5.2.13

System of equation

EE25BTECH11010 - Arsh Dhoke

Question

Solve the following system of linear equations 2x-2y=2 and 4x-4y=5.

Input parameters

Description	Vector
$(\mathbf{n_1})$	$\begin{pmatrix} 2 \\ -2 \end{pmatrix}$
$(\mathbf{n_2})$	$\begin{pmatrix} 4 \\ -4 \end{pmatrix}$

We can combine and write these 2 equations as

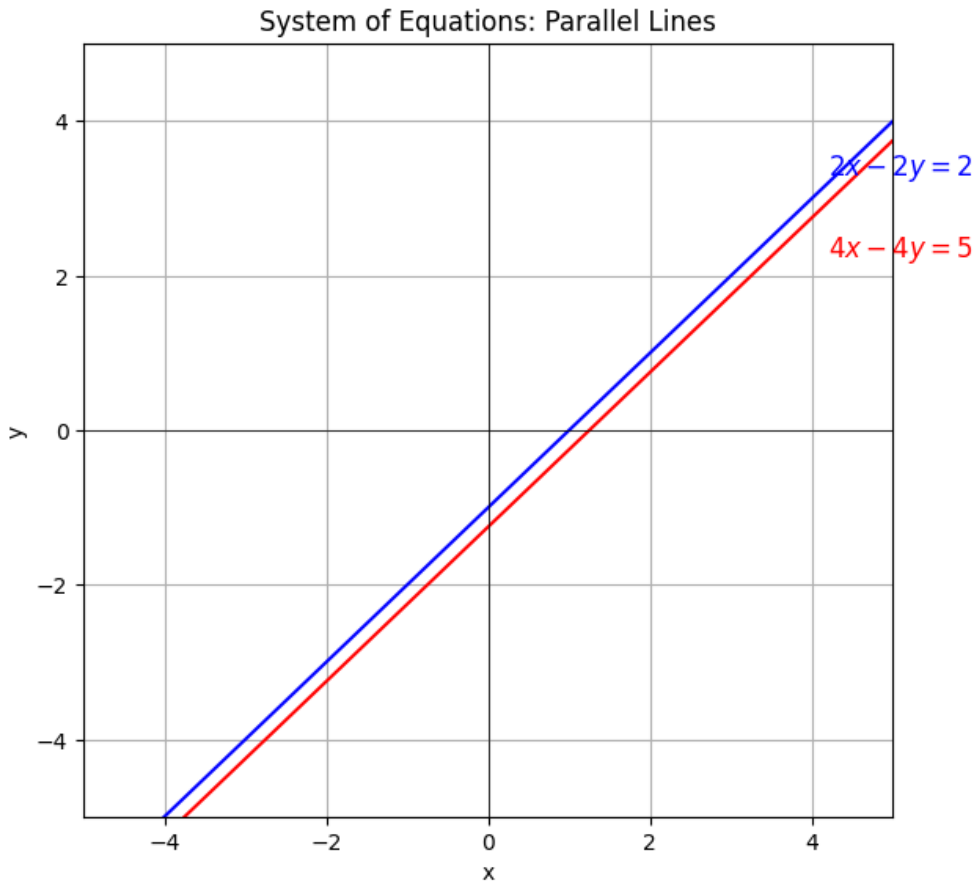
$$\begin{pmatrix} 2 & -2 \\ 4 & -4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 \\ 5 \end{pmatrix} \tag{1}$$

Gaussian Elimination

$$\left(\begin{array}{cc|c} 2 & -2 & 2 \\ 4 & -4 & 5 \end{array}\right) \xleftrightarrow{R_2 \to R_2 - 2R_1} \left(\begin{array}{cc|c} 2 & -2 & 2 \\ 0 & 0 & 1 \end{array}\right) \tag{2}$$

The second row shows 0=1 which is a contradiction.
Thus this system is inconsistent with no solution.

Graphical Representation



C Code

```c
#include<stdio.h>

double eq1(double x) {
    return x - 1.0;
}

double eq2(double x) {
    return x - 1.25;
}
```

Python Code

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

# Define x range
x = np.linspace(-5, 5, 400)

# Define equations
# Line 1: 2x - 2y = 2 -> y = x - 1
y1 = x - 1
label1 = r'$2x - 2y = 2$'

# Line 2: 4x - 4y = 5 -> y = x - 5/4
y2 = x - 1.25
label2 = r'$4x - 4y = 5$'

# --- Plot lines ---
plt.figure(figsize=(8,8))
line1, = plt.plot(x, y1, color='b')
line2, = plt.plot(x, y2, color='r')
```

Python Code

```
x_coord_1 = 4
y_coord_1 = x_coord_1 - 1
plt.annotate(label1,
             xy=(x_coord_1, y_coord_1),
             xytext=(x_coord_1 + 0.2, y_coord_1 + 0.3),

             color='b',
             fontsize=12)

x_coord_2 = 4
y_coord_2 = x_coord_2 - 1.25
plt.annotate(label2,
             xy=(x_coord_2, y_coord_2),
             xytext=(x_coord_2 + 0.2, y_coord_2 - 0.5),

             color='r',
             fontsize=12)
```

Python Code

```
# --- Standard Plot Setup ---
plt.grid(True)
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.xlim(-5, 5)
plt.ylim(-5, 5)

plt.xlabel('x')
plt.ylabel('y')
plt.title('System of Equations: Parallel Lines')
plt.savefig("/home/arsh-dhoke/ee1030-2025/ee25btech11010/matgeo
    /5.2.13/figs/q10.png")
plt.show()
```

Python+ C Code

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

# Load shared library
lib = ctypes.CDLL("./code.so")

# Define argtypes and restype for functions
lib.eq1.argtypes = [ctypes.c_double]
lib.eq1.restype = ctypes.c_double

lib.eq2.argtypes = [ctypes.c_double]
lib.eq2.restype = ctypes.c_double

# Generate x values
x_vals = np.linspace(-2, 5, 200)

# Compute y values using C functions
```

Python+ C Code

```
y1_vals = [lib.eq1(float(x)) for x in x_vals]
y2_vals = [lib.eq2(float(x)) for x in x_vals]

# Plot
plt.plot(x_vals, y1_vals, label="2x - 2y = 2")
plt.plot(x_vals, y2_vals, label="4x - 4y = 5")
plt.xlabel("x")
plt.ylabel("y")
plt.title("Graph of the two equations")
plt.legend()
plt.grid(True)
plt.savefig("/home/arsh-dhoke/ee1030-2025/ee25btech11010/matgeo
    /5.2.13/figs/q10.png")
plt.show()
```