Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 8

System of Linear Equation

Submitted by:
Chipongian, John Patrick Ryan J.

Instructor:
Engr. Dylan Josh D. Lopez

December 22, 2020

I. Objectives

This laboratory activity aims to implement the principles and techniques of the system of linear equations. It aims to familiarize with linear equations and be able to solve it using different algebra techniques.

II. Methods

In completing this laboratory activity, built-in functions of python and numpy were used. [1] Function np.array() was used in creating the matrices of the linear equations that are given for the activity. [2] Function np.linalg.solve() is used to compute the exact solution of x in a linear matrix equation.

The linear equation used in this laboratory activity is used to compute the currents of the resistor that is present in the circuit. Figure 1 shows the circuit that was used in the laboratory activity.

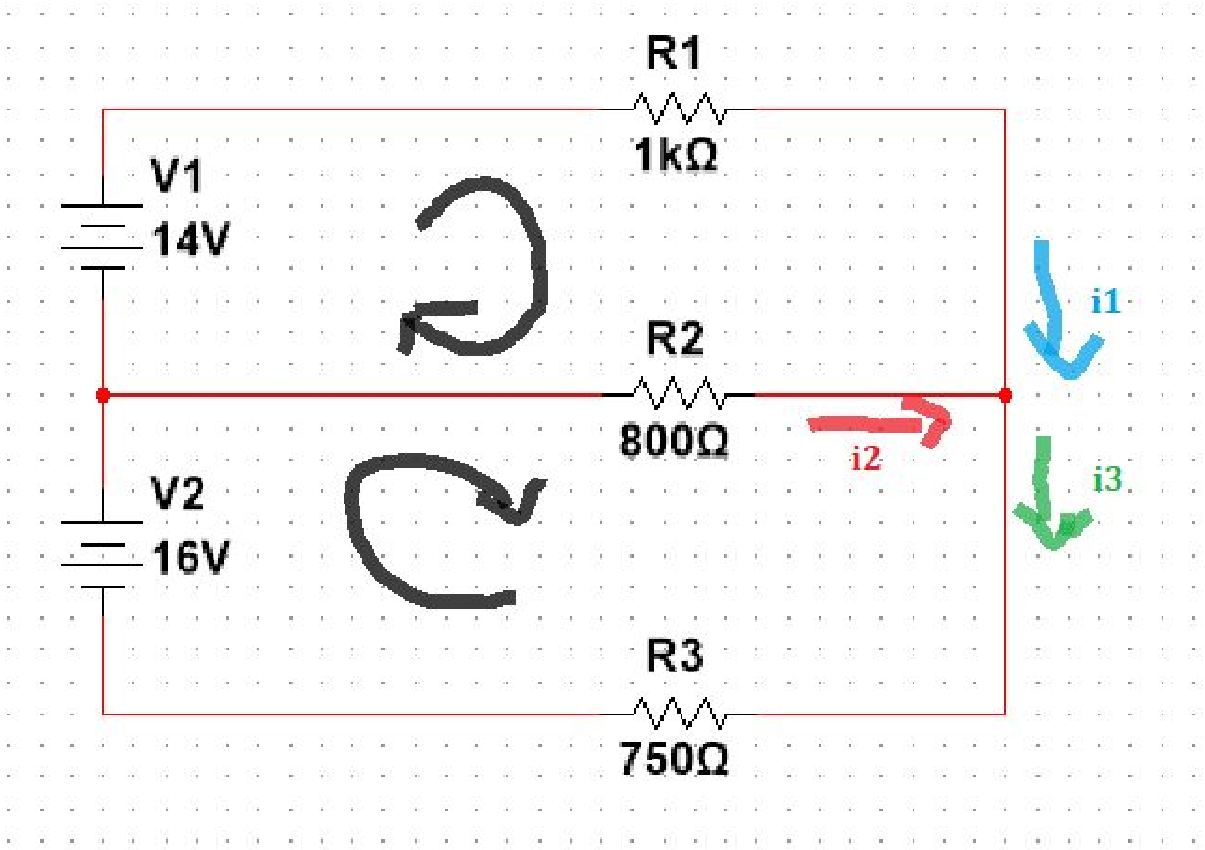

Figure 1

The linear equations that would be used in the circuit given above are the loops 1 and 2 and the summation of the junctions of the circuits. Figure 2 shows the linear equations of the circuit while the figure 3 shows the vector form of the linear equations.

$$1000I_1 - 800I_2 = 14$$
$$800I_2 + 750I_3 = 16$$
$$I_1 + I_2 - I_3 = 0$$

Figure 2

$$\begin{bmatrix} 1000 & -800 & 0 \\ 0 & 800 & 750 \\ 1 & 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 14 \\ 16 \\ 0 \end{bmatrix}$$

Figure 3

III. Results

```
LE = np.array([
    [1000,-800,0],
    [0,800,750],
    [1,1,-1]
])
V = np.array([
    [14],
    [16],
    [0]
])

currents = np.linalg.solve(LE,V)
for i in range(currents.size):
    print(f'I{i+1} = {round(float(currents[i]),4)} A')
```

Figure 4

Figure 4 shows the codes used to complete this laboratory activity. Firstly, the linear equation given is made into matrices. The LE matrix consists of the coefficients of the linear equations while the V matrix consists of dependent values of the given linear equations. The function "currents" is used to compute the currents(I) of the resistors in the circuit. [2] This is done by using np.linalg.solve() which computes for the exact solution for the "x" in the linear equation. A similar function may be used to compute the currents of the resistors, this is the function np.linalg.inv(). The difference of these two functions is while np.linalg.solve() computes for the exact value of "x", [3] the function np.linalg.inv() only solves the inverse of a matrix. After then, the loops were made to compute for all the currents(I) of the resistors. [4] It is then rounded off to four decimal numbers by using the round() function.

```
I1 = 0.016 A
I2 = 0.0026 A
I3 = 0.0186 A
```

figure 5

Using the codes earlier, the values of I_1, I_2, and I_3 were computed. The values of it are shown in figure 5.

IV. Conclusion

In completing this lab activity, I am able to use the system of linear equations to many different linear samples of linear equations. Additionally, I am able to use a system of linear equations to python codings and be able to apply it to different engineering problems.

System of linear equations is applied to robotics in its kinematics equations. [5] This is used on computing the configuration space may be a space of valid joint angles for a robot manipulator.

References

[1] W3schools.com. 2020. *Numpy Creating Arrays*. [online] Available at: <https://www.w3schools.com/python/numpy_creating_arrays.asp> [Accessed 22 December 2020].

[2] Numpy.org. 2020. *Numpy.Linalg.Solve — Numpy V1.19 Manual*. [online] Available at: <https://numpy.org/doc/stable/reference/generated/numpy.linalg.solve.html> [Accessed 22 December 2020].

[3] Numpy.org. 2020. *Numpy.Linalg.Inv — Numpy V1.21.Dev0 Manual*. [online] Available at: <https://numpy.org/devdocs/reference/generated/numpy.linalg.inv.html> [Accessed 22 December 2020].

[4] W3schools.com. 2020. *Python Round() Function*. [online] Available at: <https://www.w3schools.com/python/ref_func_round.asp> [Accessed 22 December 2020].

[5] Ratliff, N., 2020. *Linear Algebra And Robot Modeling*. [online] Ipvs.informatik.uni-stuttgart.de. Available at: <https://ipvs.informatik.uni-stuttgart.de/mlr/wp-content/uploads/2014/04/advanced_robotics_ss14_lecture1_notes1.pdf> [Accessed 22 December 2020].