Problem 2.4.8

ee25btech11018-Darisy Sreetej

September 12, 2025

Problem Statement

Find a unit vector perpendicular to each of the vectors $(\mathbf{a}+\mathbf{b})$ and $(\mathbf{a}+\mathbf{b})$ where, $\mathbf{a} = \hat{i} + \hat{j} + \hat{k}$ and $\mathbf{b} = \hat{i} + 2\hat{j} + 3\hat{k}$ is

Solution

Let the desired vector be $\mathbf{x}$. Then, $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$

$$\left(\mathbf{a}+\mathbf{b}\right) = \begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}\begin{pmatrix}1\\1\end{pmatrix} \tag{1.1}$$

$$\left(\mathbf{a}-\mathbf{b}\right) = \begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}\begin{pmatrix}1\\-1\end{pmatrix} \tag{1.2}$$

According to the given question ,

$$\therefore \begin{pmatrix}\mathbf{a}+\mathbf{b} & \mathbf{a}-\mathbf{b}\end{pmatrix}^T \mathbf{x} = 0 \tag{1.3}$$

(0.4) can be expressed as

$$\left\{\begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}\right\}^T \mathbf{x} = 0 \tag{1.4}$$

$$\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}^T \begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}^T \mathbf{x} = 0 \tag{1.5}$$

Solution

$$\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}^T\begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}^T\mathbf{x} = 0 \tag{1.6}$$

or,

$$\begin{pmatrix}\mathbf{a} & \mathbf{b}\end{pmatrix}^T\mathbf{x} = 0 \tag{1.7}$$

which can be expressed as

$$\begin{pmatrix}1 & 1 & 1\\ 1 & 2 & 3\end{pmatrix}\xleftrightarrow{R_2=R_2-R_1}\begin{pmatrix}1 & 1 & 1\\ 0 & 1 & 2\end{pmatrix} \tag{1.8}$$

and

$$\begin{pmatrix}1 & 1 & 1\\ 1 & 2 & 3\end{pmatrix}\xleftrightarrow{R_2=R_2-2R_1}\begin{pmatrix}1 & 1 & 1\\ -1 & 0 & 1\end{pmatrix} \tag{1.9}$$

Solution

yielding,

$$x_2 + 2x_3 = 0 \tag{1.10}$$
$$-x_1 + x_3 = 0 \tag{1.11}$$

$$\mathbf{x} = x_3 \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \tag{1.12}$$

The unit vector is

$$\mathbf{x} = \frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \tag{1.13}$$

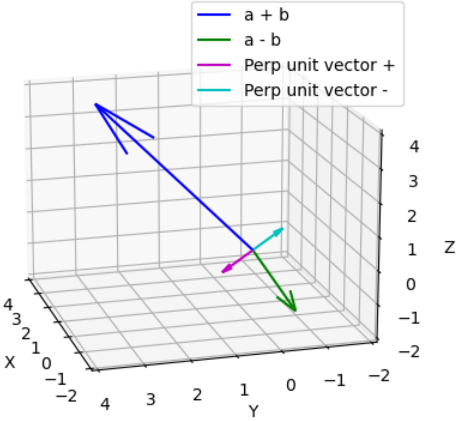
As we know that the vector can be in both the directions i.e, into and out of the plane containing **a** and **b**, so the vector perpendicular to vectors **a** and **b** would be $\pm\,(\mathbf{a} \times \mathbf{b})$.

Solution

Therefore, the desired output is

$$\mathbf{x} = \pm\frac{1}{\sqrt{6}}\begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \tag{1.14}$$

Plot-Graph



Figure

C Code for finding vectors

```c
void get_unit_vectors(double result[6]) {
    double a[3] = {1, 1, 1};
    double b[3] = {1, 2, 3};
    double sum[3], cross[3], mag;
    int i;
    for(i=0;i<3;i++) sum[i]=a[i]+b[i];
    cross[0] = sum[1]*b[2] - sum[2]*b[1];
    cross[1] = sum[2]*b[0] - sum[0]*b[2];
    cross[2] = sum[0]*b[1] - sum[1]*b[0];
    mag = sqrt(cross[0]*cross[0] + cross[1]*cross[1] + cross[2]*
        cross[2]);
    if (mag == 0.0) {
        for(i=0;i<6;i++) result[i]=-999; // error
        return;
    }
    for(i=0;i<3;i++) {
        result[i] = cross[i]/mag; // positive unit vector
        result[i+3] = -cross[i]/mag; // negative unit vector }
}
```

Calling C Function

```
import ctypes
import numpy as np

# Load the shared library
lib = ctypes.CDLL('./unitvector.so')

# Define argument and return types
lib.get_unit_vectors.argtypes = [ctypes.POINTER(ctypes.c_double)]
lib.get_unit_vectors.restype = None

# Create an array of 6 doubles to store results from C
result_arr = (ctypes.c_double * 6)()

# Call the C function
lib.get_unit_vectors(result_arr)

# Convert the result to numpy array for easy handling
results = np.array(result_arr)
```

Calling C Function

```python
if results[0] == -999:
    print("Error: vectors are parallel or no unique perpendicular vector.")
else:
    print(f"Positive unit vector: ({results[0]:.3f}, {results[1]:.3f}, {results[2]:.3f})")
    print(f"Negative unit vector: ({results[3]:.3f}, {results[4]:.3f}, {results[5]:.3f})")
```

Python Code for Plotting

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

# Given vectors
a = np.array([1, 1, 1])
b = np.array([1, 2, 3])
sum_vec = a + b
diff_vec = a - b

# Cross product to find perpendicular unit vector to both (a+b)
    and (a-b)
cross = np.cross(sum_vec, diff_vec)
mag = np.linalg.norm(cross)

if mag == 0:
    raise ValueError("Vectors are parallel; no unique
        perpendicular vector.")
```

Python Code for Plotting

```python
unit_pos = cross / mag
unit_neg = -unit_pos

# Plotting
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

origin = np.zeros(3)

# Plot vectors a+b and a-b
ax.quiver(*origin, *sum_vec, color='b', label="a + b")
ax.quiver(*origin, *diff_vec, color='g', label="a - b")

# Plot positive and negative perpendicular unit vectors
ax.quiver(*origin, *unit_pos, color='m', label="Perp unit vector +")
ax.quiver(*origin, *unit_neg, color='c', label="Perp unit vector -")
```

Python Code for Plotting

```
# Set limits and labels
ax.set_xlim([min(0, -2), max(4, 4)])
ax.set_ylim([min(0, -2), max(4, 4)])
ax.set_zlim([min(0, -2), max(4, 4)])
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')
ax.legend()

plt.title('3D plot of (a+b), (a-b) and perpendicular unit vectors
    ')
plt.show()
```