4.7.11

Vishwambhar - EE25BTECH11025

13th september, 2025

Question

Show that the path of a moving point such that its distance from two lines $3x - 2y = 5$ and $3x + 2y = 5$ are equal is a straight line.

Given

Given line equations can be written as:

$$\mathbf{n}_1^\top \mathbf{x} = c_1 \tag{1}$$

$$\mathbf{n}_1 = \begin{pmatrix} 3 \\ -2 \end{pmatrix}; c_1 = 5 \tag{2}$$

$$\mathbf{n}_2^\top \mathbf{x} = c_2 \tag{3}$$

$$\mathbf{n}_2 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}; c_2 = 5 \tag{4}$$

let the point equidistant from the given lines be:

$$\mathbf{P} = \begin{pmatrix} x \\ y \end{pmatrix} \tag{5}$$

Proof

From distance formula:

$$d_1 = \frac{|\mathbf{n}_1^\top \mathbf{P} - c_1|}{||\mathbf{n}_1||} \tag{6}$$

$$d_2 = \frac{|\mathbf{n}_2^\top \mathbf{P} - c_2|}{||\mathbf{n}_2||} \tag{7}$$

$$\because d_1 = d_2 \tag{8}$$

$$\frac{|\mathbf{n}_1^\top \mathbf{P} - c_1|}{||\mathbf{n}_1||} = \frac{|\mathbf{n}_2^\top \mathbf{P} - c_2|}{||\mathbf{n}_2||} \tag{9}$$

$$\because ||\mathbf{n}_1|| = ||\mathbf{n}_2|| = \sqrt{3^2 + 2^2} = \sqrt{13} \tag{10}$$

$$\mathbf{n}_1^\top \mathbf{P} - c_1 = \pm\left(\mathbf{n}_2^\top \mathbf{P} - c_2\right) \tag{11}$$

checking

First, by taking $+$:

$$\mathbf{n}_1^\top \mathbf{P} - c_1 = +\left(\mathbf{n}_2^\top \mathbf{P} - c_2\right) \tag{12}$$

$$\mathbf{n}_1^\top \mathbf{P} - \mathbf{n}_2^\top \mathbf{P} = c_1 - c_2 \tag{13}$$

$$\left(\mathbf{n}_1 - \mathbf{n}_2\right)^\top \mathbf{P} = c_1 - c_2 \tag{14}$$

$$\begin{pmatrix} 0 & -4 \end{pmatrix} \mathbf{P} = 0 \tag{15}$$

Now by taking $-$:

$$\mathbf{n}_1^\top \mathbf{P} - c_1 = -\left(\mathbf{n}_2^\top \mathbf{P} - c_2\right) \tag{16}$$

$$\mathbf{n}_1^\top \mathbf{P} + \mathbf{n}_2^\top \mathbf{P} = c_1 + c_2 \tag{17}$$

$$\left(\mathbf{n}_1 + \mathbf{n}_2\right)^\top \mathbf{P} = c_1 + c_2 \tag{18}$$

$$\begin{pmatrix} 6 & 0 \end{pmatrix} \mathbf{P} = 10 \tag{19}$$

Conclusion

Since equations (15) and (19) are in the form of line equation $\mathbf{n}^\top \mathbf{x} = c$, the given path of the moving point is a line.

C Code

```c
#include <stdio.h>

int vector1[2]={3, -2};
int constant1[1]={5};
int vector2[2]={3, 2};
int constant2[1]={5};

void give_data(double *points){
    points[0] = vector1[0];points[1] = vector1[1];
    points[2] = constant1[0];
    points[3] = vector2[0]; points[4] = vector2[1];
    points[5] = constant2[0];
}
```

C Code

```c
void give_findata(double *points2){
    double finalvector1[2]; double finalvector2[2];
    double finalconstant1; double finalconstant2;
    for(int i = 0; i<2; i++){
        finalvector1[i] = vector1[i] - vector2[i];
        finalvector2[i] = vector1[i] + vector2[i];
    }
    finalconstant1 = constant1[0] - constant2[0];
    finalconstant2 = constant1[0] + constant2[0];
    points2[0] = finalvector1[0];points2[1] = finalvector1[1];
    points2[2] = finalconstant1;
    points2[3] = finalvector2[0]; points2[4] = finalvector2[1];
    points2[5] = finalconstant2;
}
```

Python Code 1

```
import ctypes as ct

lib = ct.CDLL("./problem.so")
lib.give_data.argtypes = [ct.POINTER(ct.c_double)]
lib.give_findata.argtypes = [ct.POINTER(ct.c_double)]

points = ct.c_double*8
data = points()
lib.give_data(data)
finpoints = ct.c_double*8
findata = finpoints()
lib.give_findata(findata)

def send_data():
    return data, findata
```

Python Code 2

```
import matplotlib.pyplot as plt
import numpy as np
from call import send_data
data, findata = send_data()
a = np.linspace(-10, 10, 100)
b = ((data[0]*a)-data[2])/(-data[1])
A = np.linspace(-10, 10, 100)
B = ((-data[3]*A)+data[5])/data[4]
y = np.linspace(-10, 10, 100)
x = ((findata[4]*y)-findata[5])/(-findata[3])
X = np.linspace(-10, 10, 100)
Y = ((findata[0]*X)-findata[2])/(-findata[1])
```

Python Code 2

```
plt.plot(a, b, 'r-')
plt.plot(A, B, 'r-')
plt.plot(x, y, 'k-')
plt.plot(X, Y, 'k-')
plt.text(10, 12.3, "3x-2y=5", fontsize=10, color='black')
plt.text(-8.3, 15, "3x+2y=5", fontsize=10, color='black')
plt.text(15.2, -0.06, "y=0", fontsize=10, color='black')
plt.text(1.6, 14.6, "x=10/6", fontsize=10, color='black')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.axis('equal')
plt.grid(True)
plt.savefig("../figs/plot.png")
plt.show()
```

Plot

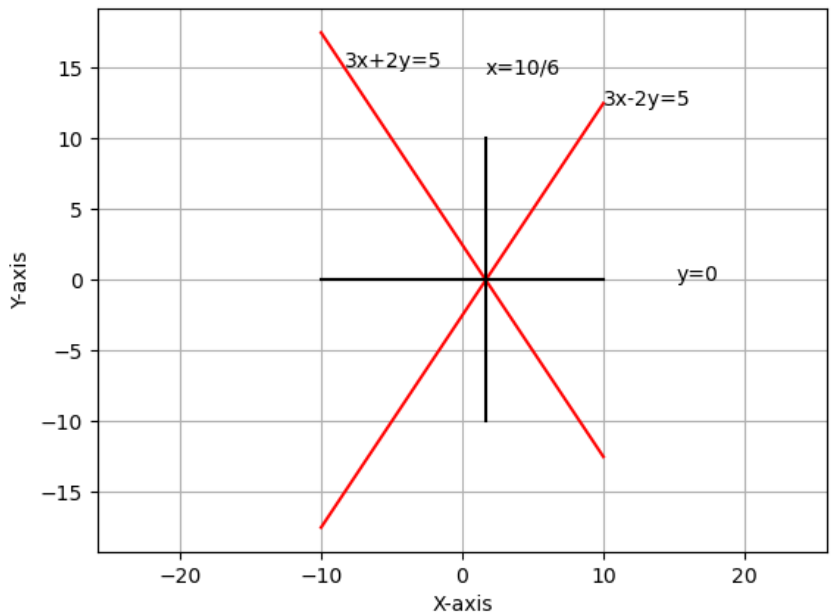

Figure: Plot of given lines and path of the moving points