

4.2.22

EE25BTECH11019 – Darji Vivek M.

Question:

Show that the two lines

$$a_1x + b_1y + c_1 = 0, \quad a_2x + b_2y + c_2 = 0$$

where $b_1b_2 \neq 0$ are parallel if

$$\frac{a_1}{b_1} = \frac{a_2}{b_2}.$$

Solution

Form the 2×2 coefficient matrix of normals:

$$\mathbf{M} = \begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \end{pmatrix}. \quad (1)$$

Assume

$$\frac{a_1}{b_1} = \frac{a_2}{b_2}.$$

Then there exists $k \in \mathbb{R}$ such that

$$a_2 = ka_1, \quad b_2 = kb_1. \quad (2)$$

Solution

Write the rows of **M** as row vectors:

$$R_1 = (a_1 \ a_2) = (a_1 \ ka_1) = a_1 (1 \ k). \quad (3)$$

$$R_2 = (b_1 \ b_2) = (b_1 \ kb_1) = b_1 (1 \ k). \quad (4)$$

$$R_2 \mapsto R_2 - \frac{b_1}{a_1} R_1, \quad R_2 = (0, 0). \quad (5)$$

Thus the row-echelon form of \mathbf{M} has exactly one nonzero row, so

$$\text{rank}(\mathbf{M}) = 1. \quad (6)$$

if $\text{rank}(\mathbf{M}) = 1$ then the two rows (or columns) are proportional, therefore

$$\frac{a_1}{b_1} = \frac{a_2}{b_2}.$$

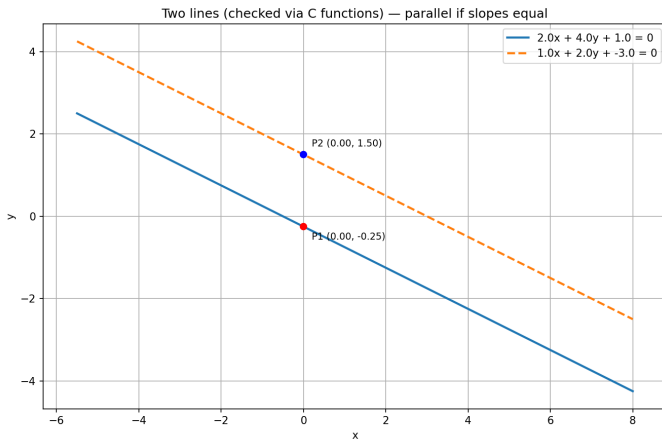


Figure: parallel lines

C Code: parallel_funcs.c

```
#include <stdio.h>
#include <math.h>

#define EPS 1e-9

// Return 1 if lines are parallel, else 0
int is_parallel(double a1, double b1, double a2, double b2) {
    double det = a1*b2 - a2*b1;
    if (fabs(det) < EPS) return 1;
    return 0;
}

// Evaluate line  $a*x + b*y + c = 0$ 
void eval_line(double a, double b, double c,
               double *x_in, double *y_out, int n) {
    for (int i = 0; i < n; ++i) {
        y_out[i] = (-a * x_in[i] - c) / b;
    }
}
```

Python: Load C Library

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

libpath = os.path.join('.', 'libparallel.so')
lib = ctypes.CDLL(libpath)

# Signatures
lib.is_parallel.argtypes = [ctypes.c_double, ctypes.c_double,
                             ctypes.c_double, ctypes.c_double]
lib.is_parallel.restype = ctypes.c_int
lib.eval_line.argtypes = [ctypes.c_double, ctypes.c_double, ctypes.
                           c_double,
                           ctypes.POINTER(ctypes.c_double),
                           ctypes.POINTER(ctypes.c_double),
                           ctypes.c_int]
lib.eval_line.restype = None
```


Python: Check Parallelism

```
def check_parallel(a1, b1, a2, b2):  
    return bool(lib.is_parallel(a1, b1, a2, b2))  
  
def eval_line(a, b, c, xs):  
    n = len(xs)  
    XTYPE = ctypes.c_double * n  
    x_arr = XTYPE(*xs)  
    y_arr = XTYPE()  
    lib.eval_line(a, b, c, x_arr, y_arr, n)  
    return np.array([y_arr[i] for i in range(n)])  
  
# Example lines  
a1, b1, c1 = 2.0, 4.0, 1.0  
a2, b2, c2 = 1.0, 2.0, -3.0  
print("Are parallel? ->", check_parallel(a1, b1, a2, b2))
```

Python: Plotting Lines

```
xs = np.linspace(-10, 10, 600)
ys1 = eval_line(a1, b1, c1, xs)
ys2 = eval_line(a2, b2, c2, xs)

plt.figure()
plt.plot(xs, ys1, label=f'{a1}x+{b1}y+{c1}=0')
plt.plot(xs, ys2, '--', label=f'{a2}x+{b2}y+{c2}=0')
plt.legend(); plt.grid(True)
plt.axis('equal')
plt.savefig('parallel_lines_ctypes.png', dpi=150)
plt.show()
```