Presentation - Matgeo

Aryansingh Sonaye
AI25BTECH11032
EE1030 - Matrix Theory

September 2, 2025

Problem Statement

Find the angle between the following pairs of lines:

(a) $\mathbf{r} = 2\hat{i} - 5\hat{j} + \hat{k} + \lambda(3\hat{i} + 2\hat{j} + 6\hat{k}), \quad \mathbf{r} = 7\hat{i} - 6\hat{j} + \mu(\hat{i} + 2\hat{j} + 2\hat{k})$

(b) $\mathbf{r} = 3\hat{i} + \hat{j} - 2\hat{k} + \lambda(\hat{i} - \hat{j} - 2\hat{k}), \quad \mathbf{r} = 2\hat{i} - \hat{j} - 5\hat{k} + \mu(3\hat{i} - 5\hat{j} - 4\hat{k})$

(c) $\dfrac{x-2}{2} = \dfrac{y-1}{5} = \dfrac{z+3}{-3}, \quad \dfrac{x+2}{-1} = \dfrac{y-4}{8} = \dfrac{z-5}{-4}$

(d) $\dfrac{x}{2} = \dfrac{y}{5} = \dfrac{z}{1}, \quad \dfrac{x-5}{4} = \dfrac{y-2}{1} = \dfrac{z-3}{8}$

Description of Variables used

Case	$\mathbf{a_1}$	$\mathbf{d_1}$	$\mathbf{a_2}$	$\mathbf{d_2}$
(a)	$\begin{pmatrix}2\\-5\\1\end{pmatrix}$	$\begin{pmatrix}3\\2\\6\end{pmatrix}$	$\begin{pmatrix}7\\-6\\0\end{pmatrix}$	$\begin{pmatrix}1\\2\\2\end{pmatrix}$
(b)	$\begin{pmatrix}3\\1\\-2\end{pmatrix}$	$\begin{pmatrix}1\\-1\\-2\end{pmatrix}$	$\begin{pmatrix}2\\-1\\-5\end{pmatrix}$	$\begin{pmatrix}3\\-5\\-4\end{pmatrix}$
(c)	$\begin{pmatrix}2\\1\\-3\end{pmatrix}$	$\begin{pmatrix}2\\5\\-3\end{pmatrix}$	$\begin{pmatrix}-2\\4\\5\end{pmatrix}$	$\begin{pmatrix}-1\\8\\-4\end{pmatrix}$
(d)	$\begin{pmatrix}0\\0\\0\end{pmatrix}$	$\begin{pmatrix}2\\5\\1\end{pmatrix}$	$\begin{pmatrix}5\\2\\3\end{pmatrix}$	$\begin{pmatrix}4\\1\\8\end{pmatrix}$

Table: Points and direction vectors for Problem 2.2.25

Theoretical Solution

General Formula:

If two lines are given in vector form, their direction vectors are denoted as

$$\mathbf{d_1}, \quad \mathbf{d_2}.$$

The angle θ between the lines is the angle between these two vectors, given by

$$\cos\theta = \frac{\mathbf{d_1}^T \mathbf{d_2}}{\|\mathbf{d_1}\| \, \|\mathbf{d_2}\|}. \tag{2.1}$$

Here:

$\mathbf{d_1}, \mathbf{d_2}$ are the direction vectors of the given lines,

$\mathbf{d_1}^T \mathbf{d_2}$ is the matrix product (dot product),

$\|\mathbf{d_i}\| = \sqrt{\mathbf{d_i}^T \mathbf{d_i}}$ is the magnitude (norm) of vector $\mathbf{d_i}$.

Theoretical Solution

(a) $\mathbf{r} = 2\hat{i} - 5\hat{j} + \hat{k} + \lambda(3\hat{i} + 2\hat{j} + 6\hat{k}), \quad \mathbf{r} = 7\hat{i} - 6\hat{j} + \mu(\hat{i} + 2\hat{j} + 2\hat{k})$

Solution:

$$\mathbf{d_1} = \begin{pmatrix} 3 \\ 2 \\ 6 \end{pmatrix}, \qquad \mathbf{d_2} = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} \tag{2.2}$$

$$\mathbf{d_1}^T \mathbf{d_2} = \begin{pmatrix} 3 & 2 & 6 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} = 19 \tag{2.3}$$

$$\|\mathbf{d_1}\| = \sqrt{\mathbf{d_1}^T \mathbf{d_1}} = \sqrt{49} = 7, \quad \|\mathbf{d_2}\| = \sqrt{\mathbf{d_2}^T \mathbf{d_2}} = \sqrt{9} = 3 \tag{2.4}$$

$$\cos\theta = \frac{19}{21} \tag{2.5}$$

Final Answer:

$$\theta = \cos^{-1}\left(\tfrac{19}{21}\right) \tag{2.6}$$

Theoretical Solution

(b) $\mathbf{r} = 3\hat{i} + \hat{j} - 2\hat{k} + \lambda(\hat{i} - \hat{j} - 2\hat{k}), \quad \mathbf{r} = 2\hat{i} - \hat{j} - 5\hat{k} + \mu(3\hat{i} - 5\hat{j} - 4\hat{k})$

Solution:

$$\mathbf{d_1} = \begin{pmatrix} 1 \\ -1 \\ -2 \end{pmatrix}, \qquad \mathbf{d_2} = \begin{pmatrix} 3 \\ -5 \\ -4 \end{pmatrix} \tag{2.7}$$

$$\mathbf{d_1}^T \mathbf{d_2} = \begin{pmatrix} 1 & -1 & -2 \end{pmatrix} \begin{pmatrix} 3 \\ -5 \\ -4 \end{pmatrix} = 16 \tag{2.8}$$

$$\|\mathbf{d_1}\| = \sqrt{\mathbf{d_1}^T \mathbf{d_1}} = \sqrt{6}, \qquad \|\mathbf{d_2}\| = \sqrt{\mathbf{d_2}^T \mathbf{d_2}} = \sqrt{50} \tag{2.9}$$

$$\cos\theta = \frac{16}{\sqrt{6} \cdot \sqrt{50}} = \frac{8}{5\sqrt{3}} \tag{2.10}$$

Theoretical Solution

Final Answer:

$$\theta = \cos^{-1}\left(\frac{8}{5\sqrt{3}}\right) \tag{2.11}$$

(c) $\dfrac{x-2}{2} = \dfrac{y-1}{5} = \dfrac{z+3}{-3}, \quad \dfrac{x+2}{-1} = \dfrac{y-4}{8} = \dfrac{z-5}{-4}$

Solution:

$$\mathbf{d_1} = \begin{pmatrix} 2 \\ 5 \\ -3 \end{pmatrix}, \qquad \mathbf{d_2} = \begin{pmatrix} -1 \\ 8 \\ -4 \end{pmatrix} \tag{2.12}$$

$$\mathbf{d_1}^T\mathbf{d_2} = \begin{pmatrix} 2 & 5 & -3 \end{pmatrix} \begin{pmatrix} -1 \\ 8 \\ -4 \end{pmatrix} = 50 \tag{2.13}$$

$$\|\mathbf{d_1}\| = \sqrt{\mathbf{d_1}^T\mathbf{d_1}} = \sqrt{38}, \qquad \|\mathbf{d_2}\| = \sqrt{\mathbf{d_2}^T\mathbf{d_2}} = 9 \tag{2.14}$$

Theoretical Solution

$$\cos\theta = \frac{50}{9\sqrt{38}} \tag{2.15}$$

Final Answer:

$$\theta = \cos^{-1}\left(\tfrac{50}{9\sqrt{38}}\right) \tag{2.16}$$

(d) $\dfrac{x}{2} = \dfrac{y}{5} = \dfrac{z}{1}, \quad \dfrac{x-5}{4} = \dfrac{y-2}{1} = \dfrac{z-3}{8}$

Theoretical Solution

Solution:

$$\mathbf{d_1} = \begin{pmatrix} 2 \\ 5 \\ 1 \end{pmatrix}, \qquad \mathbf{d_2} = \begin{pmatrix} 4 \\ 1 \\ 8 \end{pmatrix} \tag{2.17}$$

$$\mathbf{d_1}^T \mathbf{d_2} = \begin{pmatrix} 2 & 5 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \\ 8 \end{pmatrix} = 21 \tag{2.18}$$

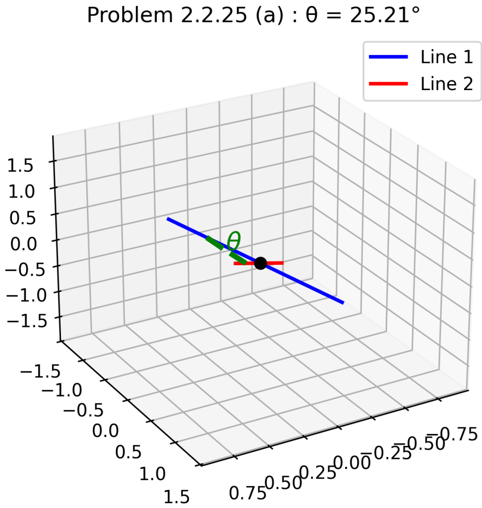
$$\|\mathbf{d_1}\| = \sqrt{\mathbf{d_1}^T \mathbf{d_1}} = \sqrt{30}, \qquad \|\mathbf{d_2}\| = \sqrt{\mathbf{d_2}^T \mathbf{d_2}} = 9 \tag{2.19}$$

$$\cos\theta = \frac{21}{9\sqrt{30}} = \frac{7}{3\sqrt{30}} \tag{2.20}$$

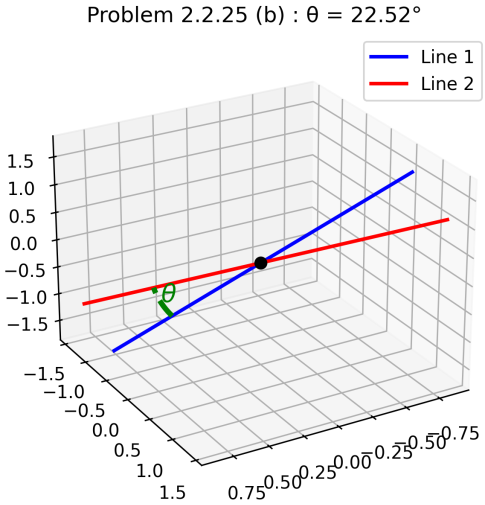
Final Answer:

$$\theta = \cos^{-1}\left(\frac{7}{3\sqrt{30}}\right) \tag{2.21}$$

Plot

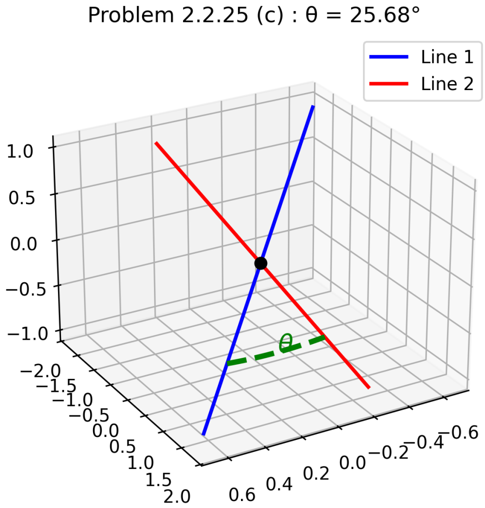

Problem 2.2.25 (a) : θ = 25.21°
Line 1
Line 2
θ
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

Plot



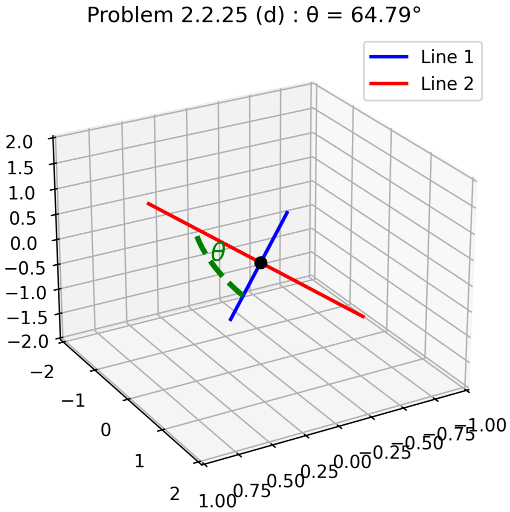
Problem 2.2.25 (b) : θ = 22.52°
Line 1
Line 2
θ
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

Plot



Problem 2.2.25 (c) : θ = 25.68°
Line 1
Line 2
1.0
0.5
0.0
−0.5
−1.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
θ

Plot


Problem 2.2.25 (d) : θ = 64.79°
Line 1
Line 2
θ
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2
−1
0
1
2
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Code - C

```c
#include <math.h>

// Function to compute angle in DEGREES between two vectors
double compute_angle_deg(double d1[3], double d2[3]) {
    // Dot product (matrix multiplication style)
    double dot = d1[0]*d2[0] + d1[1]*d2[1] + d1[2]*d2[2];

    // Norms
    double norm1 = sqrt(d1[0]*d1[0] + d1[1]*d1[1] + d1[2]*d1[2]);
    double norm2 = sqrt(d2[0]*d2[0] + d2[1]*d2[1] + d2[2]*d2[2]);

    // cos(theta)
    double cos_theta = dot / (norm1 * norm2);
```

Code - C

```
    // Clamp for numerical safety
    if (cos_theta > 1.0) cos_theta = 1.0;
    if (cos_theta < -1.0) cos_theta = -1.0;

    // Return angle in degrees
    return acos(cos_theta) * (180.0 / M_PI);
}
```

Code - Python(with shared C code)

The code to obtain the required plot is

```
import numpy as np
import matplotlib.pyplot as plt
import ctypes

# Load C shared library
lib = ctypes.CDLL("./libangle.so")
lib.compute_angle_deg.restype = ctypes.c_double

def compute_angle_deg(d1, d2):
    arr1 = (ctypes.c_double * 3)(*d1)
    arr2 = (ctypes.c_double * 3)(*d2)
    return lib.compute_angle_deg(arr1, arr2)
```

Code - Python(with shared C code)

```
# All 4 cases of Problem 2.2.25
cases = [
    (np.array([3, 2, 6]), np.array([1, 2, 2]), "a"),
    (np.array([1, -1, -2]), np.array([3, -5, -4]), "b"),
    (np.array([2, 5, -3]), np.array([-1, 8, -4]), "c"),
    (np.array([2, 5, 1]), np.array([4, 1, 8]), "d")
]

for d1, d2, label in cases:
    theta_deg = compute_angle_deg(d1, d2)
    theta = np.radians(theta_deg)

    # Normalize vectors for visualization
    d1 = d1 / np.linalg.norm(d1)
    d2 = d2 / np.linalg.norm(d2)
```

Code - Python(with shared C code)

```
# Generate line points (shorter range for clarity)
t = np.linspace(-2, 2, 100)
line1 = np.outer(t, d1)
line2 = np.outer(t, d2)

# Construct orthonormal basis in plane of d1, d2
u1 = d1
u2 = d2 - (d2 @ u1) * u1
if np.linalg.norm(u2) > 1e-8:
    u2 /= np.linalg.norm(u2)

# Arc between d1 and d2
arc_t = np.linspace(0, theta, 50)
arc_points = np.array([np.cos(a)*u1 + np.sin(a)*u2 for a in arc_t]) *
    1.2
```

Code - Python(with shared C code)

```
# Plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot(line1[:,0], line1[:,1], line1[:,2], color="blue", label="Line-1",
    lw=2)
ax.plot(line2[:,0], line2[:,1], line2[:,2], color="red", label="Line-2", lw
    =2)
ax.plot(arc_points[:,0], arc_points[:,1], arc_points[:,2],
        color="green", lw=3, ls="--")

# Label $\theta$ near the arc
mid = arc_points[len(arc_points)//2]
ax.text(mid[0], mid[1], mid[2], r"$\theta$", fontsize=14, color="
    green")

# Origin
ax.scatter(0, 0, 0, color="black", s=40)
```

Code - Python(with shared C code)

```
# Titles
ax.set_title(f"Problem 2.2.25 ({label}) : theta = {theta_deg:.2f}")
ax.legend()

# Set fixed camera view for clarity
ax.view_init(elev=25, azim=60)

# Save & show
plt.savefig(f"figure_{label}.png", dpi=300)
plt.show()
```

Code - Python only

```
import numpy as np
import matplotlib.pyplot as plt

# All 4 cases of Problem 2.2.25
cases = [
    (np.array([3, 2, 6]), np.array([1, 2, 2]), "a"),
    (np.array([1, −1, −2]), np.array([3, −5, −4]), "b"),
    (np.array([2, 5, −3]), np.array([−1, 8, −4]), "c"),
    (np.array([2, 5, 1]), np.array([4, 1, 8]), "d")
]

# Print table header
print("Problem2.2.25−Angles−between−lines")
print("
    −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−"
    )
```

Code - Python only

```
for d1, d2, label in cases:
    # Compute angle using numpy
    cos_theta = (d1 @ d2) / (np.linalg.norm(d1) * np.linalg.norm(d2))
    cos_theta = np.clip(cos_theta, -1.0, 1.0) # numerical safety
    theta = np.arccos(cos_theta)
    theta_deg = np.degrees(theta)

    # Print result in terminal
    print(f"Case ({label}): \theta = {theta_deg:.2f}")

    # Normalize for visualization
    d1 = d1 / np.linalg.norm(d1)
    d2 = d2 / np.linalg.norm(d2)
```

Code - Python only

```
# Generate shorter line segments
t = np.linspace(-2, 2, 100)
line1 = np.outer(t, d1)
line2 = np.outer(t, d2)

# Orthonormal basis for arc
u1 = d1
u2 = d2 - (d2 @ u1) * u1
if np.linalg.norm(u2) > 1e-8:
    u2 /= np.linalg.norm(u2)

# Arc between d1 and d2
arc_t = np.linspace(0, theta, 50)
arc_points = np.array([np.cos(a)*u1 + np.sin(a)*u2 for a in arc_t]) *
    1.2
```

Code - Python only

```
# Plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot(line1[:,0], line1[:,1], line1[:,2], color="blue", label="Line-1",
    lw=2)
ax.plot(line2[:,0], line2[:,1], line2[:,2], color="red", label="Line-2", lw
    =2)
ax.plot(arc_points[:,0], arc_points[:,1], arc_points[:,2],
        color="green", lw=3, ls="--")

# Label $\theta$
mid = arc_points[len(arc_points)//2]
ax.text(mid[0], mid[1], mid[2], r"$\theta$", fontsize=14, color="
    green")

# Origin
ax.scatter(0, 0, 0, color="black", s=40)
```

Code - Python only

```
# Title
ax.set_title(f"Problem 2.2.25 ({label}) : \theta = {theta_deg:.2f}")
ax.legend()

# Adjust camera view
ax.view_init(elev=25, azim=60)

# Save and show
plt.savefig(f"/sdcard/ee1030-2025/ai25btech11032/Matgeo/2.2.25/
    figs/newfigure_{label}_python.png", dpi=300)
plt.show()
```