Matgeo 9-9.3-9

EE24BTECH11010-Balaji B

November 6, 2024

Question

If the area of the region rounded by the curve $y^2 = 4ax$ and the line $x = 4a$ is $\frac{256}{3}$ sq.units then using integration, find the value of a , where $a > 0$.

Parameters

The parameters for the problem are given as follows:

Parameter	**Description**
$\mathbf{x_1}$	First intersection point
$\mathbf{x_2}$	Second intersection point
$\mathbf{h}$	Point on the given line
$\mathbf{m}$	Direction vector of given line
A	Area of the region

Table: Parameter Used

Intersection Points

The equation of a parabola in Matrix form is

$$\mathbf{x}^\top \mathbf{V}\mathbf{x} + 2\mathbf{u}^\top \mathbf{x} + f = 0 \tag{3.1}$$

The equation of a line in vector form is

$$\mathbf{x} = \mathbf{h} + \kappa \mathbf{m} \tag{3.2}$$

For the given parabola $y^2 = 4ax$, The values of $\mathbf{V}$,$\mathbf{u}$,f are

$$\mathbf{V} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \tag{3.3}$$

$$\mathbf{u} = \begin{pmatrix} -2a \\ 0 \end{pmatrix} \tag{3.4}$$

$$f = 0 \tag{3.5}$$

For the given line $x = 4a$, The values of $\mathbf{h}$, $\mathbf{m}$ are

$$\mathbf{h} = \begin{pmatrix} 4a \\ 0 \end{pmatrix} \tag{3.6}$$

$$\mathbf{m} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{3.7}$$

Substituting the line equation in parabola equation gives the values of κ

$$(\mathbf{h} + \kappa\mathbf{m})^\top \mathbf{V} (\mathbf{h} + \kappa\mathbf{m}) + 2\mathbf{u}^\top (\mathbf{h} + \kappa\mathbf{m}) + f = 0 \tag{3.8}$$

$$\left(\begin{pmatrix} 4a \\ 0 \end{pmatrix} + \kappa\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right)^\top \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \left(\begin{pmatrix} 4a \\ 0 \end{pmatrix} + \kappa\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right) + 2\begin{pmatrix} -2a \\ 0 \end{pmatrix}^\top \left(\begin{pmatrix} 4a \\ 0 \end{pmatrix} + \kappa\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right) + 0 = 0 \tag{3.9}$$

$$\begin{pmatrix} 4a & \kappa \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 4a \\ \kappa \end{pmatrix} + 2\begin{pmatrix} -2a & 0 \end{pmatrix} \begin{pmatrix} 4a \\ \kappa \end{pmatrix} = 0 \tag{3.10}$$

$$\begin{pmatrix} 4a & \kappa \end{pmatrix} \begin{pmatrix} 0 \\ \kappa \end{pmatrix} + 2\left(-8a^2\right) = 0 \tag{3.11}$$

$$\kappa^2 - 16a^2 = 0 \tag{3.12}$$

$$\kappa_1 = 4a \tag{3.13}$$

$$\kappa_2 = -4a \tag{3.14}$$

The intersection points are

$$\mathbf{x_1} = \mathbf{h} + \kappa_1 \mathbf{m} \tag{3.15}$$

$$\mathbf{x_1} = \begin{pmatrix} 4a \\ 4a \end{pmatrix} \tag{3.16}$$

$$\mathbf{x_2} = \mathbf{h} + \kappa_2 \mathbf{m} \tag{3.17}$$

$$\mathbf{x_2} = \begin{pmatrix} 4a \\ -4a \end{pmatrix} \tag{3.18}$$

Area Calculation

The Area under the curve is given by

$$A = \int_{-4a}^{4a} \left(\frac{y^2}{4a} \right) dy \tag{3.19}$$

$$A = \left(\frac{1}{4a} \right) \left(\frac{(4a)^3 - ((-4a)^3)}{3} \right) \tag{3.20}$$

$$A = \left(\frac{1}{12a} \right) \left(128a^3 \right) \tag{3.21}$$

$$A = \frac{32a^2}{3} \tag{3.22}$$

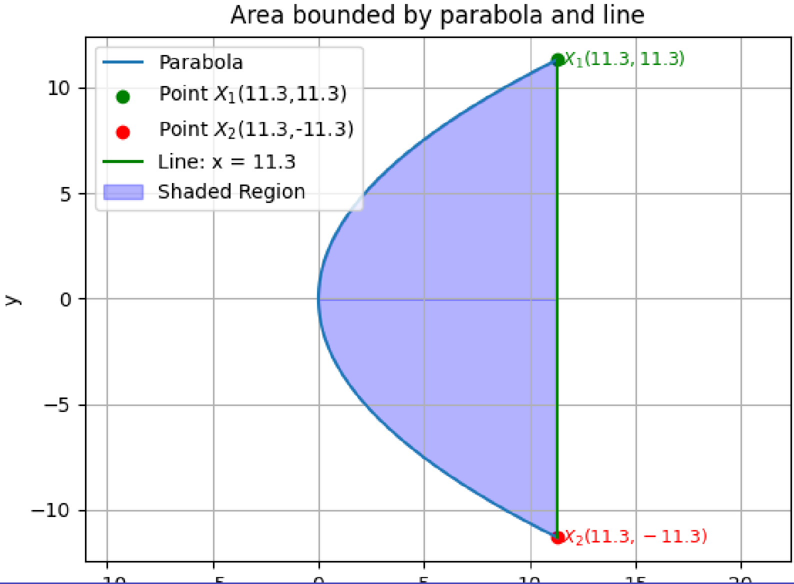
Finding a

The area of region bounded by the line $x = 4a$ and the parabola $y^2 = 4ax$ is given as $\frac{256}{3}$
Comparing the above area with the given area we get, the value of a as

$$\frac{32a^2}{3} = \frac{256}{3} \tag{3.23}$$

$$a = 2\sqrt{2} \tag{3.24}$$

Graph


Area bounded by parabola and line
Parabola
Point X_1(11.3,11.3)
Point X_2(11.3,-11.3)
Line: x = 11.3
Shaded Region
$X_1(11.3, 11.3)$
$X_2(11.3, -11.3)$
10
5
0
−5
−10
y

C Code

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>

#include "libs/matfun.h"
#include "libs/geofun.h"

typedef struct
{
    double **p;
}Point;
```

```
typedef struct
{
    double **V;
    double **u;
    double f;
}Parabola;

void InitializePoint(Point *point)
{
    point->p = createMat(2,1);
}

void InitializeParabola(Parabola *parabola)
{
    parabola->V = createMat(2,2);
    parabola->u = createMat(2,1);
    parabola->f = 0;
}
```

```
void parab_y2_4ax_gen(FILE *fptr, Point *x1, Point *x2, double a, int
    num_points)
{
    double tinit = (x1->p[0][0] / (2 * a));
    double tfinal = (x2->p[1][0] / (2 * a));

    Point coord;
    InitializePoint(&coord);
    if (tfinal > tinit)
    {
        for (double i = tinit; i <= tfinal; i += 1.0 / num_points)
        {
            coord.p[0][0] = a*i*i;
            coord.p[1][0] = 2*a*i;
            fprintf(fptr,"%lf,%lf\n", coord.p[0][0], coord.p[1][0] );
        }
    }
    else
```

```
 {
        for (double i = tfinal; i <= tinit; i += 1.0 / num_points)
        {
             coord.p[0][0] = a*i*i;
             coord.p[1][0] = 2*a*i;
             fprintf(fptr,"%lf,%lf\n", coord.p[0][0], coord.p[1][0] );
        }
    }
}

Point* intersect_of_parab_line(Parabola *parabola, Point *point_line, Point
      *slope)
{
    double m1 = slope->p[0][0];
    double m2 = slope->p[1][0];

    double h1 = point_line->p[0][0];
    double h2 = point_line->p[1][0];
```

```
double V1 = parabola->V[0][0];
    double V2 = parabola->V[1][0];
    double V3 = parabola->V[0][1];
    double V4 = parabola->V[1][1];

    double u1 = parabola->u[0][0];
    double u2 = parabola->u[1][0];

    double A = V1 * m1 * m1 + (V2 + V3) * m1 * m2 + V4 * m2 *
        m2;
    double B = 2 * (m1 * (V1 * h1 + V2 * h2 + u1) + m2 * (V3 * h1
        + V4 * h2 + u2));
    double C = h1 * (V1 * h1 + V2 * h2) + h2 * (V3 * h1 + V4 * h2)
        + 2 * (u1 * h1 + u2 * h2) + parabola->f;

    if (B * B - 4 * A * C < 0)
    {
        return NULL;
    }
```

```
else
    {
        Point *intersection = malloc(2*sizeof(Point));
        InitializePoint(&intersection[0]);
        InitializePoint(&intersection[1]);

        if (B * B - 4 * A * C == 0) {
            double k = -B / (2 * A);
            intersection[0].p[0][0] = point_line->p[0][0] + k * slope->p
                [0][0];
            intersection[0].p[1][0] = point_line->p[1][0] + k * slope->p
                [1][0];
            return intersection;
        }
         else
        {
            double k1 = (-B + sqrt(B * B - 4 * A * C)) / (2 * A);
            double k2 = (-B - sqrt(B * B - 4 * A * C)) / (2 * A);
```

```
intersection[0].p[0][0] = point_line->p[0][0] + k1 * slope->p[0][0];
            intersection[0].p[1][0] = point_line->p[1][0] + k1 * slope->
                p[1][0];
            intersection[1].p[0][0] = point_line->p[0][0] + k2 * slope->
                p[0][0];
            intersection[1].p[1][0] = point_line->p[1][0] + k2 * slope->
                p[1][0];
            return intersection;
            }
      }
}

int main()
{
    Parabola parabola;
    Point p1;
    Point p2;
```

```
Point slope;
Point point_line;

InitializeParabola(&parabola);
InitializePoint(&p1);
InitializePoint(&p2);
InitializePoint(&slope);

parabola.V[0][0] = 0;
parabola.V[0][1] = 0;
parabola.V[1][0] = 0;
parabola.V[1][1] = 1;
parabola.u[0][0] = 0;
parabola.u[1][0] = -4*sqrt(2);
parabola.f = 0;

point_line.p[0][0] = 8*sqrt(2);
point_line.p[1][0] = 0;
```

```
slope.p[0][0] = 0;
slope.p[1][0] = 1;

Point *intersection = intersect_of_parab_line(&parabola, &point_line,
    &slope);

p1.p[0][0] = intersection[0].p[0][0];
p1.p[1][0] = intersection[0].p[1][0];

p2.p[0][0] = intersection[1].p[0][0];
p2.p[1][0] = intersection[1].p[1][0];

FILE *fptr = fopen("parab.txt", "w");
if (fptr == NULL)
{
    printf("Error opening file!\n");
    return 1;
```

```
}

    parab_y2_4ax_gen(fptr, &p1, &p2, -parabola.u[1][0] / 2, 150);
    fclose(fptr);
    return 0;

}
```

Python Code

```
import numpy as np
import matplotlib.pyplot as plt
import math

# Load points from the file
points = np.loadtxt('parab.dat', delimiter=',', max_rows=len(list(open("
    ./parab.dat")))-1)

# Extract x and y values
x = points[:, 0]
y = points[:, 1]

# Define points A, B, C, D
x1 = np.array([11.3, 11.3])
x2 = np.array([11.3, -11.3])
```

```
# Plot the parabola
plt.plot(x, y, label='Parabola')

# Plot the specific points
plt.scatter(x1[0], x1[1], color='green', marker='o', label='Point $X_1$
    (11.3,11.3)')
plt.scatter(x2[0], x2[1], color='red', marker='o', label='Point $X_2$
    (11.3,-11.3)')

# Define the line equation x = 8*\sqrt2
line_y = np.linspace(min(y), max(y), 400)
line_x = line_y * 0 + 11.3

# Plot the line x=11.3
plt.plot(line_x, line_y, color='green', label='Line: x = 11.3')

plt.text(x1[0] + 0.2, x1[1] - 0.3, '$X_1(11.3,11.3)$', color='green',
    fontsize=9)
```

```
# Fill the closed region between the parabola and the line x=11.3
plt.fill_between(x, y, where=(x <= 11.3), color='blue', alpha=0.3, label=
    'Shaded Region')

# Label the axes and add a title
plt.xlabel("x")
plt.ylabel("y")
plt.title("Parabola, Line, and Points $X_1$ & $X_2$")
plt.grid(True)
plt.legend(loc='upper left')
plt.axis('equal')

# Save the plot to a file
plt.savefig('../figs/fig.png')
plt.show()
```