MATGEO Presentation: 4.11.32

Subhodeep Chakraborty
ee25btech11055,
IIT Hyderabad.

October 1, 2025

1. Problem
2. Solution
   - Plot
3. C Code
4. Python Code
   - Using shared objects
   - Plot
   - In pure Python
   - Plot

Problem Statement

Find the equation of the line passing through $(2, -1, 2)$ and $(5, 3, 4)$ and of the plane passing through $(2, 0, 3)$, $(1, 1, 5)$ and $(3, 2, 4)$. Also, find their point of intersection. $(12, 2018)$

Given data

Given:

$$\mathbf{A} = \begin{pmatrix} 2 \\ -1 \\ 2 \end{pmatrix} \tag{3.1}$$

$$\mathbf{B} = \begin{pmatrix} 5 \\ 3 \\ 4 \end{pmatrix} \tag{3.2}$$

$$\mathbf{P} = \begin{pmatrix} 2 \\ 0 \\ 3 \end{pmatrix} \tag{3.3}$$

$$\mathbf{Q} = \begin{pmatrix} 1 \\ 1 \\ 5 \end{pmatrix} \tag{3.4}$$

$$\mathbf{R} = \begin{pmatrix} 3 \\ 2 \\ 4 \end{pmatrix} \tag{3.5}$$

Formulae

We know, for line $\mathbf{x} = \mathbf{h} + k\mathbf{m}$ and plane $\mathbf{n}^\top \mathbf{y} = 1$,

$$\mathbf{h} = \mathbf{A} \tag{3.6}$$

$$\mathbf{m} = \mathbf{B} - \mathbf{A} \tag{3.7}$$

$$\begin{pmatrix} P & Q & R \end{pmatrix}^\top \mathbf{n} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \tag{3.8}$$

Solving

Thus

$$\begin{pmatrix} P & Q & R \end{pmatrix}^{\top} \mathbf{n} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \tag{3.9}$$

$$\begin{pmatrix} 2 & 0 & 3 \\ 1 & 1 & 5 \\ 3 & 2 & 4 \end{pmatrix} \mathbf{n} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \tag{3.10}$$

Solving

On solving

$$\xleftrightarrow{R_2=2R_2-R_1;R_3=2R_3-3R_1} \left(\begin{array}{ccc|c} 2 & 0 & 3 & 1 \\ 0 & 2 & 7 & 1 \\ 0 & 4 & 2 & -1 \end{array}\right) \tag{3.11}$$

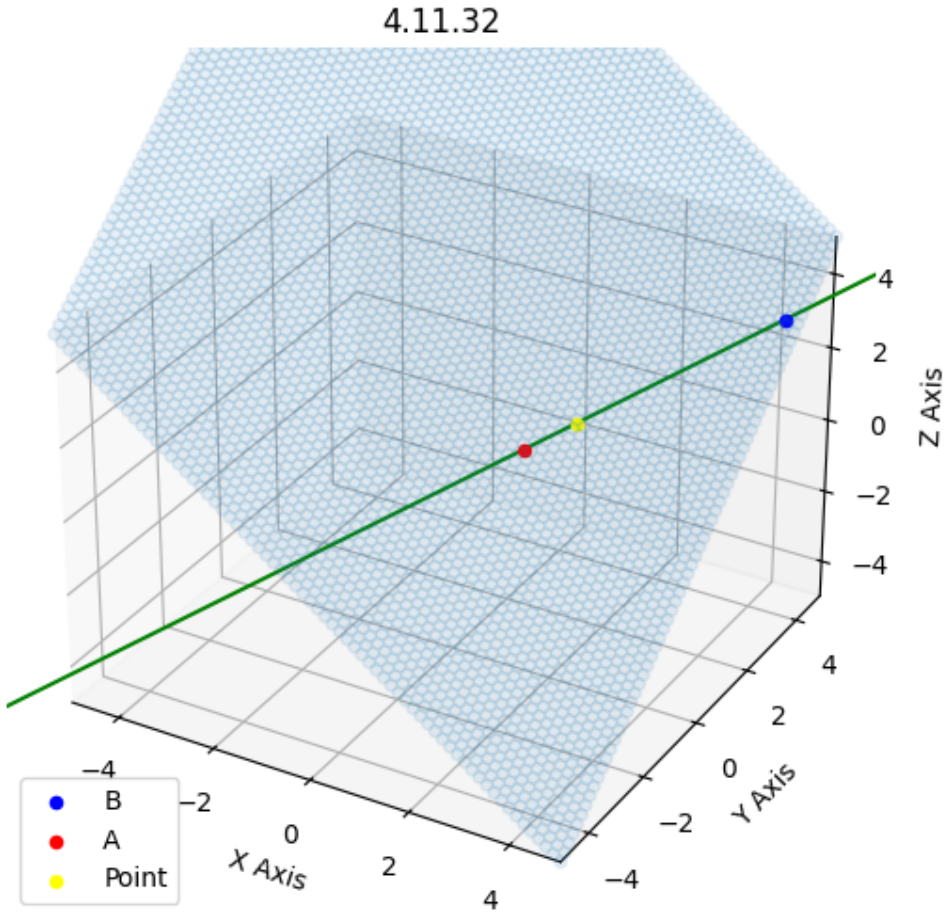
$$\xleftrightarrow{R_3=R_3-2R_2} \left(\begin{array}{ccc|c} 2 & 0 & 3 & 1 \\ 0 & 2 & 7 & 1 \\ 0 & 0 & -12 & -3 \end{array}\right) \tag{3.12}$$

$$\xleftrightarrow{R_1=4R_1+R_3;R_2=12R_2+7R_3} \left(\begin{array}{ccc|c} 8 & 0 & 0 & 1 \\ 0 & 24 & 0 & -9 \\ 0 & 0 & -12 & -3 \end{array}\right) \tag{3.13}$$

$$\xleftrightarrow{R_1=R_1/8;R_2=R_2/24;R_3=-R_3/12} \left(\begin{array}{ccc|c} 1 & 0 & 0 & 1/8 \\ 0 & 1 & 0 & -3/8 \\ 0 & 0 & 1 & 1/4 \end{array}\right) \tag{3.14}$$

$$\mathbf{n} = \frac{1}{8}\begin{pmatrix} 1 \\ -3 \\ 2 \end{pmatrix} \tag{3.15}$$

Plot



C code for generating points on line

```
void point_gen(const double* P1, const double* P2, double t, double
    * result_point) {
    result_point[0] = P1[0] + t * (P2[0] - P1[0]);
    result_point[1] = P1[1] + t * (P2[1] - P1[1]);
    result_point[2] = P1[2] + t * (P2[2] - P1[2]);
}
```

C code for generating points on plane

```
void generate_plane_points(
    // Output params
    double* x_coords, double* y_coords, double* z_coords,
    // Grid params
    double x_min, double x_max, int x_steps,
    double y_min, double y_max, int y_steps,
    // Plane stuff
    double n1, double n2, double n3, double c) {
    double x_step_val = (x_max - x_min) / (x_steps - 1);
    double y_step_val = (y_max - y_min) / (y_steps - 1);
    int index = 0;
    for (int i = 0; i < x_steps; i++) {
        for (int j = 0; j < y_steps; j++) {
            double current_x = x_min + i * x_step_val;
            double current_y = y_min + j * y_step_val;
            double current_z;
```

```
                // Vertical plane check
                if ((c < 1e-9)&&(c > -1e-9)) {
                    current_z = 0.0;
                } else {
                    current_z = (-n1 * current_x - n2 * current_y + c) /
                        n3;
                }
                x_coords[index] = current_x;
                y_coords[index] = current_y;
                z_coords[index] = current_z;
                index++;
            }
        }
}
```

Python code for plotting using C

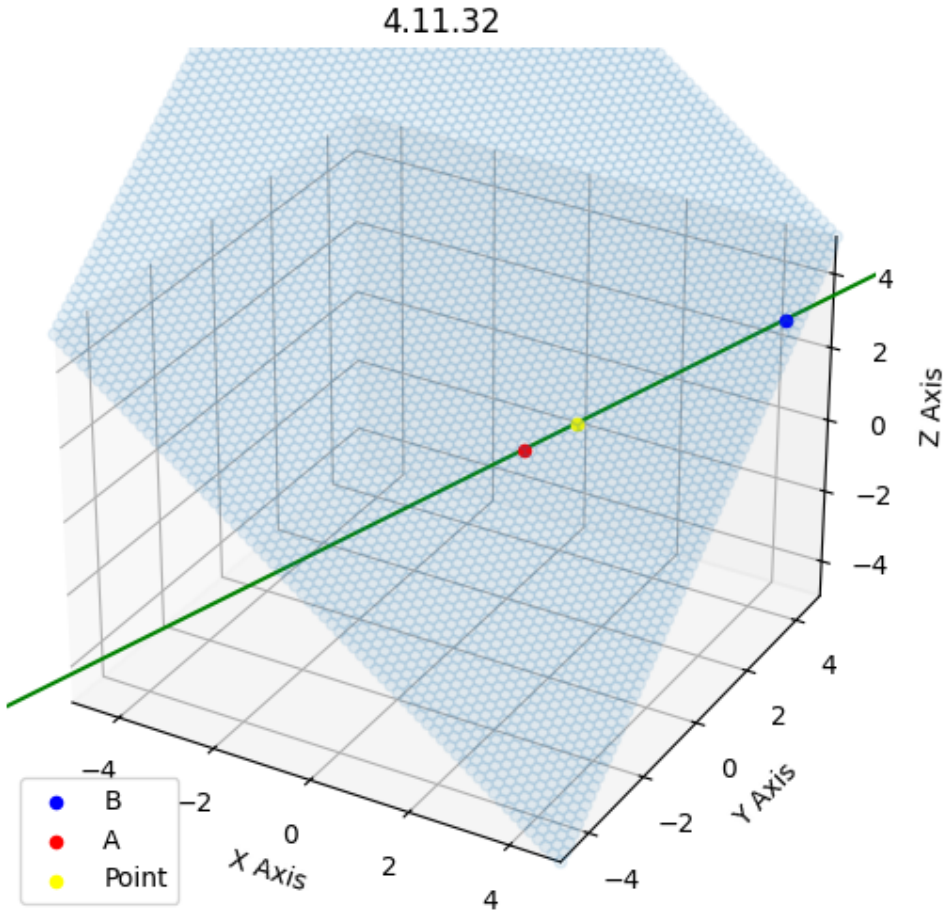
```
import ctypes
import numpy as np
import numpy.linalg as LA
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

libline = ctypes.CDLL("./line.so")

get_point = libline.point_gen
get_point.argtypes = [
    ctypes.POINTER(ctypes.c_double), # P1
    ctypes.POINTER(ctypes.c_double), # P2
    ctypes.c_double, # t
    ctypes.POINTER(ctypes.c_double), # result_point
]
get_point.restype = None
```

```
lib = ctypes.CDLL("./plane.so")

lib.generate_plane_points.argtypes = [
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.c_double,
    ctypes.c_double,
    ctypes.c_int,
    ctypes.c_double,
    ctypes.c_double,
    ctypes.c_int,
    ctypes.c_double,
    ctypes.c_double,
    ctypes.c_double,
    ctypes.c_double,
]
lib.generate_plane_points.restype = None
```

```
DoubleArray3 = ctypes.c_double * 3

a = DoubleArray3(2, -1, 2)
b = DoubleArray3(5, 3, 4)
p = DoubleArray3(2, 0, 3)
q = DoubleArray3(1, 1, 5)
r = DoubleArray3(3, 2, 4)

l1 = DoubleArray3(-28, -41, -18)
l2 = DoubleArray3(32, 39, 22)
n = DoubleArray3(0.2, -0.2, 0.2)
c = 1

fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(111, projection="3d")

t_values = np.linspace(0, 1, 100)
line_points_x, line_points_y, line_points_z = [], [], []
```

```
ax.plot(
    line_points_x,
    line_points_y,
    line_points_z,
    color="gray",
)
ax.scatter(b[0], b[1], b[2], color="blue", label="b")
ax.scatter(a[0], a[1], a[2], color="red", label="a")
ax.scatter(c[0], c[1], c[2], color="green", label="Point")

ax.set_xlabel("X Axis")
ax.set_ylabel("Y Axis")
ax.set_zlabel("Z Axis")
ax.set_title("2.9.6")
ax.legend()
ax.grid(True)
plt.savefig("../figs/plot.png")
plt.show()
```

Plot



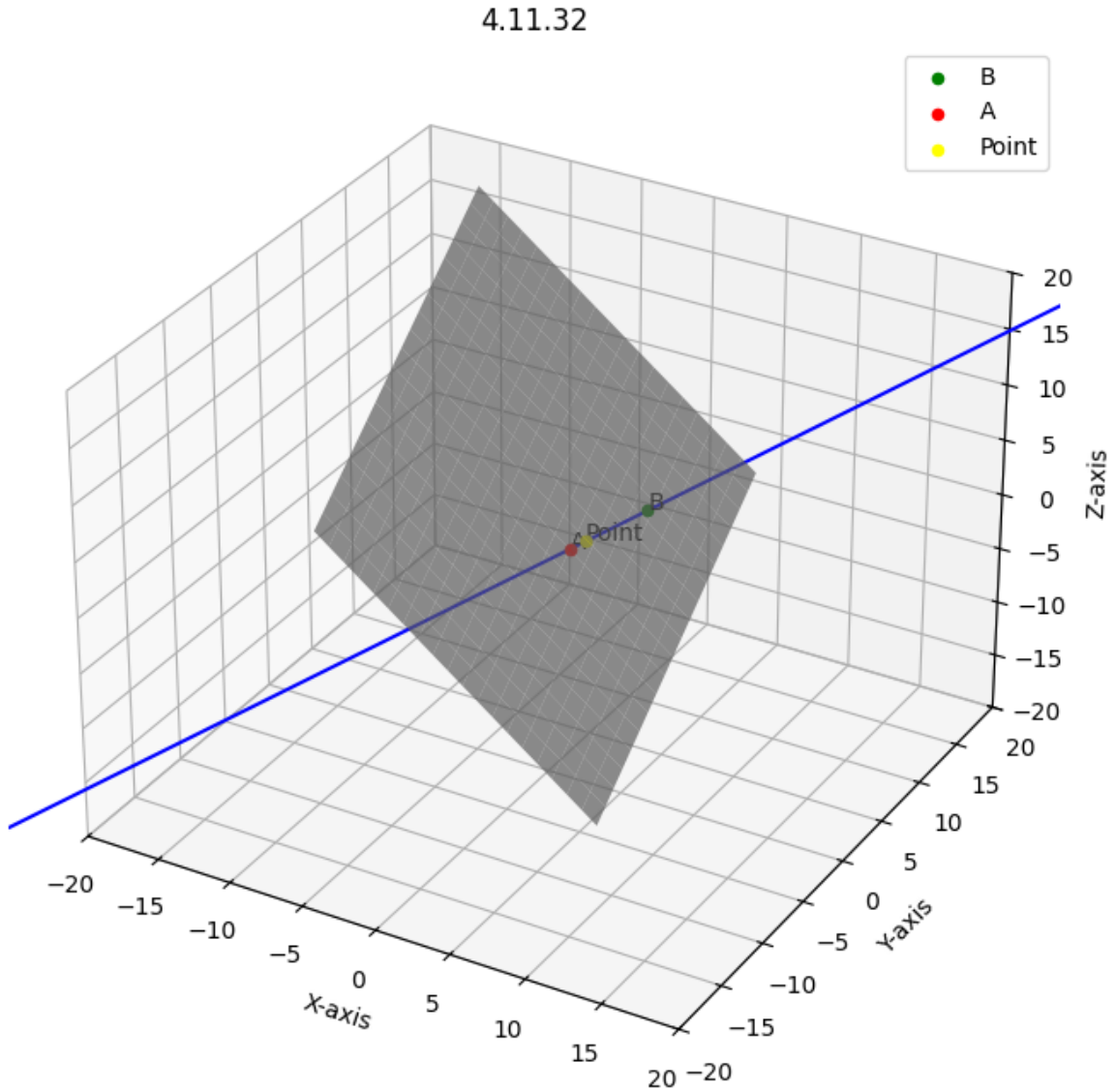
Pure Python code

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

a = np.array([5, 1, 6]).T
b = np.array([3, 4, 1]).T
c = np.array([13 / 5, 23 / 5, 0])

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

ax.plot([a[0], c[0]], [a[1], c[1]], [a[2], c[2]], color="blue", label="b")

ax.scatter(b[0], b[1], b[2], color="blue", label="b")
ax.scatter(a[0], a[1], a[2], color="red", label="a")
ax.scatter(c[0], c[1], c[2], color="green", label="Point")
```

Pure Python code

```
ax.text(a[0], a[1], a[2], "A")
ax.text(b[0], b[1], b[2], "B")
ax.text(c[0], c[1], c[2], "Point")

ax.set_xlabel("X-axis")
ax.set_ylabel("Y-axis")
ax.set_zlabel("Z-axis")
ax.set_title("2.9.6")
ax.set_xlim([-5, 5])
ax.set_ylim([-5, 5])
ax.set_zlim([-5, 5])
ax.legend()
ax.grid(True)

plt.savefig("../figs/python.png")
plt.show()
```

Plot


4.11.32
B
A
Point
B
A
Point
20
15
10
5
0
−5
−10
−15
−20
Z-axis
−20
−15
−10
−5
0
5
10
15
20
X-axis
20
15
10
5
0
−5
−10
−15
−20
Y-axis