U=RIsolve: teaching & self-learning the Nodal Voltage Analysis Method the easy way

Lino Sousa
Instituto Superior de Engenharia do Porto
1140355@isep.ipp.pt

Mário Alves
Instituto Superior de Engenharia do Porto
mjf@isep.ipp.pt

Francisco Pereira
Instituto Superior de Engenharia do Porto
fdp@isep.ipp.pt

***Abstract* - This extended abstract outlines an application - dubbed *U=RIsolve* - for helping students to learn the *Nodal (or Node) Voltage Analysis Method* (NVM, for short) for electrical circuit analysis. This tool outputs the NVM equations for a given circuit previously simulated in QUCS. QUCS generates a text file (called *netlist*) which identifies all the circuit components/connections and that serves as an input to the *U=RIsolve* application. From the feedback received from a set of students/professors that volunteered to test the tool at a preliminary stage, we feel enthusiastic about the way this tool may improve self-learning of electrical circuit analysis methods.**

Keywords - Self-Learning tools, QUCS, Electrical Circuit Simulation, Nodal Voltage Analysis Method, Electrical Circuit Analysis, Kirchhoff Laws, HTML, JavaScript.

INTRODUCTION

Teaching and learning electrical circuit analysis are challenging tasks. In electrical engineering, we constantly deal with students' theoretical and practical difficulties in all analysis methods, both for DC (Direct Current) and AC (Alternated Current) circuits. This is true for the "baseline" *Branch Currents Method* (based on the *Kirchhoff Current and Voltage laws* – KCL/KVL) the *Superposition Theorem Method*, the *Mesh-Current Method* (MCM, aka *Loop-Current Method*) and the *Node-Voltage Method* (NVM).

Particularly, when applying the MCM and NVM methods, students' often fail in the selection of the right *Loops* (for the MCM) and *Nodes* (for the NVM). Even if/when they succeed in this selection, there is another very error-prone step: writing the corresponding equations (KVL based on the Loop Currents for the selected *Loops*, for the MCM; and KCL for the selected Nodes, for the NVM).

Circuit simulators have long been used for designing, testing and fine-tuning electric/electronic circuits, but also in the educational context [1][4–9]. Concerning our educational context (courses on DC/AC circuit analysis, Electrical and Computer Engineering degree, ISEP) and the landscape of simulation toolsets, we have been favoring the use of the QUCS simulator [1–2]. QUCS is easy-to-use, multiplatform, freeware/open-source and features basic functionality for teaching/learning DC/AC circuit analysis. Students are required to use QUCS for preparing their lab experiments (comparing analytical vs. experimental results) and as a self-learning tool.

Nonetheless, there is a fundamental and open problem. Circuit simulators (as QUCS) purely output "measurements", e.g. Branch Current and Node Voltage values; i.e., simulators do not show to the student how to get to those results; specifically, how to apply the analysis methodology that the student is supposed to master. With current simulators, the student is only able to compare results (analytical/experimental against simulation).

To overcome this gap, we have the *U=RIsolve* application [3], which outputs all the reasoning/equations for the NVM. It takes (as input) a "*netlist*", a text file generated by QUCS that unequivocally represents the circuit under analysis and generates all steps/equations of the NVM.

To our best knowledge, no such type of application has been developed to date. This application helps: 1) the student to learn how to apply the Node-Voltage analysis method to any DC/AC circuit; 2) the student to compare the output of the U=RIsolve application to his/her own analysis; 3) the educator to produce examples (*off-class* and *in-class*).

OUTLINE OF THE *U=RISOLVE* APPLICATION

The *U=RIsolve* application (available at [10]) has been developed in HTML and *JavaScript* and is expected to run in any web browser and over any operating system (Fig. 1).

Figure 1 - *U=RIsolve* web interface

The user must first simulate the circuit in QUCS, explicitly identifying all *Nodes* with a *'Wire Label'* (letters 'A', 'B', etc.), including the *Ground*, and then generate the *'netlist'* (a text file that univocally represents the circuit). The user then uses *U=RIsolve* to select/upload the *netlist* file ('*Selecionar ficheiro*') and trigger the circuit analysis ('*Analisar Circuito*'), generating a pop-up window with the NVM equations. The users can "refresh" and analyze other circuits, just having to simulate then in QUCS and generate the correspondent "netlist" files, without losing the ones previously simulated.

Due to size restrictions, we just outline an example of a DC circuit (Fig. 2) composed of four Nodes that have been properly labeled in QUCS: *A*, *B*, *C* and *Ground*. The currents' identification and direction represented in Fig. 2 are automatically determined by the *U=RIsolve* application. They are shown to ease the explanation .

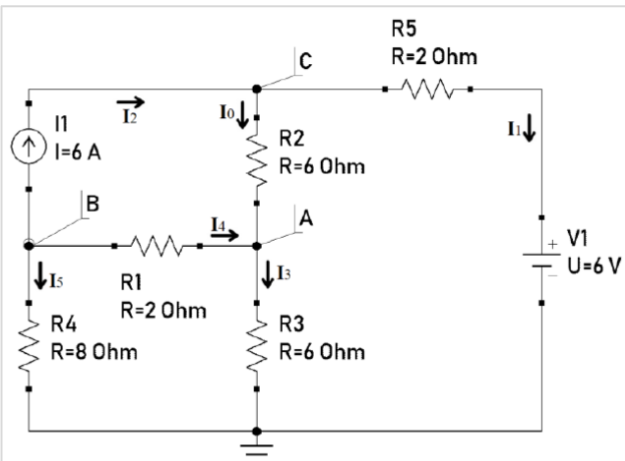

Figure 2 - Example of a DC circuit simulated in QUCS; its *netlist* file is used as an input to the *U=RIsolve* application

Taking the corresponding *netlist* (not shown here due to page limits) as an input, the *U=RIsolve* application generates a pop-up window with the NVM equations/reasoning (Fig. 3).

Equações dos nós:

- **Nó C:** $i0 + i1 = i2$
- **Nó B:** $i2 + i4 + i5 = 0$
- **Nó A:** $i3 = i0 + i4$

Equações das correntes:

- $i0 = \frac{V_C - V_A}{R_2}$ (Sentido de **C** para **A**)
- $i1 = \frac{V_C - V_1}{R_5}$ (Sentido de **C** para **gnd**)
- $i2 = I_1$ (Sentido de **B** para **C**)
- $i3 = \frac{V_A}{R_3}$ (Sentido de **A** para **gnd**)
- $i4 = \frac{V_B - V_A}{R_1}$ (Sentido de **B** para **A**)
- $i5 = \frac{V_B}{R_4}$ (Sentido de **B** para **gnd**)

Equações Finais:

- **Nó C:** $\frac{V_C - V_A}{R_2} + \frac{V_C - V_1}{R_5} = I_1$
- **Nó B:** $I_1 + \frac{V_B - V_A}{R_1} + \frac{V_B}{R_4} = 0$
- **Nó A:** $\frac{V_A}{R_3} = \frac{V_C - V_A}{R_2} + \frac{V_B - V_A}{R_1}$

Figure 3 - *U=RIsolve* output, for the example in Figure 2

As Figure 3 illustrates, the application starts by presenting the Node Equations for Nodes A, B and C (e.g. '*i0 + i1 = i2*', for Node C). Then, each Current is presented as a function of Nodes voltage (e.g. '$i0 = (V_C - V_A)/R_2$'), as well as a description to univocally correspond that Current to a specific Branch and to identify its direction (e.g. '*Sentido de C para A*'). The last part ('*Equações Finais*') aggregates the final equations into a system, which the student may solve in order to determine the Nodes voltage (V_A, V_B, V_C) and, subsequently, the Branch Currents (i0-i5).

CONCLUSIONS

We just started to introduce this tool to the FEELE course students ("DC electrical circuit fundamentals", 1st semester of 1st year of the Electrical and Computer Engineering degree at ISEP). Therefore, we do not have a tangible and massive feedback about its usefulness and usability yet. However, after the first functional version of the application has been released, we asked a specific set of teachers and students to test the application and report us suggestions and problems. That preliminary feedback allowed us to correct some bugs and refine the application [10], and also to consolidate our belief that this type of self-learning application may be very helpful for our students.

REFERENCES

1. Baser, Mustafa, "Promoting conceptual change through active learning using open source software for physics simulations", Australasian Journal of Educational Technology, v.22, n. 3, 2006, pp. 336-354.

2. Jhan, Stefan, Margraf Michael, Habchi, Vincent and Jacob, Raimund, "QUCS Technical Papers", 2005, Sourceforge.

3. Sousa, Lino, "U=RIsolve - Aplicativo de análise de circuitos elétricos simulados no QUCS", Electrical Engineering graduation dissertation, Instituto Superior de Engenharia do Porto, 2018.

4. Zavalani, Orion. (2015), "Computer-based simulation development of a design course project in electrical engineering", Computer Applications in Engineering Education, Wiley Periodicals, 2015, v.23, pp. 587-595.

5. Pudlowsky, Zenon, "Developing Computer-Aided Education in Electrical Engineering", International Journal of Electrical Engineering Education, Manchester U.P., 1994, v. 31, pp. 111-127.

6. Campbell, Ollin, Bourne, John, Mosterman, Pieter and Brodersen, Arthur, "The Effectiveness of Learning Simulations for Electronic Laboratories", Journal of Engineering Education, 2002, v.91, pp. 81-87.

7. Magana, Alejandra, Brophy, Sean and Bodner, George, "Instructors' Intended Learning Outcomes for Using Computational Simulations as Learning Tools", Journal of Engineering Education, 2012, v.101, pp. 220-243.

8. Zeashan, Khan and Abid, Muhammad, "Role of laboratory setup in project-based learning of freshmen electrical engineering in Pakistan", International Journal of Electrical Engineering Education, Sage, 2017, v.54, pp. 150-163.

9. Rubner, Geoff, "First-year undergraduate teaching of electrical and electronic engineering: innovation and inspiration", International Journal of Electrical Engineering Education, Sage, 2017, v.54, pp. 281-282.

10. U=RIsolve last version (2.5) available online: https://www.dropbox.com/s/9g1ugehefpcbazj/U%3DRIsolve%20v2.5.rar?dl=0