4.4.4

AI25BTECH11008 - Chiruvella Harshith Sharan

Question

A line passes through the point with position vector

$$\mathbf{A} = 2\hat{i} - \hat{j} + 4\hat{k}$$

and is in the direction of the vector

$$\mathbf{d} = \hat{i} + \hat{j} - 2\hat{k}.$$

Find the equation of the line?

Theoretical Solution

The given point and direction vector are:

$$\mathbf{A} = \begin{pmatrix} 2 \\ -1 \\ 4 \end{pmatrix} \tag{1}$$

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

The vector equation of a line is

$$\mathbf{r} = \mathbf{A} + \lambda \mathbf{d}, \quad \lambda \in \mathbb{R} \tag{3}$$

Substituting the values:

$$\mathbf{r} = \begin{pmatrix} 2 \\ -1 \\ 4 \end{pmatrix} + \lambda \begin{pmatrix} 1 \\ 1 \\ -2 \end{pmatrix} \tag{4}$$

Final Result

Thus, the equation of the line is

$$\mathbf{r} = \begin{pmatrix} 2+\lambda \\ -1+\lambda \\ 4-2\lambda \end{pmatrix}, \quad \lambda \in \mathbb{R} \tag{5}$$

Or in symmetric form,

$$\frac{x-2}{1} = \frac{y+1}{1} = \frac{z-4}{-2} \tag{6}$$

C Code

```
#include <stdio.h>

int main(void) {
    /* Given point A and direction d */
    double A_x = 2.0, A_y = -1.0, A_z = 4.0;
    double d_x = 1.0, d_y = 1.0, d_z = -2.0;

    /* Print the vector equation (symbolically) */
    printf(Vector equation of the line:\n);
    printf( r = A + lambda * d\n);
    printf( A = (%.1f, %.1f, %.1f)\n, A_x, A_y, A_z);
    printf( d = (%.1f, %.1f, %.1f)\n\n, d_x, d_y, d_z);

    /* Print the explicit parametric form */
    printf(Parametric form (components):\n);
    printf( x = %.1f + lambda * %.1f -> x = 2 + lambda\n, A_x,
        d_x);
    printf( y = %.1f + lambda * %.1f -> y = -1 + lambda\n, A_y,
        d_y);
```

C Code

```
/* Print the symmetric form */
printf(Symmetric form (if denominators non-zero):\n);
printf( (x - 2)/1 = (y + 1)/1 = (z - 4)/(-2)\n\n);

/* Read a lambda value from the user and compute the point on
     the line */
double lambda;
printf(Enter a value for lambda (e.g. 0, 1, -1, 2.5): );
if (scanf(%lf, &lambda) != 1) {
    fprintf(stderr, Invalid input. Exiting.\n);
    return 1;
}

double x = A_x + lambda * d_x;
double y = A_y + lambda * d_y;
double z = A_z + lambda * d_z;
```

C Code

```
printf(\nFor lambda = %.4g:\n, lambda);
printf( Point on line: (x, y, z) = (%.6g, %.6g, %.6g)\n, x, y
    , z);

/* Optionally, show a few sample lambda values */
double samples[] = {-2.0, -1.0, 0.0, 1.0, 2.0};
int n = sizeof(samples) / sizeof(samples[0]);
printf(\nSample points on the line:\n);
for (int i = 0; i < n; ++i) {
    double t = samples[i];
    double xs = A_x + t * d_x;
    double ys = A_y + t * d_y;
    double zs = A_z + t * d_z;
    printf( lambda = %5.2g -> (%.6g, %.6g, %.6g)\n, t, xs, ys
        , zs);
}
```

Python Code

```python
# plot_line_fig1.py
# Produces a 3D plot of the line through A = (2, -1, 4) with direction d = (1, 1, -2)
# Saves output as line_fig1.png

import numpy as np
import matplotlib.pyplot as plt
from pathlib import Path

# Given point and direction
A = np.array([2, -1, 4])
d = np.array([1, 1, -2])

# Parameter t for the line
t = np.linspace(-5, 5, 400)
line = A.reshape(3,1) + np.outer(d, t) # shape (3, len(t))
```

Python Code

```python

# Create 3D plot (no explicit Axes3D import required)
fig = plt.figure(figsize=(6,6))
ax = fig.add_subplot(111, projection='3d')

# Plot the line
ax.plot(line[0], line[1], line[2], linewidth=2, label='Line
    through A in direction d')

# Mark the point A
ax.scatter([A[0]], [A[1]], [A[2]], s=60, label='Point A (2,-1,4)'
    )

# Annotate point A
ax.text(A[0], A[1], A[2], ' A(2,-1,4)', fontsize=10)
```

Python Code

```python

# Axis labels
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')

# Try to set equal aspect ratio (works on matplotlib >= 3.3)
try:
    ax.set_box_aspect([1,1,1])
except Exception:
    # Fallback: approximate equal aspect by setting limits manually
    all_pts = np.hstack((line, A.reshape(3,1)))
    mins = all_pts.min(axis=1) - 1
    maxs = all_pts.max(axis=1) + 1
    ax.set_xlim(mins[0], maxs[0])
    ax.set_ylim(mins[1], maxs[1])
    ax.set_zlim(mins[2], maxs[2])

```

Plot

beamer/figs/fig1.jpg