MATGEO Presentation: 4.3.45

Subhodeep Chakraborty
ee25btech11055,
IIT Hyderabad.

October 1, 2025

Problem Statement

Find the coordinates of the point where the line through $(5, 1, 6)$ and $(3, 4, 1)$ crosses the YX-plane.

Given data

Given:

$$\mathbf{A} = \begin{pmatrix} 5 \\ 1 \\ 6 \end{pmatrix} \tag{3.1}$$

$$\mathbf{B} = \begin{pmatrix} 3 \\ 4 \\ 1 \end{pmatrix} \tag{3.2}$$

Formulae

We know:

$$\mathbf{x} = \mathbf{h} + k\mathbf{m} \tag{3.3}$$

$$= \mathbf{A} + k\left(\mathbf{B} - \mathbf{A}\right) \tag{3.4}$$

$$\mathbf{e_3}^\top \mathbf{x} = 0 \tag{3.5}$$

Solving

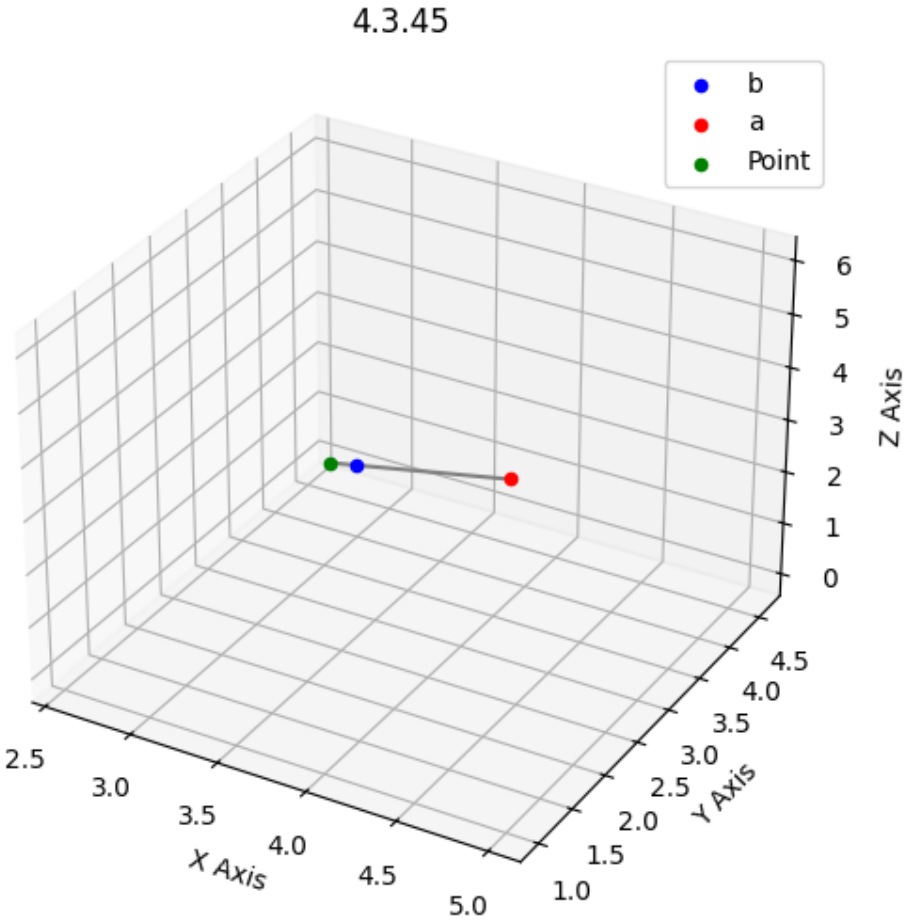
Thus

$$\mathbf{x} = \begin{pmatrix} 5 - 2k \\ 1 + 3k \\ 6 - 5k \end{pmatrix} \tag{3.6}$$

$$\mathbf{e_3}^\top \mathbf{x} = 0 \implies k = 6/5 \tag{3.7}$$

$$\mathbf{x} = \begin{pmatrix} 13/5 \\ 23/5 \\ 0 \end{pmatrix} \tag{3.8}$$

Plot


4.3.45
b
a
Point
Z Axis
6
5
4
3
2
1
0
2.5
3.0
3.5
4.0
4.5
5.0
X Axis
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
Y Axis

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]);
}
```

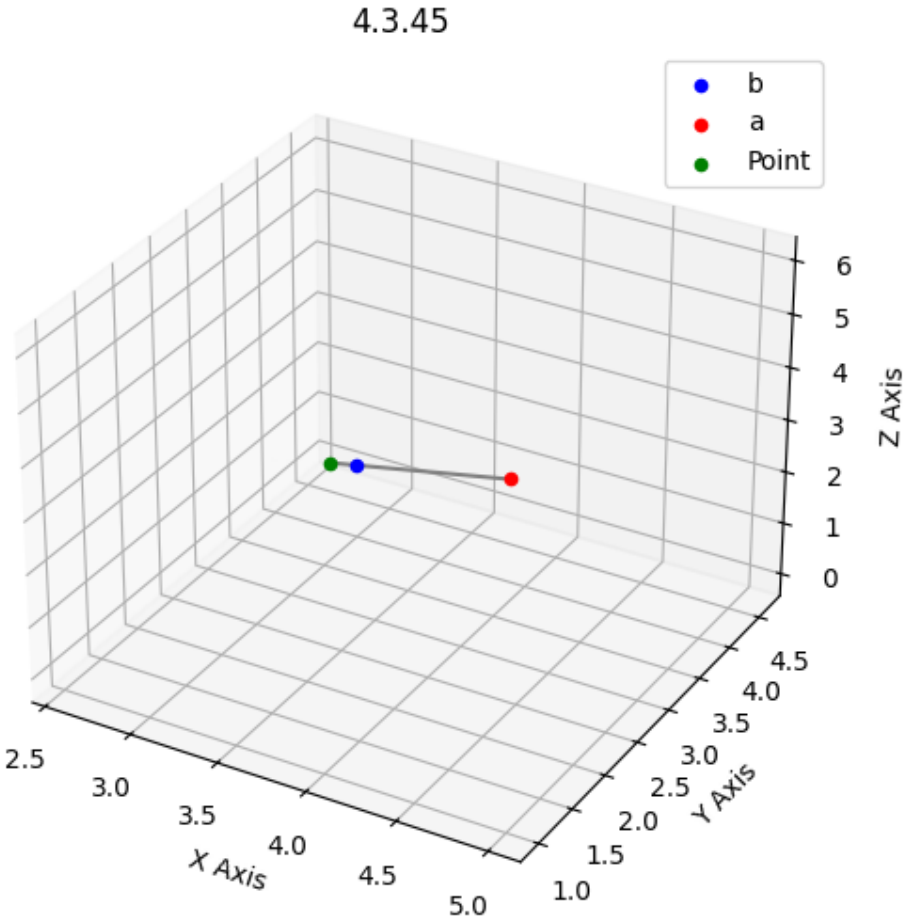
Python code for plotting using C

```
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
```

```
DoubleArray3 = ctypes.c_double * 3
a = DoubleArray3(5, 1, 6)
b = DoubleArray3(3, 4, 1)
c = DoubleArray3(13 / 5, 23 / 5, 0)

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 = [], [], []

for t in t_values:
    result_arr = DoubleArray3()

    get_point(a, c, t, result_arr)

    line_points_x.append(result_arr[0])
    line_points_y.append(result_arr[1])
    line_points_z.append(result_arr[2])
```

```
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


4.3.45
b
a
Point
Z Axis
0
1
2
3
4
5
6
Y Axis
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
X Axis
2.5
3.0
3.5
4.0
4.5
5.0

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