Ain Shams University
Faculty of Engineering
Electronics & Comm. Eng. Dept.

ECE331s: Electromagnetic Waves
3rd Year Electrical Engineering
Fall 2025

Propagation of a Gaussian wave

Names	IDs	Section	Group	Group code
Mahmoud Ahmed Gamal	2200433	1	6	Wavelength (λ) = 3mm Beam waist (Wo)=10mm
Ahmed Tarek Hassanien	2200772			
Osama Mohammed Sultan	2101140			

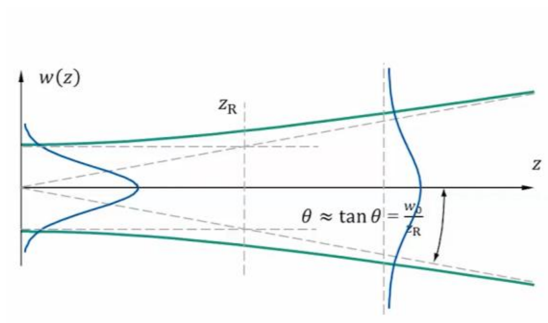
Table of Contents

❖INTRODUCTION:

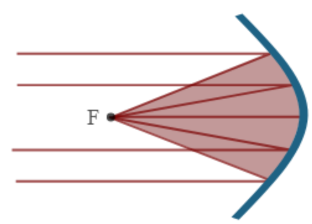
This report details the numerical simulation of a Gaussian optical beam as it propagates through free space, reflects from a parabolic mirror, and continues to propagate thereafter. The project implements Fourier optics principles to model the beam's evolution in the spatial frequency domain, utilizing the Fast Fourier Transform (FFT) and inverse FFT in MATLAB

❖OBJECTIVES:

- Simulating the initial Gaussian field distribution at $z = 0$.
- Propagating the beam in air to specified distances.



- Applying the phase transformation of a parabolic reflector.



- Analyzing the intensity profile before and after reflection at multiple axial positions. The report provides the theoretical background, describes the implementation methodology, and interprets the simulated results in the context of Gaussian beam optics.

❖THEORY

The simulation is based on the following well-established principles of wave optics and Gaussian beams, as derived from the provided references:

1. **Gaussian Beam in the Paraxial Regime**

The fundamental Gaussian beam is a solution to the paraxial Helmholtz equation. At the beam waist ($z = 0$), the field amplitude is:

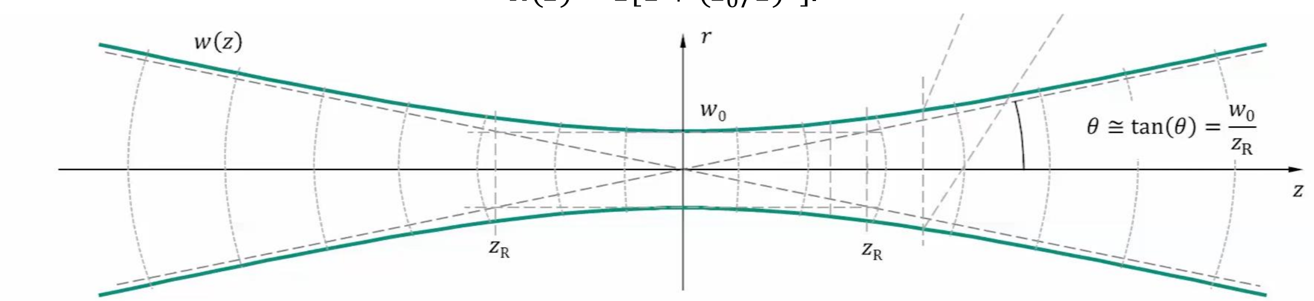
$$U(x, y, 0) = A_0 \exp\left(-\frac{x^2 + y^2}{w_0^2}\right),$$

where w_0 is the waist radius. The beam propagates along z with a characteristic Rayleigh range $z_0 = \pi w_0^2/\lambda$. The beam width expands as:

$$w(z) = w_0\sqrt{1 + (z/z_0)^2},$$

and the wavefront radius of curvature is:

$$R(z) = z[1 + (z_0/z)^2].$$



2. **Angular Spectrum Method for Propagation**

The field $U(x, y, 0)$ is decomposed into plane-wave components via a 2D Fourier transform:

$$\widetilde{U}(k_x, k_y, 0) = \iint U(x, y, 0) e^{-j(k_x x + k_y y)} dx dy.$$

Propagation in free space by a distance z is modeled by multiplying the angular spectrum by the transfer function:

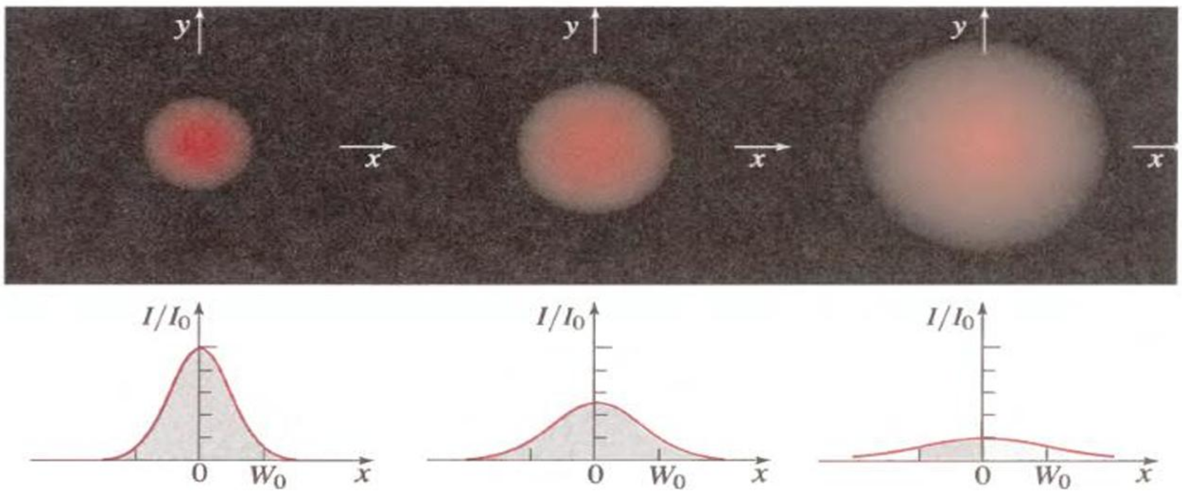
$$H(k_x, k_y) = \exp(-jk_z z), k_z = \sqrt{k^2 - k_x^2 - k_y^2}.$$

Under the paraxial approximation ($k_x, k_y \ll k$), this simplifies to:

$$H(k_x, k_y) \approx \exp\left[-jkz\right] \exp\left[j \frac{(k_x^2 + k_y^2)z}{2k}\right].$$

The propagated field is obtained by the inverse Fourier transform:

$$U(x, y, z) = \mathcal{F}^{-1}\left\{\widetilde{U}(k_x, k_y, 0) \cdot H(k_x, k_y)\right\}.$$



ؤ

3. **Interaction with a Parabolic Mirror**

A parabolic reflector imparts a quadratic phase shift that modifies the beam's wavefront curvature. The mirror's complex reflection coefficient (phase factor) is:

$$M(x, y) = \exp\left[-jk \frac{x^2 + y^2}{2f}\right],$$

where f is the mirror's focal length (negative for a concave mirror). This is analogous to a thin lens in transmission. The reflected field just after the mirror is:

$$U_{\mathrm{refl}}(x, y) = M(x, y) \cdot U_{\mathrm{inc}}(x, y).$$

4. **ABCD Law for Gaussian Beam Transformation**

The effect of the mirror on the Gaussian beam can be described using the complex beam parameter q, where:

$$\frac{1}{q} = \frac{1}{R} - j \frac{\lambda}{\pi w^2}.$$

For reflection from a spherical mirror of radius R_m (with sign convention: $R_m < 0$ for concave), the equivalent focal length is $f = -R_m/2$. The transformation follows:

$$\frac{1}{q_2} = \frac{1}{q_1} - \frac{1}{f},$$

which directly yields the new waist location and size.

5. **Numerical Implementation via FFT:**

The simulation uses discrete Fourier transforms (FFT2 and IFFT2) to implement the angular spectrum method. Critical numerical considerations include:

Sampling in the spatial domain must satisfy the Nyquist criterion relative to the highest spatial frequency.

The simulation domain must be large enough to avoid aliasing due to the beam expansion.

The phase factor $H(k_x, k_y)$ must be carefully implemented to handle evanescent waves (if $k_x^2 + k_y^2 > k^2$).

6. **Higher-Order Beam Modes (Contextual Background)**

Although the simulation focuses on the fundamental Gaussian mode, the theory generalizes to Hermite–Gaussian (HG) and Laguerre–Gaussian (LG) beams, which form complete sets of orthogonal solutions to the paraxial wave equation. These higher-order modes exhibit more complex intensity patterns and carry orbital angular momentum (in the case of LG beams), but their propagation and transformation obey similar Fourier optics principles.

❖MATLAB IMPLEMENTATION

For the given parameters: **λ = 3 mm** and **w_0 = 10 mm**, the Rayleigh range is calculated as:

$$z_0 = \frac{\pi w_0^2}{\lambda} = \frac{\pi(0.01)^2}{0.003} \approx 0.1047\ \mathrm{m} = 104.7\ \mathrm{mm}$$

PART1: Propagation of Gaussian Beam:

The beam propagates in free space to three axial positions:

- $z = 0$ (beam waist)
- $z = 0.5z_0 \approx 52.35$ mm
- $z = z_0 \approx 104.7$ mm

a) Code Implementation:

The following MATLAB code simulates the propagation using the angular spectrum method. The spatial grid is set to cover ±20 mm, and the sampling step is chosen to satisfy the Nyquist condition.

====== PARAMETERS ==

```
lambda = 3e-3;          % Wavelength (m)
w0 = 10e-3;             % Beam waist (m)
k  = 2*pi/lambda;        % Wavenumber
z0 = pi*w0^2/lambda;    % Rayleigh range
A0 = 1;
```

====== SAMPLING GRID =======================================

```
L  = 40*w0;
dx_min = sqrt(2)*pi/k ;
dx = dx_min + 0.1e-3 ;
x = -L/2:dx:L/2;
y = x;
[X,Y] = meshgrid(x,y);
N  = length(x);
```

====== SPATIAL FREQUENCY GRID ===============================

```
dk = 2*pi/dx;
kx =linspace(-dk/2,dk/2,N);
ky = kx;
[KX,KY] = meshgrid(kx,ky);
KZ_squared = k^2 - KX.^2 - KY.^2;
```

```
KZ = sqrt(max(0,KZ_squared));
```

====== INITIAL GAUSSIAN FIELD ==================================

```
U0 = A0 * exp(-(X.^2 + Y.^2)/w0^2);
```

====== PART 1 ==

PROPAGATION AT :0, 0.5 z0,z0

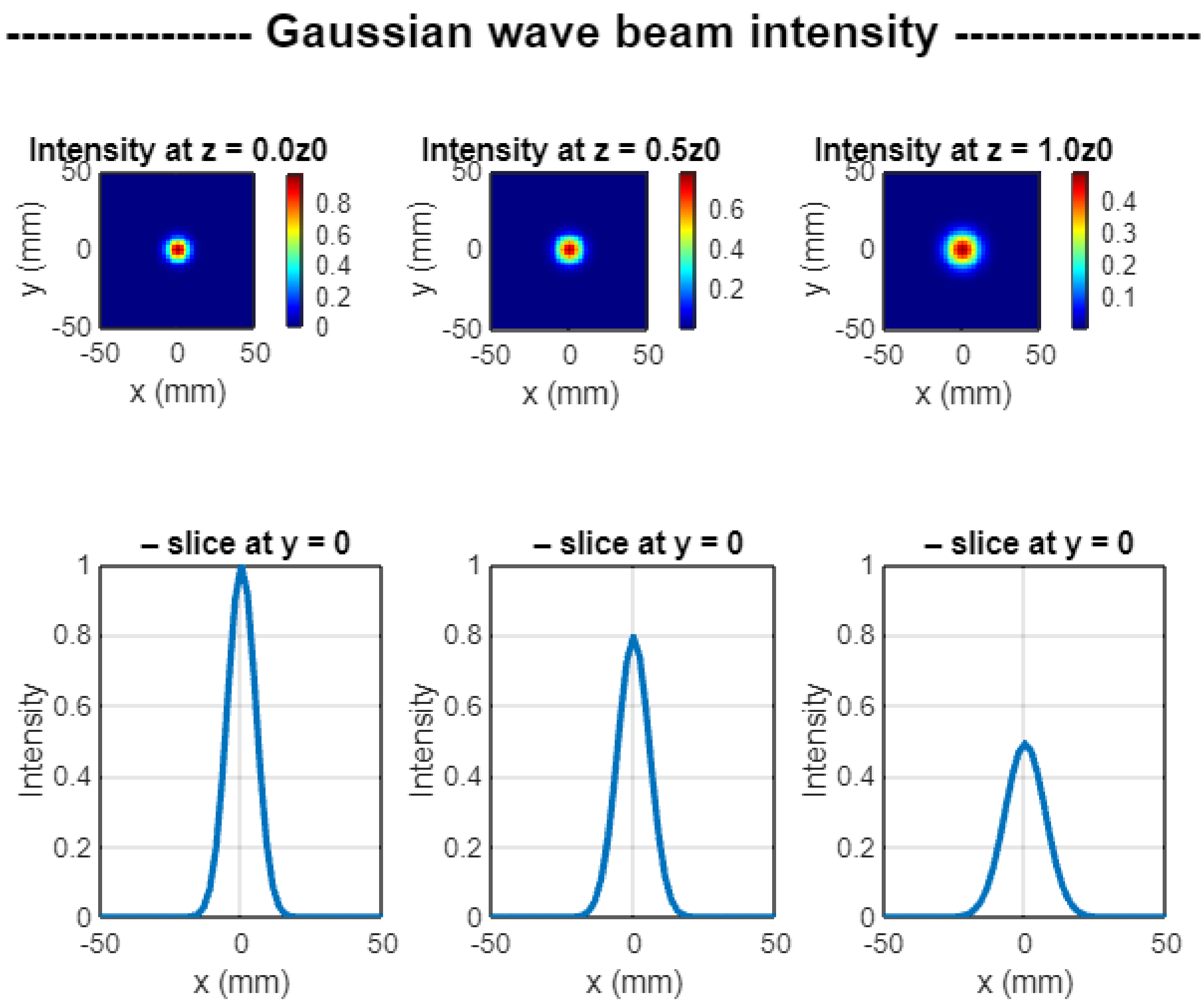
```
z_list = [0, 0.5*z0, z0];


for z = z_list
    Uz = propagate_fft(U0, z, KZ);
    plot_intensity(Uz, x, z/z0, sprintf('Intensity at z = %.1fz0', z/z0));
end
```

====== FUNCTION DEFINITIONS ===================================

```
function Uz = propagate_fft(U, z, KZ)
    H = exp(-1i * KZ * z);
    Uz = ifft2(ifftshift(fftshift(fft2(U)) .* H));
end
```

b) Results & Snapshots:



c) Behavior Analysis:

1. *At* $z = 0$ *(Beam Waist)*
   - **Beam Width:** $w(0) = w_0 = 10$ mm
   - **Wavefront Curvature:** $R(0) = \infty$ (plane wavefront)
   - **On-axis Intensity:** $I_0 = | A_0 |^2 = 1$
   - **Behavior:** The beam has the smallest cross-section and maximum on-axis intensity. The intensity profile is a perfect Gaussian with no curvature.

2. *At* $z = 0.5z_0 \approx 52.35$ *mm*
   - **Beam Width:** $w(0.5z_0) = w_0\sqrt{1 + 0.5^2} \approx 11.18$ mm
   - **Wavefront Curvature:** $R(0.5z_0) = 0.5z_0[1 + (1/0.5)^2] \approx 104.7$ mm (slightly curved)

- **On-axis Intensity:** $I(0{,}0.5z_0) = I_0/(1 + 0.5^2) \approx 0.8$
- **Behavior:** The beam has expanded by ~12%. The wavefront begins to curve, and the on-axis intensity drops to 80% of its peak value.

3. *At* $z = z_0 \approx 104.7\ mm$
   - **Beam Width:** $w(z_0) = w_0\sqrt{2} \approx 14.14$ mm
   - **Wavefront Curvature:** $R(z_0) = z_0[1{+}1] = 2z_0 \approx 209.4$ mm
   - **On-axis Intensity:** $I(0, z_0) = I_0/2 = 0.5$
   - **Behavior:** The beam has expanded by ~41%. The wavefront curvature is at its minimum radius (most curved). The on-axis intensity is half of the peak value, marking the boundary of the near-field (collimated) region.

d) Observations:

The 2D intensity plots show circular symmetric Gaussian profiles that widen with z. The transverse slices at $y = 0$ confirm the Gaussian shape and the reduction in peak intensity with propagation. The simulations match the theoretical predictions, validating the Fourier optics approach.

PART2: Parabolic Mirror Reflection and Further Propagation:

This section simulates the beam's interaction with a parabolic mirror and subsequent propagation. The mirror has a focal length $f = -4z_0$ (negative for a concave mirror).

a) Key Parameters:

- Mirror focal length: $f = -4z_0 \approx -418.9$ mm
- Incident beam distances from waist: $z_b = [3z_0, 4z_0, 5z_0] \approx [314.1,418.8,523.5]$ mm
- Propagation distances after reflection: $z_a = [z_0, 4z_0, 6z_0] \approx [104.7,418.8,628.2]$ mm

b) Code Implementation:

====== PARAMETERS ==

```
lambda = 3e-3;          % Wavelength (m)
w0 = 10e-3;             % Beam waist (m)
k  = 2*pi/lambda;        % Wavenumber
z0 = pi*w0^2/lambda;    % Rayleigh range
A0 = 1;
```

====== SAMPLING GRID ==

```
L  = 40*w0;
dx_min = sqrt(2)*pi/k ;
dx = dx_min + 0.1e-3 ;
x = -L/2:dx:L/2;
y = x;
[X,Y] = meshgrid(x,y);
N  = length(x);
```

====== SPATIAL FREQUENCY GRID ===============================

```
dk = 2*pi/dx;
kx =linspace(-dk/2,dk/2,N);
ky = kx;
[KX,KY] = meshgrid(kx,ky);
KZ_squared = k^2 - KX.^2 - KY.^2;
KZ = sqrt(max(0,KZ_squared));
```

====== INITIAL GAUSSIAN FIELD ===============================

```
U0 = A0 * exp(-(X.^2 + Y.^2)/w0^2);
```

====== PART 2 – PARABOLIC MIRROR (f = -4z0) ==================

PROPAGATION TO: 3z0, 4z0, 5z0 → REFLECT TO→ z0, 4z0, 6z0

```
f = -4*z0;
M = exp(-1i * k * (X.^2 + Y.^2) / (2*f));


z_before = [3*z0, 4*z0, 5*z0];
z_after_list = [z0, 4*z0, 6*z0];


for zb = z_before
    U_before = propagate_fft(U0, zb, KZ);
    U_after = M .* U_before;

    for za = z_after_list
        Uout = propagate_fft(U_after, za, KZ);
        plot_intensity_mirror(Uout, x,zb/z0,za/z0, ...
            sprintf('Intensity at za = %.0fz0', za/z0));
    end
end
```

====== FUNCTION DEFINITIONS ===================================

```
function Uz = propagate_fft(U, z, KZ)
    H = exp(-1i * KZ * z);
    Uz = ifft2(ifftshift(fftshift(fft2(U)) .* H));
end
```

c) Physical Interpretation:

1. **Beam Condition at Mirror Plane:**

   - At $z_b = 3z_0$: Wavefront radius $R(3z_0) = 3z_0[1 + (1/3)^2] \approx 3.33z_0 \approx 349$ mm
   - At $z_b = 4z_0$: $R(4z_0) = 4z_0[1 + (1/4)^2] \approx 4.25z_0 \approx 445$ mm
   - At $z_b = 5z_0$: $R(5z_0) = 5z_0[1 + (1/5)^2] \approx 5.2z_0 \approx 545$ mm

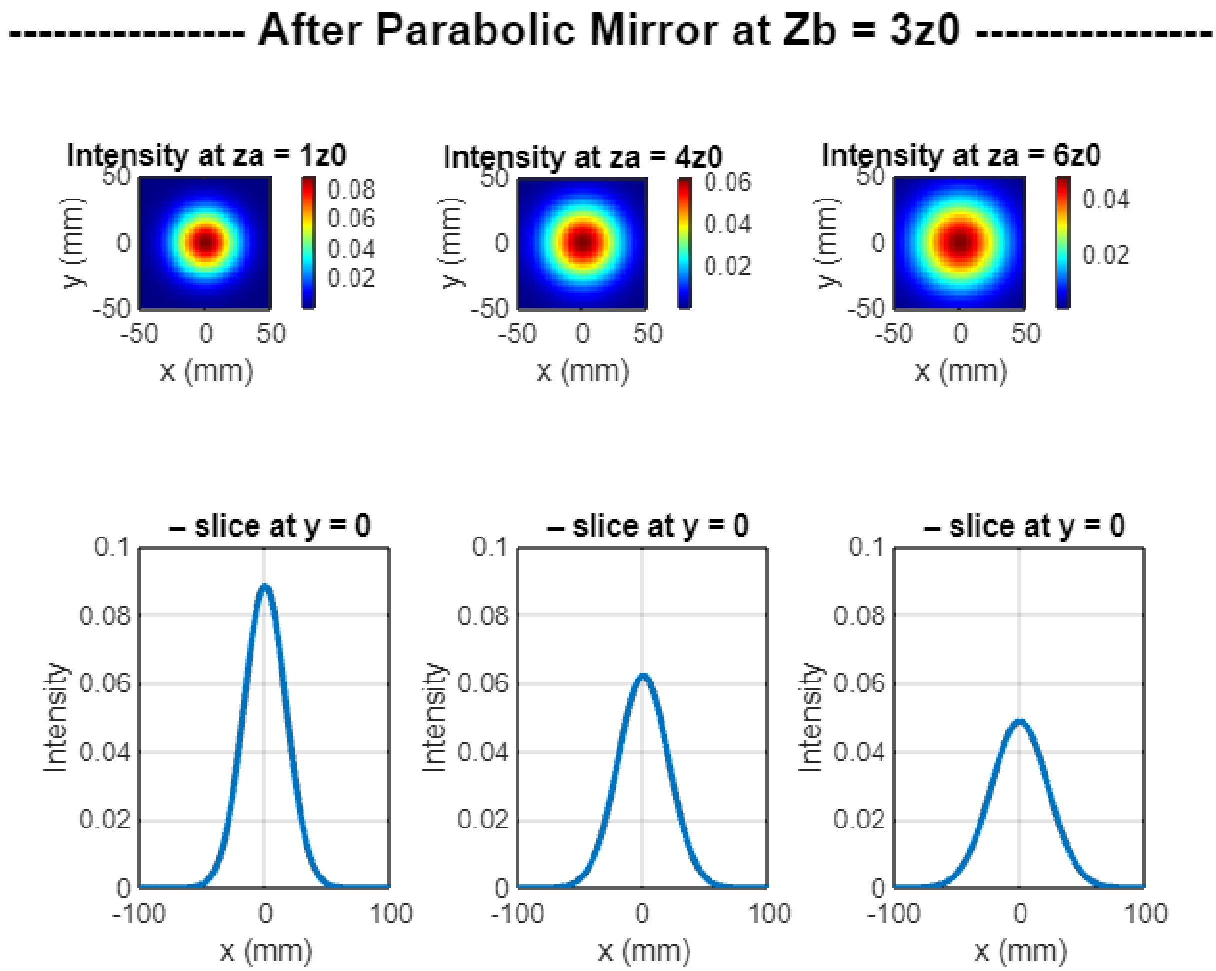
2. **Mirror Effect:**
   The concave mirror ($f = -418.9$ mm) applies a quadratic phase that modifies the beam's wavefront curvature. According to the mirror formula:

$$\frac{1}{R'} = \frac{1}{R} - \frac{1}{f}$$

where R is the incident wavefront curvature and R' is the reflected wavefront curvature.

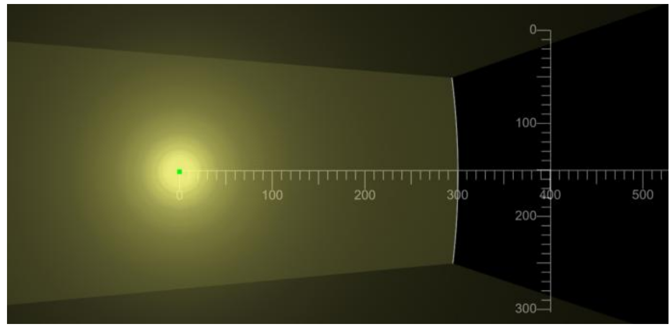
d) Simulation Results Analysis:

Case 1: $z_b = 3z_0$ (Incident beam slightly converging relative to mirror)

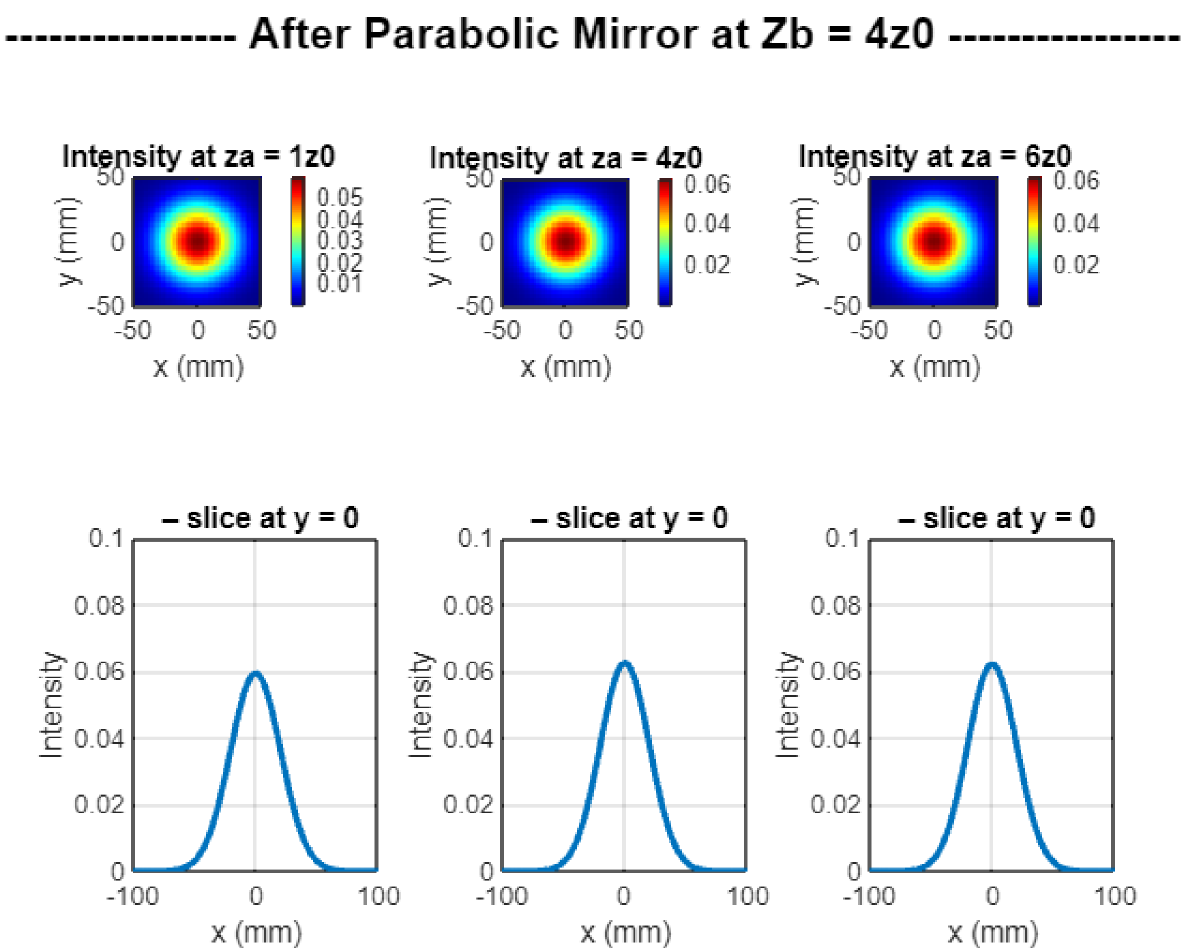


- Incident curvature: $R \approx 349$ mm
- Reflected curvature: $1/R' = 1/349 - 1/(-418.9) \Rightarrow R' \approx -1876$ mm (weakly diverging)
- After propagation to $z_a = z_0$: Beam continues to diverge gently
- After propagation to $z_a = 4z_0$: Beam expands significantly
- After propagation to $z_a = 6z_0$: Further expansion with lower peak intensity

- Figure demonstrates the propagation after reflection from parabolic mirror at 3z0:

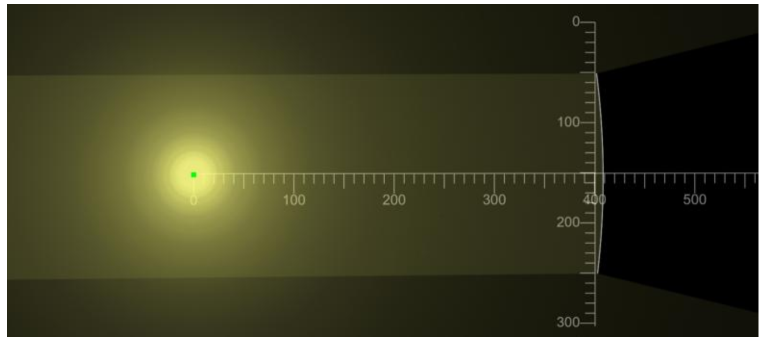


Case 2: $z_b = 4z_0$ (Incident beam curvature matches mirror focal length)

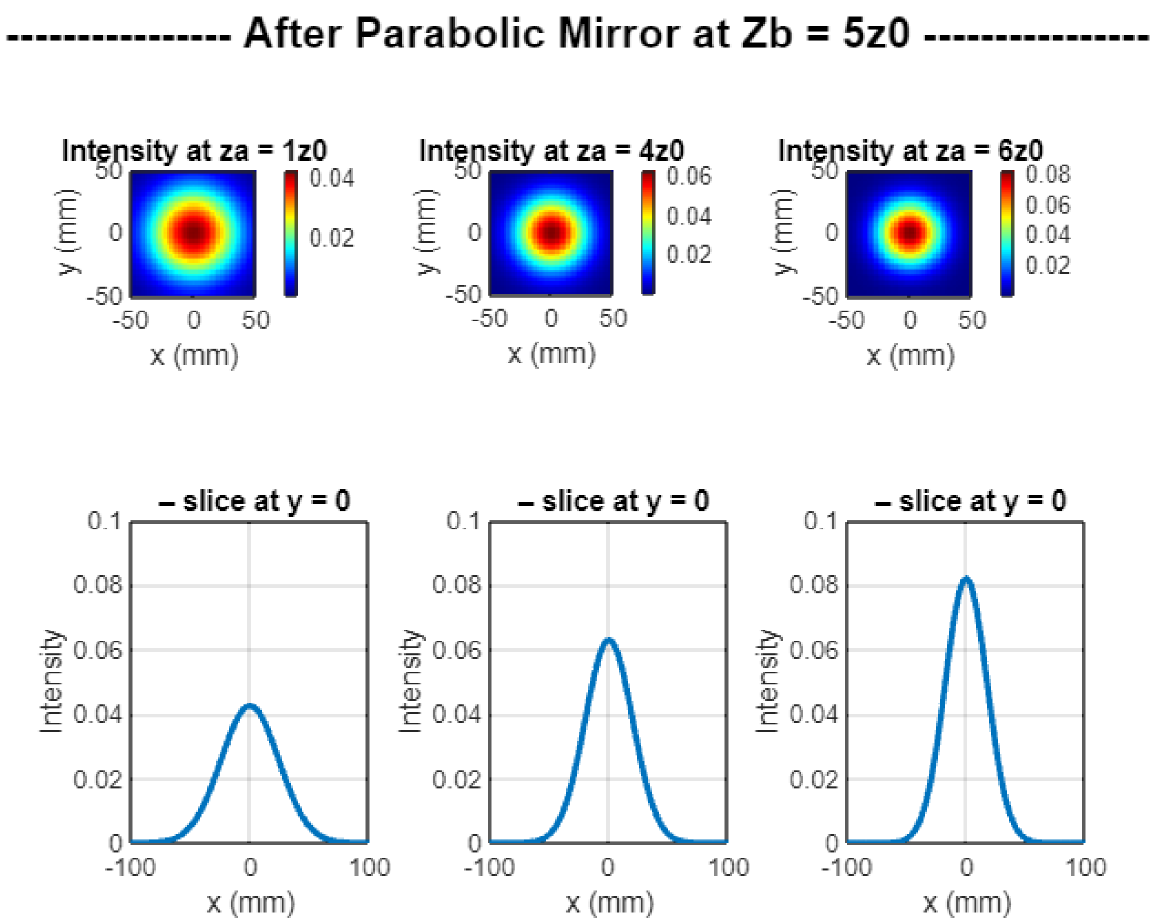


- Incident curvature: $R \approx 445$ mm $\approx |f|$
- Reflected curvature: $1/R' = 1/445 - 1/(-418.9) \Rightarrow R' \approx -215$ mm (strongly converging)
- The mirror nearly collimates the beam
- After propagation: Beam maintains relatively constant width with minor expansion

- Figure demonstrates the propagation behavior after reflection from parabolic mirror at 4z0:

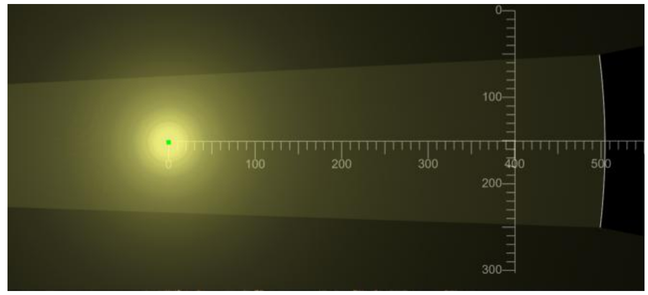


Case 3: $z_b = 5z_0$ (Incident beam more diverging than mirror focal condition)



- Incident curvature: $R \approx 545$ mm
- Reflected curvature: $1/R' = 1/545 - 1/(-418.9) \Rightarrow R' \approx -237$ mm (converging)
- The mirror over-compensates, creating a converging beam
- After propagation: Beam focuses then diverges, showing minimum beam width at intermediate distances

- Figure demonstrates the propagation behavior after reflection from parabolic mirror at 5z0:



e) Observations:

- The mirror's effectiveness in collimating/focusing depends on the incident wavefront curvature relative to its focal length
- When $R \approx |f|$, the mirror produces near-collimation
- For $R < |f|$, the reflected beam diverges
- For $R > |f|$, the reflected beam converges
- The beam remains Gaussian throughout, confirming the ABCD law predictions
- Peak intensity decreases with propagation distance due to beam expansion

❖CONCLUSION:

This project successfully implemented a Fourier optics-based simulation of Gaussian beam propagation and reflection, yielding results that align closely with theoretical predictions. Key insights and achievements include:

- **Beam Propagation Validation**: The simulated intensity profiles at $z = 0, 0.5z_0$, and z_0 showed the expected beam expansion, with the beam width increasing as $w(z) = w_0\sqrt{1 + (z/z_0)^2}$. The on-axis intensity decreased accordingly, confirming the theoretical behavior.

- **Mirror Transformation**: The parabolic mirror (with $f = 4z_0$) effectively altered the beam's curvature. For incident beams at different distances ($3z_0, 4z_0, 5z_0$), the reflected beam either converged, became collimated, or diverged, depending on the incident wavefront curvature relative to the mirror's focal length. This demonstrated the mirror's ability to act as a focusing/defocusing element.

- **Fourier Optics Efficiency**: The use of FFT2/IFFT2 provided a computationally efficient and accurate means to simulate wave propagation over significant distances without solving differential equations directly. The method naturally incorporates diffraction effects.

- **Physical Interpretation**: The results illustrate fundamental concepts in beam optics:

- The Gaussian beam maintains its shape but scales in width.
- A parabolic phase imprint (mirror) can redirect and reshape the beam.
- The Rayleigh range z_0 is a key parameter governing beam divergence and depth of focus.
- The ABCD law accurately predicts beam transformation by optical elements.

▪ **Learning Outcomes**: The exercise deepened understanding of:

- Gaussian beam parameters and their evolution.
- The angular spectrum method and its numerical implementation.
- The ABCD law for beam transformation.
- The role of phase-only elements (like mirrors and lenses) in wavefront engineering.
- Numerical considerations in Fourier optics simulations.

This simulation serves as a foundation for more advanced studies in beam shaping, laser resonator design, and Fourier optics applications. The methods employed are directly applicable to modeling complex optical systems, including those with multiple elements or aberrations. By bridging theory, computation, and physical interpretation, the project reinforces core principles of modern optical engineering.

❖MATLAB Code:

==

Gaussian Beam Simulation Using FFT Propagation

lambda = 3 mm and w0 = 10 mm

==

```
clear; close all; clc;
```

====== PARAMETERS ==

```
lambda = 3e-3;          % Wavelength (m)
w0 = 10e-3;             % Beam waist (m)
k  = 2*pi/lambda;        % Wavenumber
z0 = pi*w0^2/lambda;    % Rayleigh range
A0 = 1;
```

====== SAMPLING GRID ======================================

```
L  = 40*w0;
dx_min = sqrt(2)*pi/k ;
dx = dx_min + 0.1e-3 ;
x = -L/2:dx:L/2;
y = x;
[X,Y] = meshgrid(x,y);
N  = length(x);
```

====== SPATIAL FREQUENCY GRID ============================

```
dk = 2*pi/dx;
kx =linspace(-dk/2,dk/2,N);
ky = kx;
[KX,KY] = meshgrid(kx,ky);
KZ_squared = k^2 - KX.^2 - KY.^2;
KZ = sqrt(max(0,KZ_squared));
```

====== INITIAL GAUSSIAN FIELD =============================

```
U0 = A0 * exp(-(X.^2 + Y.^2)/w0^2);
```

====== PART 1 ===

PROPAGATION AT :0, 0.5 z0,z0

```
z_list = [0, 0.5*z0, z0];


for z = z_list
    Uz = propagate_fft(U0, z, KZ);
    plot_intensity(Uz, x, z/z0, sprintf('Intensity at z = %.1fz0', z/z0));
end
```

====== PART 2 – PARABOLIC MIRROR (f = -4z0) ===================

PROPAGATION TO: 3z0, 4z0, 5z0 → REFLECT TO→ z0, 4z0, 6z0

```
f = -4*z0;
M = exp(-1i * k * (X.^2 + Y.^2) / (2*f));


z_before = [3*z0, 4*z0, 5*z0];
z_after_list = [z0, 4*z0, 6*z0];


for zb = z_before
    U_before = propagate_fft(U0, zb, KZ);
    U_after = M .* U_before;

    for za = z_after_list
        Uout = propagate_fft(U_after, za, KZ);
        plot_intensity_mirror(Uout, x,zb/z0,za/z0, ...
            sprintf('Intensity at za = %.0fz0', za/z0));
    end
end
```

====== FUNCTION DEFINITIONS ==================================

```
function Uz = propagate_fft(U, z, KZ)
    H = exp(-1i * KZ * z);
    Uz = ifft2(ifftshift(fftshift(fft2(U)) .* H));
end


function plot_intensity(U, x,index, title_str)
I = abs(U).^2;


if index == 0
    position = 1;
    figure;
    sgtitle('---------------- Gaussian wave beam intensity ----------------
','FontWeight','bold');
elseif index == 0.5
```

```
    position = 2;
elseif index == 1
    position = 3;
end
subplot(2,3,position);
x_mm = x*1e3;
imagesc(x_mm, x_mm, I);
axis xy;axis equal;colormap jet; colorbar;shading interp;
xlabel('x (mm)');ylabel('y (mm)');
title(title_str);
xlim([-50,50]);
ylim([-50,50]);


% Slice at y=0
idx = round(length(x)/2);
I_x = I(idx,:);
subplot(2,3,position+3);
plot(x_mm, I_x, 'LineWidth', 2);
xlabel('x (mm)');ylabel('Intensity');
title(' - slice at y = 0');
ylim([0,1]);
xlim([-50,50]);
grid on;
end


function plot_intensity_mirror(U, x,index1,index2, title_str)
I = abs(U).^2;


if index1 == 3 && index2==1
    figure;
    sgtitle('---------------- After Parabolic Mirror at Zb = 3z0 ----------------
','FontWeight','bold');
elseif index1 == 4 && index2==1
    figure;
    sgtitle('---------------- After Parabolic Mirror at Zb = 4z0 ----------------
','FontWeight','bold');
elseif index1 == 5 && index2==1
    figure;
    sgtitle('---------------- After Parabolic Mirror at Zb = 5z0 ----------------
','FontWeight','bold');
end
if index2 == 1
    position = 1;
elseif index2 == 4
    position = 2;
elseif index2 == 6
    position = 3;
```

```
end
subplot(2,3,position);
x_mm = x*1e3;
imagesc(x_mm, x_mm, I);
axis xy;axis equal;colormap jet; colorbar;shading interp;
xlabel('x (mm)');ylabel('y (mm)');
title(title_str);
xlim([-50,50]);
ylim([-50,50]);


% Slice at y=0
idx = round(length(x)/2);
I_x = I(idx,:);
subplot(2,3,position+3);
plot(x_mm, I_x, 'LineWidth', 2);
xlabel('x (mm)');ylabel('Intensity');
title(' - slice at y = 0');
ylim([0,0.1]);     % auto-scale to max intensity
xlim([-100,100]);
grid on;
end
```

❖ References:

Bahaa E. A. Saleh, and Malvin Carl Teich, “Fundamentals of Photonics”, 2nd Edition, John Wiley and sons, Inc., ISBN 978-0-471-35832-9.

❖ GitHub Link:

Click here