MatGeo Assignment 4.13.76

AI25BTECH11007

September 30, 2025

Question

Two motorcycles A and B are running at a speed more than the allowed speed on the road represented by the following lines:

$$\mathbf{r} = \lambda(\mathbf{i} + 2\mathbf{j} - \mathbf{k}), \qquad \mathbf{r} = (3\mathbf{i} + 3\mathbf{j}) + \mu(2\mathbf{i} + \mathbf{j} + \mathbf{k}).$$

Based on the following information, answer the questions:

1. Find the shortest distance between the given lines.
2. Find a point at which the motorcycles may collide.

Solution

Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be the points on the given lines respectively.

$$\mathbf{x}_1 = \lambda \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix}, \qquad \mathbf{x}_2 = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix} + \mu \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

(a) Shortest distance:

Let $\mathbf{A} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ and $\mathbf{B} = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix}$

Let

$$\mathbf{M} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \\ -1 & 1 \end{pmatrix}$$

$$(\mathbf{M}\ \mathbf{B} - \mathbf{A}) = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ -1 & 1 & 0 \end{pmatrix} \tag{1}$$

Row Transformation-1: $R_2 \to R_2 - 2R_1$

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -3 \\ -1 & 1 & 0 \end{pmatrix} \tag{2}$$

Row Transformation-2: $R_3 \to R_3 + R_1$

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -3 \\ 0 & 3 & 3 \end{pmatrix} \tag{3}$$

Row Transformation-3: $R_3 \to R_3 + R_2$

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -3 \\ 0 & 0 & 0 \end{pmatrix} \tag{4}$$

Therefore, rank $= 2 < 3 \Rightarrow$ The Lines intersect (not skew).

$$\boxed{\text{The Shortest Distance between the given Lines} = 0 \text{ units}} \tag{5}$$

(b) Intersection Point:

At intersection, the points on the lines are equal:

$$\mathbf{x}_1 = \mathbf{x}_2 \quad \Longrightarrow \quad \lambda \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix} + \mu \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

Equating components gives a system of equations:

$$\begin{cases} \lambda - 2\mu = 3 \\ 2\lambda - \mu = 3 \\ -\lambda - \mu = 0 \end{cases}$$

Write in augmented matrix form:

$$\left[\begin{array}{cc|c} 1 & -2 & 3 \\ 2 & -1 & 3 \\ -1 & -1 & 0 \end{array}\right]$$

Row Reduction-1 : $R_2 \to R_2 - 2R_1$

$$\left[\begin{array}{cc|c} 1 & -2 & 3 \\ 0 & 3 & -3 \\ -1 & -1 & 0 \end{array}\right]$$

Row Reduction-2 : $R_3 \to R_3 + R_1$

$$\left[\begin{array}{cc|c} 1 & -2 & 3 \\ 0 & 3 & -3 \\ 0 & -3 & 3 \end{array}\right]$$

Row Reduction-3 : $R_3 \to R_3 + R_2$

$$\left[\begin{array}{cc|c} 1 & -2 & 3 \\ 0 & 3 & -3 \\ 0 & 0 & 0 \end{array}\right]$$

From R_2: $3\mu = -3 \implies \mu = -1$
From R_1: $\lambda - 2(-1) = 3 \implies \lambda + 2 = 3 \implies \lambda = 1$

$$\therefore \lambda = 1, \quad \mu = -1$$

Intersection point:

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

$$\mathbf{r} = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix} + \mu \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix}$$

$$\boxed{\text{Intersection Point} = (1,\ 2,\ -1)}$$

Plot

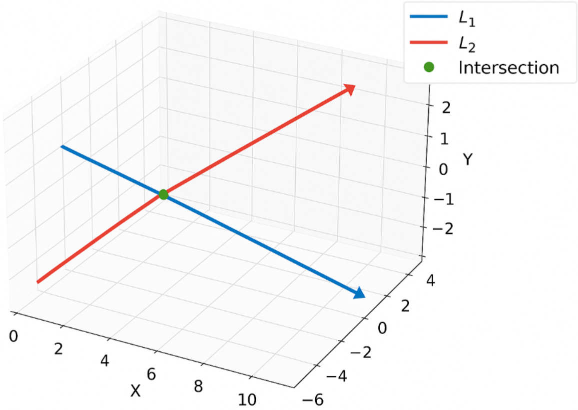

Figure: Image

C code

```
        #include <stdio.h>

int main() {
    // Augmented matrix for the system
    // lambda - 2*mu = 3
    // 2*lambda - mu = 3
    // -lambda - mu = 0
    double a[3][3] = {
        {1, -2, 3},
        {2, -1, 3},
        {-1, -1, 0}
    };

    int n = 2; // number of unknowns: lambda and mu
    double x[2]; // solution array
```

C code

```c
    // Gaussian elimination
for(int i=0; i<n; i++) {
    // Make the diagonal element 1
    double diag = a[i][i];
    for(int j=i; j<=n; j++) {
        a[i][j] /= diag;
    }

    // Eliminate the current variable from rows below
    for(int k=i+1; k<n; k++) {
        double factor = a[k][i];
        for(int j=i; j<=n; j++) {
            a[k][j] -= factor * a[i][j];
        }
    }
}
```

C code

```c
for(int i=n-1; i>=0; i--) {
    x[i] = a[i][n];
    for(int j=i+1; j<n; j++) {
        x[i] -= a[i][j] * x[j];
    }
}
double lambda = x[0];
double mu = x[1];
printf("Solution:\n");
printf("lambda = %.2lf\n", lambda);
printf("mu = %.2lf\n", mu);
double r[3];
r[0] = lambda * 1; // lambda * (1)
r[1] = lambda * 2; // lambda * (2)
r[2] = lambda * (-1); // lambda * (-1)
printf("Intersection Point: (%.2lf, %.2lf, %.2lf)\n", r[0], r
    [1], r[2]);
printf("Shortest Distance = 0 units (lines intersect)\n");
return 0;}
```

Python code

```
        import numpy as np
A = np.array([
    [1, -2, 3],
    [2, -1, 3],
    [-1, -1, 0]
], dtype=float)

# Number of unknowns
n = 2 # lambda and mu
# Perform Gaussian elimination manually
for i in range(n):
    # Make the diagonal element 1
    diag = A[i, i]
    A[i, i:] = A[i, i:] / diag
    # Eliminate the current variable from rows below
    for k in range(i+1, n):
        factor = A[k, i]
        A[k, i:] -= factor * A[i, i:]
```

Python code

```
        # Back substitution
x = np.zeros(n)
for i in range(n-1, -1, -1):
    x[i] = A[i, n] - np.dot(A[i, i+1:n], x[i+1:n])

lambda_val = x[0]
mu_val = x[1]

print(f"lambda = {lambda_val}")
print(f"mu = {mu_val}")

# Intersection point
r = np.array([lambda_val*1, lambda_val*2, lambda_val*(-1)])
print(f"Intersection Point: {r}")
print("Shortest Distance = 0 units (lines intersect)")
```