Area between two curves by using matrix method

Deepak Ahirwar
EE24BTECH11014
Electrical Enggineering ,
IIT Hyderabad.

November 6, 2024

Problem Statement

Find the area of the circle $4x^2 + 4y^2 = 9$ which is interior to the parabola $x^2 = 4y$.

Variable	**Description**
V_1, u_1, f_1	Parameters of Parabola
V_2, u_2, f_2	Parameters of circle
P_1, P_2	Points of intersection
A	Area between the conics

Table: Variables Used

Conic Parameters

The conic parameters of circle $4x^2 + 4y^2 = 9$ are :

$$V_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, u_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, f_1 = -\frac{9}{4}$$

Conic parameters of parabola $x^2 = 4y$ can be expressed as :

$$V_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, u_2 = \begin{pmatrix} 0 \\ -2 \end{pmatrix}, f_2 = 0$$

Intersection of conics

The intersection of two conics with parameters $V_i, u_i, f_i (i = 1, 2)$ is defined as :

$$x^\top (V_1 + \mu V_2) x + 2(u_1 + \mu u_2)^\top x + (f_1 + \mu f_2) = 0$$

On solving we get the points of intersection are :

$$\begin{pmatrix} \sqrt{2} \\ \frac{1}{2} \end{pmatrix}, \begin{pmatrix} -\sqrt{2} \\ \frac{1}{2} \end{pmatrix}$$

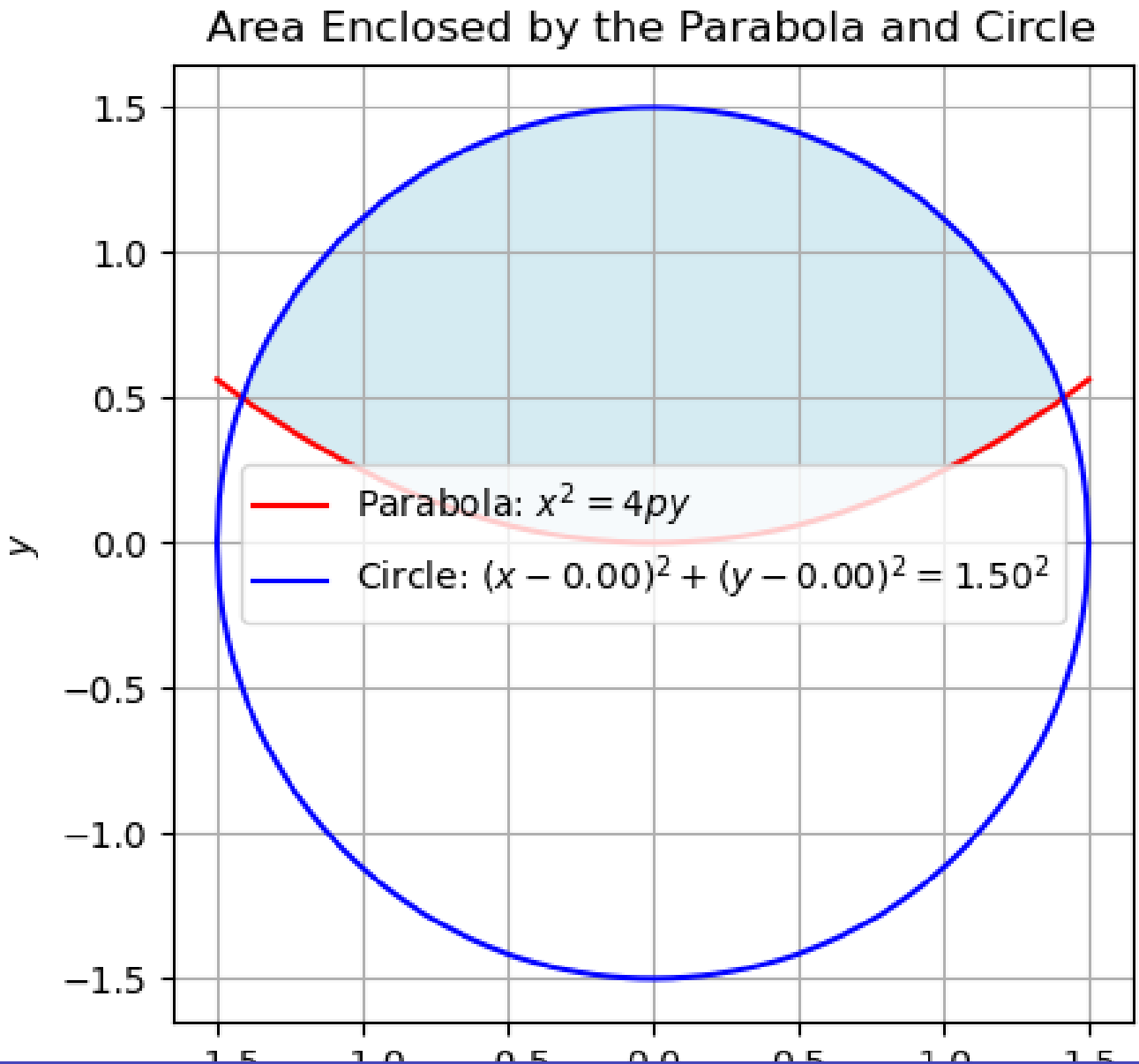
Area calculation

Area between the curves is,

$$2\int_0^{\frac{1}{2}} \left(\sqrt{\frac{9}{4}-y^2}-\sqrt{4y}\right)\,dy \tag{3.1}$$

By solving the integration, we get area is equal to 3.005 sq.units

Figure


Area Enclosed by the Parabola and Circle
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
y
Parabola: $x^2 = 4py$
Circle: $(x - 0.00)^2 + (y - 0.00)^2 = 1.50^2$

C Code

```
#include <stdio.h>

int main() {
    // Define parabola and circle parameters
    // Parabola equation: x^2 = 4py, so y = x^2 / (4 * p)
    double p = 1.0; // Parabola parameter (for equation x^2 = 4py)

    // Circle equation: (x - h)^2 + (y - k)^2 = r^2
    double r = 1.5; // Radius of the circle
    double h = 0.0; // Center of the circle (x-coordinate)
    double k = 0.0; // Center of the circle (y-coordinate)

    // Open the file to write the parameters
    FILE *file = fopen("data.txt", "w");
    if (file == NULL) {
        printf("Error opening file!\n");
        return 1;
```

```
}

    // Write the parabola parameter to the file
    fprintf(file, "%f\n", p); // Parabola parameter (p in x^2 = 4py)

    // Write the circle parameters to the file
    fprintf(file, "%f\n", r); // Circle radius
    fprintf(file, "%f\n", h); // Circle center x-coordinate
    fprintf(file, "%f\n", k); // Circle center y-coordinate

    // Close the file
    fclose(file);

    printf("Data successfully written to data.txt\n");

    return 0;
}
```

Python Code for Plotting

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import quad
from scipy.optimize import fsolve

import sys # For path to external scripts
sys.path.insert(0, '/home/deepak/EE1030/matgeo/codes/CoordGeo') #
    Path to my scripts

# Read the values from the C-generated text file using numpy.loadtxt
data = np.loadtxt('data.txt')

# Extracting parabola and circle parameters
p = data[0] # Parabola parameter (x^2 = 4py)
r = data[1] # Circle radius
h = data[2] # Circle center x-coordinate
```

```
k = data[3] # Circle center y-coordinate

# Parabola equation: x^2 = 4py, so y = x^2 / (4p)
def parabola(x, p):
    return x**2 / (4 * p)

# Circle equation: (x - h)^2 + (y - k)^2 = r^2, so y = k + sqrt(r^2 -
    (x - h)^2)
def circle(x, r, h, k):
    return k + np.sqrt(r**2 - (x - h)**2)

# Find the points of intersection between the parabola and the circle
def find_intersections(p, r, h, k):
    def intersection_eq(x):
        return circle(x, r, h, k) - parabola(x, p)

    x_int1 = fsolve(intersection_eq, -r)[0]
    x_int2 = fsolve(intersection_eq, r)[0]
```

```
    return x_int1, x_int2

# Get the intersection points
x_int1, x_int2 = find_intersections(p, r, h, k)

# Compute the area between the curves using integration
def area_between_curves(x, p, r, h, k):
    return circle(x, r, h, k) - parabola(x, p)

# Perform the integration from x_int1 to x_int2
area, _ = quad(area_between_curves, x_int1, x_int2, args=(p, r, h, k))

print(f"Area enclosed between the parabola and the circle: {area}")
# Generating points for the parabola and circle
x_vals = np.linspace(-r, r, 400)
y_parabola = parabola(x_vals, p)
y_circle_upper = circle(x_vals, r, h, k)
# Generate the lower half of the circle
```

```
 #Generate the lower half of the circle
y_circle_lower = k - np.sqrt(r**2 - (x_vals - h)**2)
# Plot the curves
plt.plot(x_vals, y_parabola, label=r'Parabola: $x^2 = 4py$', color='r')
plt.plot(x_vals, y_circle_upper, label=r'Circle: $(x - %.2f)^2 + (y - %.2
    f)^2 = %.2f^2$' % (h, k, r), color='b')
plt.plot(x_vals, y_circle_lower, color='b') # Lower part of the circle (no
    extra label)

# Fill the area between the curves
plt.fill_between(x_vals, y_parabola, y_circle_upper, where=(y_circle_upper
    >= y_parabola), color='lightblue', alpha=0.5)

# Labels and plot settings
plt.xlabel('$x$')
plt.ylabel('$y$')
plt.title('Area Enclosed by the Parabola and Circle')
plt.grid(True)
plt.legend()
```