Matgeo Presentation - Problem 4.13.26

ee25btech11021 - Dhanush sagar

September 29, 2025

Problem Statement

A straight the through a fixed point (2, 3) intersects the coordinate axes at distinct points P and Q. If O is the origin and the rectangle OPRQ is completed, then the locus of R is

solution

Equation of a line with normal vector $\mathbf{n}$ through $\begin{pmatrix} 2 \\ 3 \end{pmatrix}$:

$$\mathbf{n}^T \mathbf{x} = \mathbf{n}^T \begin{pmatrix} 2 \\ 3 \end{pmatrix} \tag{0.1}$$

The x-intercept is $\mathbf{P} = \begin{pmatrix} p \\ 0 \end{pmatrix}$. The y-intercept is $\mathbf{Q} = \begin{pmatrix} 0 \\ q \end{pmatrix}$.

The origin is

$$\mathbf{O} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{0.2}$$

If the intercepts are $\mathbf{P} = \begin{pmatrix} p \\ 0 \end{pmatrix}$, $\mathbf{Q} = \begin{pmatrix} 0 \\ q \end{pmatrix}$, then normal vector is

$$\mathbf{n} = \begin{pmatrix} \frac{1}{p} \\ \frac{1}{q} \end{pmatrix} \tag{0.3}$$

solution

The opposite vertex of rectangle OPRQ is then

$$\mathbf{R} = \begin{pmatrix} p \\ q \end{pmatrix} \tag{0.4}$$

Write $\mathbf{n}$ in terms of $\mathbf{R}$ by

$$\mathbf{n} = \begin{pmatrix} \frac{1}{p} \\ \frac{1}{q} \end{pmatrix} = \begin{pmatrix} \frac{1}{x} \\ \frac{1}{y} \end{pmatrix} \text{ where } \mathbf{R} = \begin{pmatrix} x \\ y \end{pmatrix} : \tag{0.5}$$

$$\begin{pmatrix} \frac{1}{x} & \frac{1}{y} \end{pmatrix} \begin{pmatrix} 2 \\ 3 \end{pmatrix} = 1 \tag{0.6}$$

Multiply out the left-hand side:

$$\frac{2}{x} + \frac{3}{y} = 1 \tag{0.7}$$

Clear denominators by multiplying both sides by xy:

$$2y + 3x = xy \tag{0.8}$$

solution

Rearrange to standard quadratic form:

$$xy - 3x - 2y = 0 \tag{0.9}$$

A conic in matrix form is

$$\mathbf{x}^T \mathbf{V} \mathbf{x} + 2\mathbf{u}^T \mathbf{x} + f = 0, \quad \mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix}. \tag{0.10}$$

Here, the matrix corresponding to the xy term is symmetric:

$$\mathbf{V} = \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} -3 \\ -2 \end{pmatrix}, \quad f = 0 \tag{0.11}$$

$$\mathbf{x}^T \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix} \mathbf{x} + 2 \begin{pmatrix} -3 \\ -2 \end{pmatrix}^T \mathbf{x} = 0 \tag{0.12}$$

C Source Code:nor line.c

```c
#include <stdio.h>
#include <stdlib.h>

// Line passing through (2,3) with slope m
void generate_line_points(double m, double* P, double* Q) {
    // X-intercept (y=0)
    double x_intercept = 2.0 - 3.0 / m;
    // Y-intercept (x=0)
    double y_intercept = 3.0 - 2.0 * m;

    P[0] = x_intercept; P[1] = 0.0;
    Q[0] = 0.0; Q[1] = y_intercept;
}
```

Python Script:solve-r.py

```
import ctypes
import numpy as np

# Load C library
lib = ctypes.CDLL("./nor_points.so")
lib.generate_line_points.argtypes = [
    ctypes.c_double,
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double)
]

def get_line_points(m):
    P = (ctypes.c_double * 2)()
    Q = (ctypes.c_double * 2)()
    lib.generate_line_points(ctypes.c_double(m), P, Q)
    return np.array([P[0], P[1]]), np.array([Q[0], Q[1]])
```

Python Script:solve-r.py

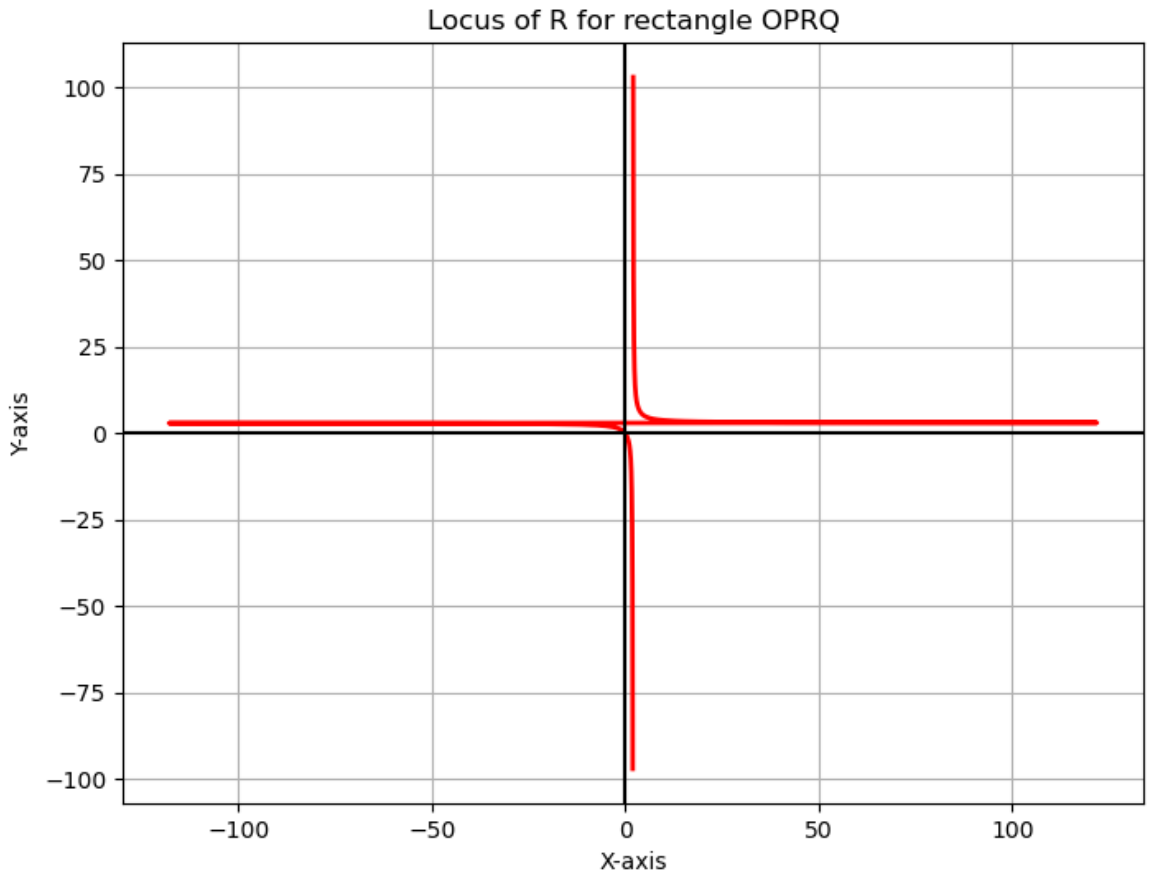
```
def compute_R_points(m_values):
    R_points = []
    for m in m_values:
        if m == 0:  # avoid horizontal line
            continue
        P, Q = get_line_points(m)
        R = P + Q
        R_points.append(R)
    return np.array(R_points)

if __name__ == "__main__":
    # Use fine slope sampling for smooth curve
    slopes = np.linspace(-50, 50, 2000)
    R_pts = compute_R_points(slopes)
    np.save("R_points.npy", R_pts)
```

Python Script: plot-r.py

```
import numpy as np
import matplotlib.pyplot as plt
# Load computed R points
R_points = np.load("R_points.npy")
# Remove any points with extremely large values (slope ~ 0)
mask = (np.abs(R_points[:,0]) < 1000) & (np.abs(R_points[:,1])
R_points = R_points[mask]
# Plot the locus of R
plt.figure(figsize=(8,6))
plt.plot(R_points[:,0], R_points[:,1], color='red', linewidth=
plt.title("Locus of R for rectangle OPRQ")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.grid(True)
plt.axhline(0, color='black')
plt.axvline(0, color='black')
plt.show()
```

Result Plot


Locus of R for rectangle OPRQ
100
75
50
25
0
−25
−50
−75
−100
Y-axis
−100
−50
0
50
100
X-axis