4.11.22

Navya Priya - EE25BTECH11045

October 11,2025

Question

Find the equations of the diagonals of the parallelogram $PQRS$ whose vertices are P $(4, 2, -6)$, Q $(5, -3, 1)$, R $(12, 4, 5)$ and S $(11, 9, -2)$.Use these equations to find the point of intersection of diagonals.

Variables Taken

P	$\begin{pmatrix} 4 \\ 2 \\ -6 \end{pmatrix}$
Q	$\begin{pmatrix} 5 \\ -3 \\ 1 \end{pmatrix}$
R	$\begin{pmatrix} 12 \\ 4 \\ 5 \end{pmatrix}$
S	$\begin{pmatrix} 11 \\ 9 \\ -2 \end{pmatrix}$

Theoretical Solution

The equation of the diagonal $\mathbf{R} - \mathbf{P}$ is

$$x = \mathbf{P} + \lambda_1 (\mathbf{R} - \mathbf{P}) \tag{1}$$

The equation of the diagonal $\mathbf{S} - \mathbf{Q}$ is

$$x = \mathbf{Q} + \lambda_2 (\mathbf{S} - \mathbf{Q}) \tag{2}$$

The centre of the parallelogram is the point of intersection of the diagonals.We solve the above two equations to find the centre using elemenetary transformations
From (1) & (2)

$$\mathbf{P} + \lambda_1 (\mathbf{R} - \mathbf{P}) = \mathbf{Q} + \lambda_2 (\mathbf{S} - \mathbf{Q}) \tag{3}$$

$$\lambda_1 (\mathbf{R} - \mathbf{P}) - \lambda_2 (\mathbf{S} - \mathbf{Q}) = \mathbf{Q} - \mathbf{P} \tag{4}$$

$$\begin{pmatrix} \mathbf{R} - \mathbf{P} & \mathbf{Q} - \mathbf{S} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \mathbf{Q} - \mathbf{P} \tag{5}$$

Theoretical Solution

$$\begin{pmatrix} 8 & -6 \\ 2 & -12 \\ 11 & 3 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -5 \\ 7 \end{pmatrix} \tag{6}$$

As there are only two variables, we consider the first two rows to compute λ_1,λ_2

$$\begin{pmatrix} 8 & -6 \\ 2 & -12 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -5 \end{pmatrix} \tag{7}$$

Forming Augmented matrix from (7),

$$\left(\begin{array}{cc|c} 8 & -6 & 1 \\ 2 & -12 & -5 \end{array}\right) \tag{8}$$

Point of Intersection

$$\begin{pmatrix} 8 & -6 & \big| & 1 \\ 2 & -12 & \big| & -5 \end{pmatrix} [R_2 \rightarrow \frac{8}{21} R_2] R_2 \rightarrow R_2 - \frac{1}{4} R_1 \begin{pmatrix} 4 & -3 & \big| & -\frac{1}{2} \\ 0 & -2 & \big| & -1 \end{pmatrix} \tag{9}$$

From (9)

$$\lambda_2 = \frac{1}{2} \tag{10}$$

Substituting the value of λ_2 in (2)

$$\mathrm{x} = \begin{pmatrix} 8 \\ 3 \\ -\frac{1}{2} \end{pmatrix} \tag{11}$$

$\therefore$ The point of intersection of the diagonals is $\begin{pmatrix} 8 \\ 3 \\ -\frac{1}{2} \end{pmatrix}$

```c
        #include <stdio.h>
// Structure for returning intersection
struct Point {
    float x;
    float y;
    float z;
};
// Function to compute intersection of diagonals
struct Point find_intersection() {
    // Coordinates of vertices
    float Px=4, Py=2, Pz=-6;
    float Qx=5, Qy=-3, Qz=1;
    float Rx=12, Ry=4, Rz=5;
    float Sx=11, Sy=9, Sz=-2;

 // Midpoints of diagonals
    struct Point O;
    O.x = (Px + Rx) / 2.0;
```

```c
    O.y = (Py + Ry) / 2.0;
    O.z = (Pz + Rz) / 2.0;
    return 0;
}
// Function to print equations of diagonals
void print_equations() {
    float Px=4, Py=2, Pz=-6;
    float Qx=5, Qy=-3, Qz=1;
    float Rx=12, Ry=4, Rz=5;
    float Sx=11, Sy=9, Sz=-2;

    printf("Equation of diagonal PR: (x,y,z) = (%.1f, %.1f, %.1f)
         + t(%.1f, %.1f, %.1f)\\n",
           Px, Py, Pz, Rx-Px, Ry-Py, Rz-Pz);

    printf("Equation of diagonal QS: (x,y,z) = (%.1f, %.1f, %.1f)
         + s(%.1f, %.1f, %.1f)\\n",
           Qx, Qy, Qz, Sx-Qx, Sy-Qy, Sz-Qz);
}
```

Call C.py

```
        import ctypes
# Load the shared library
lib = ctypes.CDLL("./parallelogram.so") # use "parallelogram.dll" on Windows
# Define struct mapping in Python
class Point(ctypes.Structure):
    fields = [("x", ctypes.c_float),
              ("y", ctypes.c_float),
              ("z", ctypes.c_float)]
# Bind function signatures
lib.find_intersection.restype = Point
lib.print_equations.restype = None
# Call C functions
lib.print_equations()
intersection = lib.find_intersection()
print(f"Intersection of diagonals: ({intersection.x}, {intersection.y}, {intersection.z})")
```

Plot.py

```
    import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Points
P = np.array([4, 2, -6])
Q = np.array([5, -3, 1])
R = np.array([12, 4, 5])
S = np.array([11, 9, -2])
O = (P + R) / 2 # intersection

# Diagonals
PR = np.vstack((P, R))
QS = np.vstack((Q, S))

fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Plot edges of parallelogram
```

Plot.py

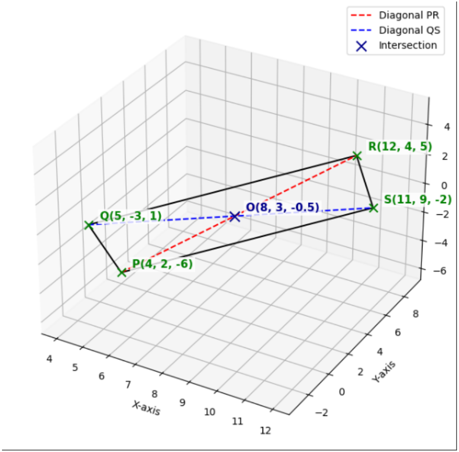
```python
edges = [(P, Q), (Q, R), (R, S), (S, P)]
for edge in edges:
    ax.plot(*zip(*edge), color='black')
   # Plot diagonals
ax.plot(*zip(*PR), color='red', linestyle='--', label='Diagonal PR')
ax.plot(*zip(*QS), color='blue', linestyle='--', label='Diagonal QS')
# Scatter points
for pt in [P, Q, R, S]:
    ax.scatter(*pt, color='green', s=70, marker='x')
ax.scatter(*O, color='darkblue', s=100, marker='x', label="Intersection")
# Format coords for labels
def clean_coords(arr):
    return tuple(int(x) if float(x).is_integer() else round(float(x),2) for x in arr)
 # Labels placed with small offset
offset = 0.3
```

Plot.py

```
labels = {
    "P": (P, (offset, offset, offset)),
    "Q": (Q, (offset, offset, offset)),
    "R": (R, (offset, offset, offset))
    "S": (S, (offset, offset, offset)),
    "O": (O, (offset, offset, offset)),}
for name, (point, delta) in labels.items():
    label_pos = point + np.array(delta)
    ax.text(*label_pos, f"{name}{clean_coords(point)}",
            fontsize=11, fontweight="bold",
            color="darkblue" if name=="O" else "green",
            bbox=dict(facecolor="white", edgecolor="none", pad=2,
                alpha=0.8))
# Axes and title
ax.set_xlabel("X-axis")
ax.set_ylabel("Y-axis")
ax.set_zlabel("Z-axis")
ax.set_title("Parallelogram PQRS with Diagonals and Intersection"
    )
```

Plot

From the graph, theoretical solution matches with the computational solution.



Parallelogram PQRS with diagonals and intersection)