TÉCNICO LISBOA

CIRCUIT THEORY AND ELECTRONICS FUNDAMENTALS

[DEEC]

DEPARTMENT OF ELECTRICAL AND COMPUTING ENGINEERING

LABORATORY 1

LINEAR CIRCUIT ANALYSIS

March 24, 2021

Group 1:

António Nunes, 95770
Francisco Branco, 95788
Pedro Alves, 95836

Professor: José Teixeira de Sousa

Contents

List of Tables

List of Figures

1 Introduction

The aim of this laboratory assignment is to study the behaviour of a circuit containing exclusively linear components. In Figure 1 the stated circuit is presented. Despite being a simple circuit, it is a perfect example since our ultimate goal is to experiment some important Circuit Theory analysing tools, presented in Section 2, so that software simulation and theoretical analysis can be stacked up against each other more easily.

In Section 2, a theoretical analysis of the circuit is presented, both by the Node Analysis (2.1) and Mesh Analysis (2.2) methods, giving us some insights on different, although equivalent forms of exploring a linear circuit. In Section 3, the circuit is analysed by computacional simulation tools, via *ngspice*, and the results are compared to the theoretical results obtained in Section 2. The conclusions of this study are outlined in Section 4.

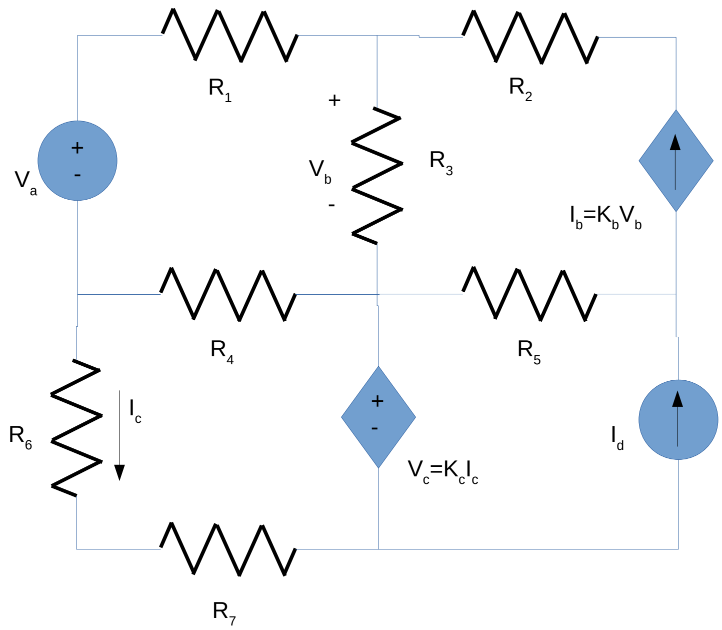

Figure 1: Linear Circuit with Resistors, Dependent and Independent Sources.

2 Theoretical Analysis

In this section, the circuit shown in Figure 1 is analysed theoretically using two methods: Nodal and Mesh Analysis.

2.1 Nodal Analysis

Before applying the method, it is important to discuss how it works. In general, it consists of discovering the voltages associated with each node of the circuit - they are the unknown variables in our equations. To obtain those, one has to apply *Kirchhoff*'s Current Law (KCL) to the nodes, always having in mind that nodes on the ends of branches containing Voltage Sources cannot be analysed in this way.

In order to obtain a fully determined system of equations, we might need to inspect for additional equations, *i.e.* equations derived from the circuit just by looking. If we apply this method to the circuit in question, we end with XX node' equations, XX supernode' equations and XX additional equations.

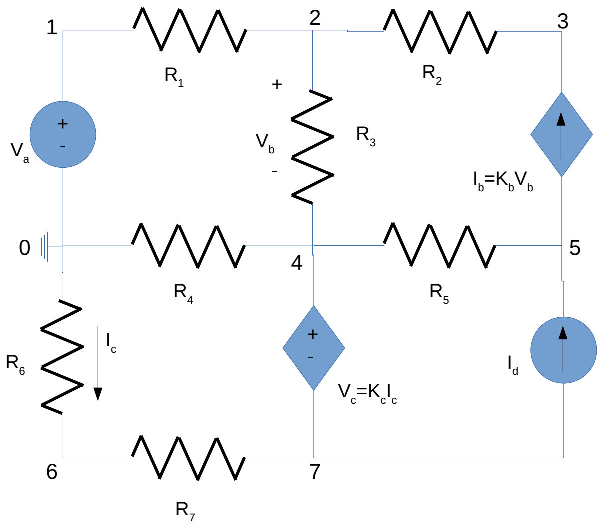

Figure 2: Node Numbering

The resultant system of equations, in the matrix and symbolic form is,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & K_cG_6 & -1 \\ G_1 & -(G_1+G_2+G_3) & G_2 & G_3 & 0 & 0 & 0 \\ 0 & K_b+G_2 & -G_2 & -K_b & 0 & 0 & 0 \\ 0 & -G_2 & G_2 & -G_5 & G_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & G_6+G_7 & G_7 \\ 0 & G_3 & 0 & -(G_3+G_4+G_5) & G_5 & G_7 & -G_7 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \end{bmatrix} = \begin{bmatrix} V_a \\ 0 \\ 0 \\ 0 \\ I_d \\ 0 \\ I_d \end{bmatrix} \quad (1)$$

Where $G_i \equiv$ conductance of the i^{th} resistor.

Since, for the Node Analysis method, the voltages used refer to the different nodes, we are free to set a reference point in whatever node we desire. Note that this has no real impact on the voltage drops on each component, since those are calculated by taking the difference of the nodal voltages on each of the component's ends. Doing so, we should clarify that the voltage associated with the relative Ground (GND), V_0, was omitted from the matrix because it leads to a trivial relation:

$$V_0 = 0 \tag{2}$$

which can be directly substituted on the remaining equations. We could do the same for V_1 but it feels unnecessary and would be making the other equations harder to decipher. That said, you can clearly see from the matrix that V_1 is directly associated with V_a:

$$V_1 = V_a \tag{3}$$

2.2 Mesh Analysis

In this second method, instead of looking for nodes, we are interested in meshes (as the method's name indicates).

Usually, we create *fictitious* currents in each mesh and we then apply *Kirchhoff*'s Voltage Law (KVL) to either the individual meshes, or to a loop. This difference occurs because sometimes, analogously to the previous method, we are unable to provide an equation for a given mesh, specially if there are Current Sources involved.

In that case, one has to look for additional equations by, for example, attributing the values of known currents to *ficticious* ones. Another option is to analyse supermeshes, *i.e.* loops, that avoid these components.

In the circuit, we have XX meshes, XX supermeshes and XX additional equations.

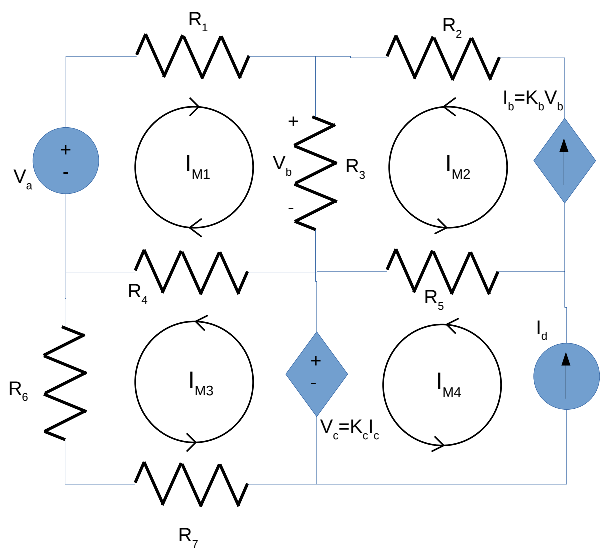

Figure 3: Mesh Numbering

Just with a glance we can realise that the equations provide a much smaller matrix, when compared to the previous one. This is expectable since there is a greater number of nodes when compared to meshes (this phenomena is detailed in Section 2.3),

$$\begin{bmatrix} R_1 + R_3 + R_4 & R_3 & R_4 \\ -K_b R_3 & 1 - K_b R_3 & 0 \\ R_4 & 0 & (R_4 + R_6 + R_7 - K_c) \end{bmatrix} \begin{bmatrix} I_{M_1} \\ I_{M_2} \\ I_{M_3} \end{bmatrix} = \begin{bmatrix} V_a \\ 0 \\ 0 \end{bmatrix} \tag{4}$$

where $I_{M_i} \equiv$ the *fictitious* current on mesh M_i.

Above, we present a $3X3$ matrix, although it is simple to observe that the circuit has 4 meshes. We obviously omitted one *ficticious current* because it is trivial to conclude by inspection that the current that flows through Mesh 4 is actually the current from the current source I_d. So, the fourth equation would be:

$$I_{M_4} = I_d \tag{5}$$

2.3 Relationship between both methods

We should note that, while the methods used clearly differ in the way they're applied, they are, obviously, intimately connected and the results given by one can be derived from the other's.

For example, given the voltage values on each node from the Node Analysis, we can easily compute the voltage drops on each of the resistors and by applying *Ohm*'s law, get the values of the current flowing through them. On the other hand, given the current values calculated with the use of the Mesh Analysis method, we can do the opposite to get the nodal voltages.

Since, for the voltage drops in each component, you need 2 values (one for each of the nodal voltages) it is easy to see that you'll quite surely end up with more equations, and thus a bigger matrix to solve, when doing a Node Analysis. However, as will be explained in greater detail in Section 2.4, this is usually the method implemented by circuit solvers like *Ngspice*.

2.4 Modified Nodal Analysis (MNA)

One can see that it is trivial to find the meshes in a plane circuit. When it comes to the nodes, there can be an increment of complexity, even though it still offers not much complexity for humans.

However, *Ngspice*, the software used to analyse the circuit in a non-manual way, (and, in general, the majority of circuit simulators) uses Modified Node Analysis - a variant of the method presented in Section 2.1.

This occurs mainly for one reason. In a topological point of view, it is easier for machines and programs to identify nodes rather than meshes. The latter requires a level of processing higher than the first, which makes it a worst choice, increasing the complexity of tasks a computer needs to execute, and so the precious time of processing.

In spite of that, the choice of node analysis might bring problems as we saw previously, regarding the existence of Voltage Sources. Hence, software adapt the nodal analysis in order to perform under any kind of linear circuit, what we call the *Modified Nodal Analysis* and by having a systematical method that works always.

It may be relevant to discuss briefly this method as it generated the results seen in 4.

In Nodal Analysis one cannot write equations for nodes connected to Voltage Sources using KCL since we don't know the current passing through these elements. However in MNA we assign an unknown current to a voltage source, hence creating new unknowns. We also relate the voltage drop between nodes and voltage source imposed. We later apply KCL to all the nodes and solve the system.

2.5 Results

In this last section we will discuss the results that the methods studied provided. The resolution of the equations on the matrix-form was done with the help of *Octave*.

The data used for solving the circuit was generated with the provided Python script, using the student number 95770, and can be found in the table below:

Name	Value [Units]
R1	1.03504497262 [$k\Omega$]
R2	2.01159104669 [$k\Omega$]
R3	3.03557466091 [$k\Omega$]
R4	4.10235086526 [$k\Omega$]
R5	3.09889833746 [$k\Omega$]
R6	2.00952426524 [$k\Omega$]
R7	1.04158528578 [$k\Omega$]
Va	5.22552047598 [$k\Omega$]
Kb	7.3172497028 [mS]
Kc	8.09359354837 [$k\Omega$]
Id	1.04791133328 [mA]

Table 1: Data used

Name	Value [V]
V1	5.2255
V2	4.9893
V3	4.5085
V4	5.0219
V5	9.0099
V6	-2.0013
V7	-3.0387

Table 2: Results using Nodal Analysis

Name	Value [mA]
I_{M_1}	0.22824
I_{M_2}	-0.239
I_{M_3}	0.99593

Table 3: Results using Mesh Analysis

For the Nodal Analysis (see subsection 2.1) it is pertinent to address that the solution is a matrix with an incredible amount of terms, so it would be almost impossible, if not painful, to insert the symbolic solution in the present document. The solutions, after substituting the values of the resistances and other constants (from table 1) are presented in table 2.

Analogously, for the Mesh Analysis (see subsection 2.2) the results are shown in table 3.

To verify the equivalence of both methods, with the use of *Ohm*'s Law, one should try to calculate the current through a circuit resistor using the voltage drop from two consecutive nodes, from the Nodal Analysis, or doing the reciprocal: to derive the nodal voltages using the current from the Mesh Analysis. Here's an example of the equivalence cited using Ohm's Law:

$$0.2362[V] = V_1 - V_2 = R_1 I_{M_1} \Longleftrightarrow I_{M_1} = \frac{V_1 - V_2}{R_1} = 0.2282[mA] \tag{6}$$

Bearing that $V_1 - V_2$ is the voltage drop between R_1.

Doing the same for the rest of the components proves, somewhat, that both analysis lead to the same results. To confirm them completely, we should compare them to the real results obtained from the simulation that we're about to explore in section 3. This comparison and verification can be found in the conclusion (section 4).

3 Simulation Analysis

3.1 Operating Point Analysis

In this section, the results given by the *Ngspice* simulation are presented in Figure 4. Table 4 shows the simulated operating point results for the circuit under analysis.

Name	**Value [A or V]**
@gb[i]	-2.38996e-04
@id[current]	1.047911e-03
@r1[i]	2.282360e-04
@r2[i]	-2.38996e-04
@r3[i]	-1.07597e-05
@r4[i]	1.224163e-03
@r5[i]	-1.28691e-03
@r6[i]	9.959274e-04
@r7[i]	9.959274e-04
v(1)	5.225520e+00
v(2)	4.989286e+00
v(3)	4.508524e+00
v(4)	5.021948e+00
v(5)	9.009942e+00
v(6)	-2.00134e+00
v(7)	-2.00134e+00
v(8)	-3.03868e+00

Table 4: Simulation of Currents and Voltages[1]

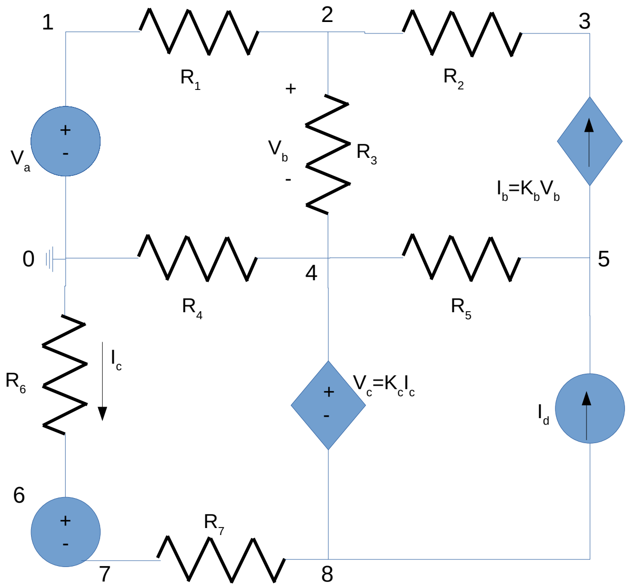

Figure 4: Circuit with Dummy Voltage Source

It should be stated that one node was added between resistors R6 and R7 in order to add a dummy 0V-voltage source in that location. This was done as a work-around the mechanics of *Ngspice* [2] to enable the voltage source V_c to be controlled by the current passing through R_6, I_c, and has no real impact on the circuit itself since that, by definition a voltage source has 0Ω resistance, and so it doesn't change the current neither the voltage drop between the nodes. This is the reason why, on Table 4, we have v(6) and v(7) outputting the same value.

[1] Variables preceded by @ denote *currents* and are expressed in Ampere; the remaining variables are *voltages* and are expressed in Volt. Gb represents the voltage controlled current source.

[2] Ngspice software only accpets controlling currents that pass in a voltage source. In our example the controling current I_c passes through a resistor.

4 Final Conclusion and General Notes

To conclude, we can realise that the results presented in Tables 2, 3 and 4 are similar, with exception of the last represented decimal places (due to rounding), which is satisfying. Despite that we must be more insightful and analyst towards the data given along this report.

So let us firstly discuss about the linearity of the circuit. Because all the components are linear, it is always expected a steady-state solution, invariable with respect to time. Also, another consequence of this is that we should expect no ”butterfly-effects”, i.e. a little nuance in the given data should not scale up to enormous disparities in the outputs.

In second it should be pointed out that although the circuit analysis is simple, the equations lead to complicated algebra, which without Octave would be very time consuming and easy to miscalculate. The best evidence of that is the *symbolic* solution matrix from both matrix equations (1), (4), which is not presented in this report due to its size and each entry complex expression, as mentioned in Section 2.5.

Finally, as already discussed in previous Sections, because the simulation software uses MNA, an equivalent method as the ones used to analyse theoretically, the results of both approaches (theoretical and practical) must be the same. As an exercise, the circuit could also be analysed using the Superposition Theorem, its *Thévenin*'s or *Norton*'s equivalent and of course, we must expect no difference in the results.