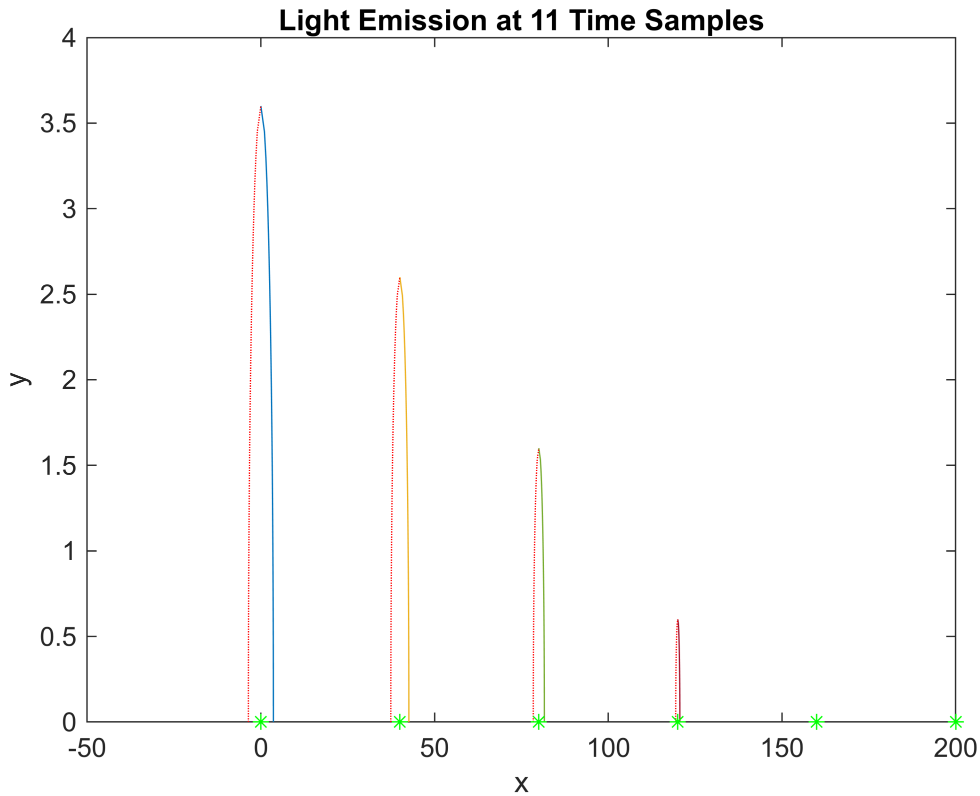
Light Emission at 11 Time Samples
1.2
1
0.8
y 0.6
0.4
0.2
0
-50
0
50
100
150
200
x

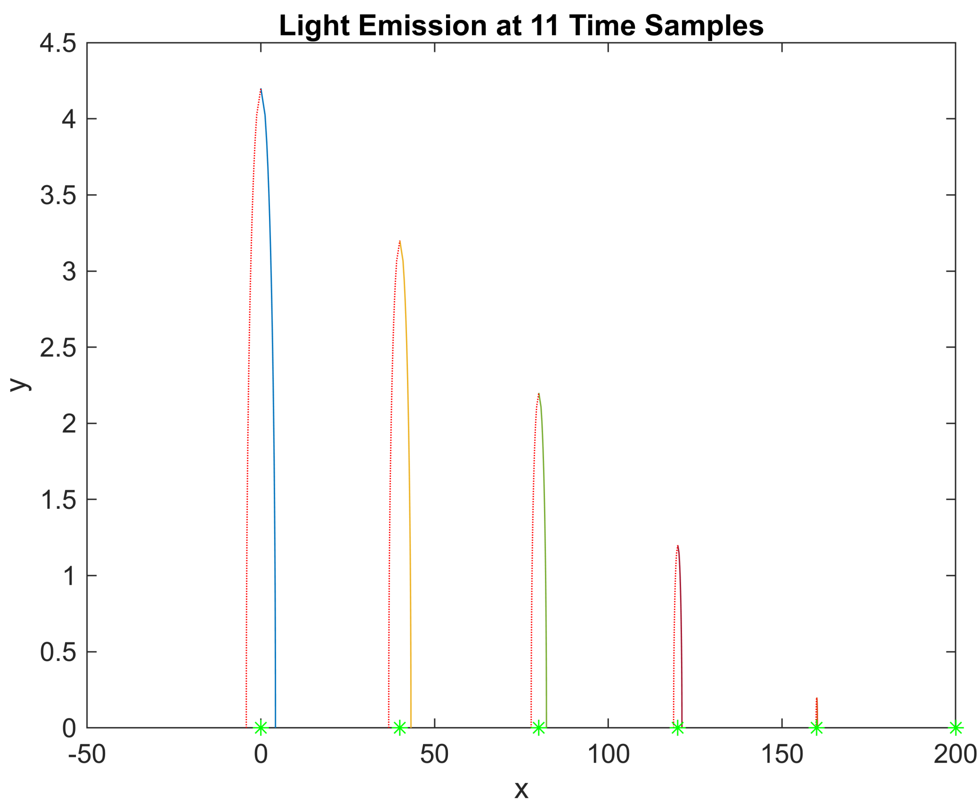


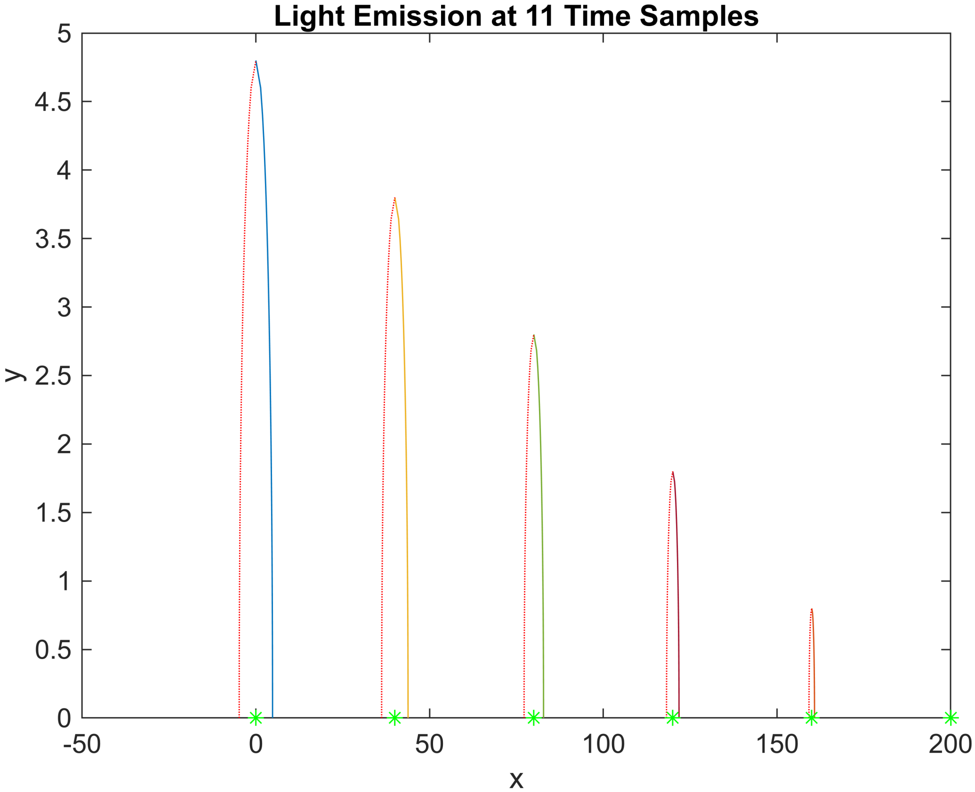
Light Emission at 11 Time Samples
1.8
1.6
1.4
1.2
1
y
0.8
0.6
0.4
0.2
0
-50
0
50
100
150
200
x

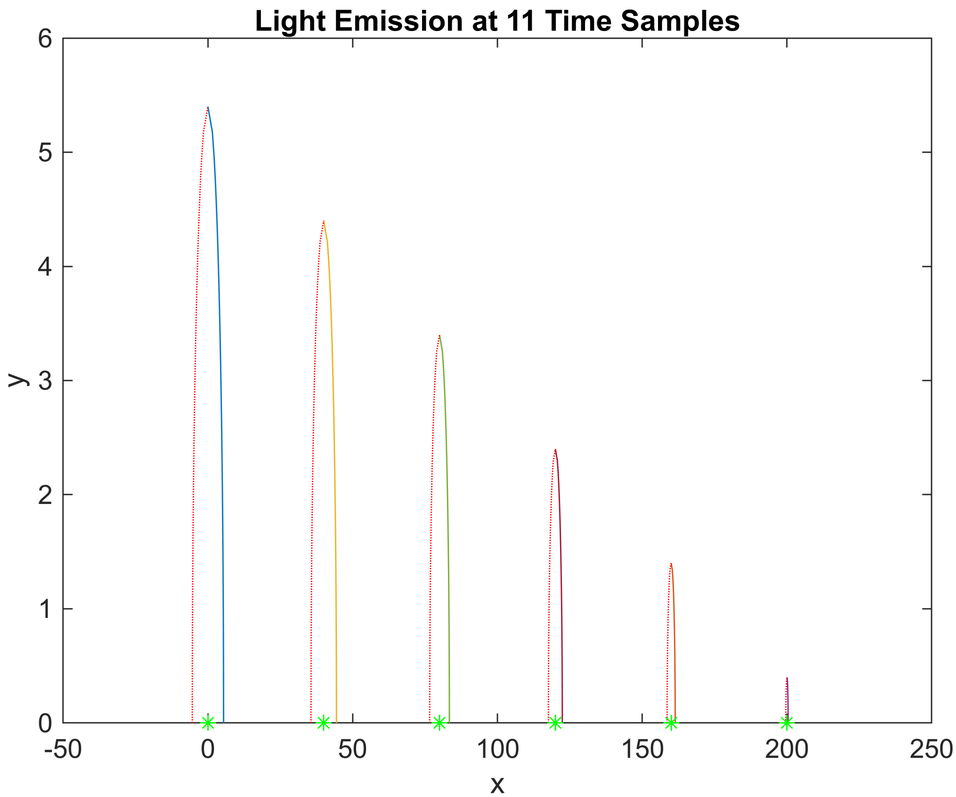

Light Emission at 11 Time Samples
y
x
2.5
2
1.5
1
0.5
0
-50
0
50
100
150
200

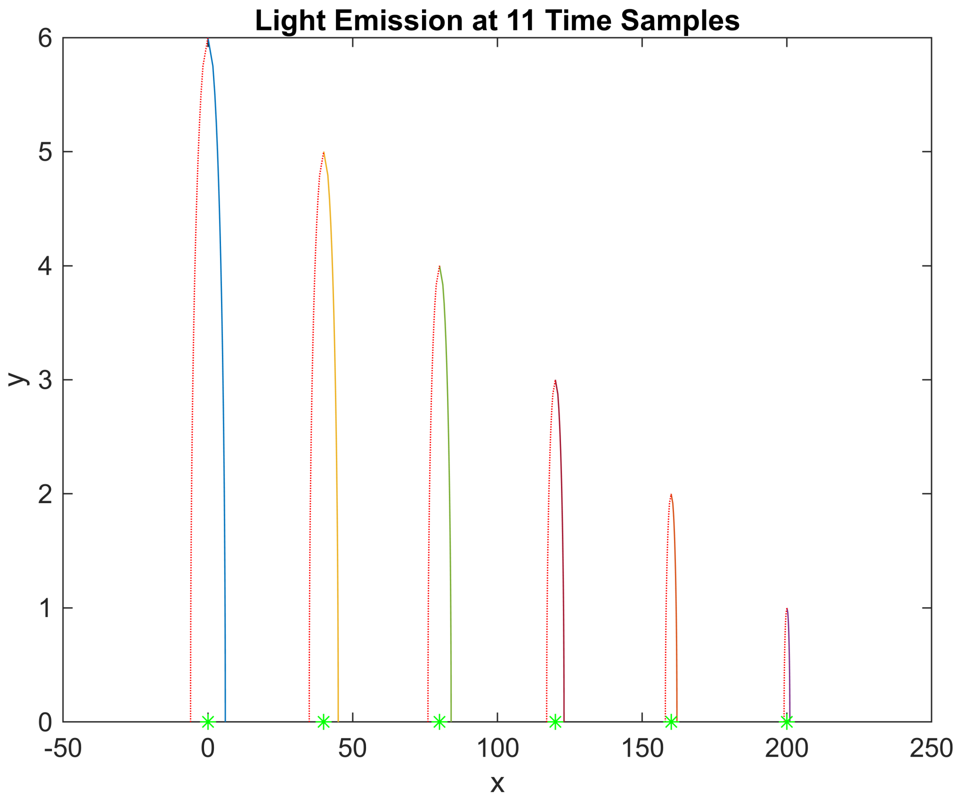

Light Emission at 11 Time Samples
y
x
3
2.5
2
1.5
1
0.5
0
-50
0
50
100
150
200


Light Emission at 11 Time Samples
y
x
4
3.5
3
2.5
2
1.5
1
0.5
0
-50
0
50
100
150
200


Light Emission at 11 Time Samples
y
x
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
-50
0
50
100
150
200


Light Emission at 11 Time Samples
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
y
-50
0
50
100
150
200
x


Light Emission at 11 Time Samples
6
5
4
3
2
1
0
y
-50
0
50
100
150
200
250
x


Light Emission at 11 Time Samples
y
x
6
5
4
3
2
1
0
-50
0
50
100
150
200
250

```
%
```