

FCA: Uma Ferramenta Livre para Análise de Circuitos Elétricos

Pablo Grigoletti^{1*}, Graçaliz Dimuro¹, Luciano Barboza², Renata Reiser¹, José U. de Nunes²

¹Escola de Informática – Universidade Católica de Pelotas (UCPel)
Rua Félix da Cunha, 412 – 96010-000 – Pelotas – RS – Brazil

²Escola de Arquitetura e Engenharia – Universidade Católica de Pelotas (UCPel)
Rua Félix da Cunha, 412 – 96010-000 – Pelotas – RS – Brazil
{pablogri, liz, lvb, reiser, junnunez}@ucpel.tche.br

Abstract. *FCA is a tool for the electric circuit analysis. It was conceived as free software, implemented using the Python programming language. The aim of this project is to develop free tools for reliable electric circuit analysis to be used in both teaching and research activities. This shall be obtained by means on interval techniques.*

Resumo. *FCA é uma ferramenta para a análise de circuitos elétricos que serve a concepção de software livre e foi implementada utilizando-se a linguagem de programação Python. O objetivo deste projeto é desenvolver ferramentas livres para análise confiável de circuitos elétricos em atividades de ensino e pesquisa, o que deverá ser obtido utilizando-se técnicas intervalares.*

1. Introdução

Um dos grandes problemas enfrentados pelos alunos do curso de Engenharia Elétrica é o alto custo de software utilizado para análise de circuitos elétricos. Existem poucos programas livres (baseados na filosofia de software livre) e de qualidade voltados à área de análise de circuitos elétricos. Pela mesma razão, as atividades de pesquisa que dependem destas ferramentas também ficam prejudicadas.

O *Free Circuit Analyser* (FCA) foi projetado e desenvolvido com o intuito de preencher esta lacuna, sendo este distribuído como software livre e licenciado através da GNU GPL. Adicionalmente, a segunda versão da ferramenta pretende incorporar técnicas da Matemática Intervalar para a modelagem de erros nos dados de entrada, controle dos erros oriundos da computação numérica e análise automática dos erros dos resultados obtidos, garantindo assim confiabilidade desses resultados.

Utilizou-se como base teórica para o desenvolvimento do FCA o método de análise de circuitos elétricos denominado *análise nodal* [Hilburn, Johnson e Johnson, 1994; Irwin, 2000], e, para a resolução de sistemas de equações lineares, a *decomposição LUP* [Cormen et al., 2002]. Esta ferramenta foi implementada utilizando-se a linguagem *Python* [Chun, 2001].

Atualmente, o FCA está sendo utilizado pelos alunos do 1º semestre de Engenharia Elétrica da UCPel. Este projeto recebe apoio financeiro do programa CTINFO/CNPq e FAPERGS.

* Bolsista IC/CNPq.

Este artigo está organizado como descrito a seguir. Na Seção 2, resume-se a análise nodal de circuitos elétricos, que se adota neste trabalho. A ferramenta FCA é descrita na Seção 3. Um exemplo de uma análise executada pela ferramenta é mostrado na Seção 4. A Seção 5 traz os trabalhos em andamento e a Conclusão.

Informações complementares sobre o projeto e a ferramenta FCA estão disponíveis em: <http://atlas.ucpel.tche.br/~pablogri>.

2. Análise de Circuitos Elétricos

Para se identificar os valores das tensões ou das correntes de um circuito elétrico são utilizadas as técnicas de análise de circuitos, sendo as mais conhecidas a *análise de malhas* e a *análise nodal*, ambas baseadas nas *Leis de Kirchhoff* [Hilburn, Johnson e Johnson, 1994; Irwin, 2000]. Estas técnicas geram sistemas de equações lineares de ordem n , cuja solução estima os valores das incógnitas do circuito.

Optou-se por utilizar a análise nodal por ser exigido menor esforço computacional para se identificar os nós de um circuito (que já fazem parte da entrada de dados) do que suas malhas. Ela baseia-se na lei de Kirchhoff das correntes, na qual as tensões nodais são as incógnitas a serem determinadas [Hilburn, Johnson e Johnson, 1994; Irwin, 2000].

A análise nodal, em muitos casos, torna-se mais fácil quando aplicada a circuitos que contenham fontes de tensão. A existência destas fontes acarreta o surgimento de supernós e de nós aparentes, de acordo com a Figura 1.

Um supernó é aquele em que dois nós terminais à uma fonte de tensão estão conectados em dois pontos quaisquer do circuito; e um nó aparente é um supernó com um de seus nós terminais conectado à referência (ponto 0).

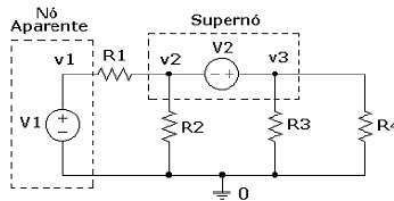


Figura 1. Circuito elétrico com supernó e nó aparente

Este método de análise de circuitos gera um sistema de equações lineares algébricas (SELA) de ordem n . A resolução deste SELA resultará nas tensões nodais do circuito.

O objetivo da *decomposição LUP* é encontrar três matrizes L , U e P , com dimensão $n \times n$, de modo que $PA=LU$, onde L é uma matriz triangular inferior unitária, U é uma matriz triangular superior e P é uma matriz de permutação (inicialmente uma matriz identidade).

Considerando um sistema do tipo $Ax=b$ e multiplicando ambos os membros da equação por uma matriz P , obtém-se $PAx=Pb$ e se $PA=LU$, então $LUx=Pb$. Considerando $Ux=y$, obtém-se:

$$Ly=Pb \quad (1)$$

$$Ux=y \quad (2)$$

Na Eq. (1), tem-se um sistema triangular inferior e obtém-se a solução para o vetor y através de uma substituição direta. Em posse do vetor y , resolve-se o sistema da Eq. (2). Este sistema é triangular superior e dele obtém-se a solução para o vetor x através de uma substituição inversa.

Na prática, observa-se que com o pivotamento, a matriz A (matriz original) será alterada em função das trocas de linhas, portanto o vetor b também deveria ser modificado para que a solução

do sistema não se altere. É para resolver este problema que existe a matriz P , para poder realizar no vetor b todas as trocas de linha que ocorreram na decomposição da matriz.

A *decomposição LUP* foi escolhida para a implementação da resolução de SELA's por ser um método numericamente estável e não possuir grande complexidade computacional [Cormen et al., 2002].

3. Descrição do FCA

O *Free Circuit Analyser* é uma ferramenta computacional multiplataforma, desenvolvida utilizando a linguagem de programação Python e baseada na filosofia de software livre.

A entrada de dados do FCA é realizada utilizando um arquivo texto contendo a definição de um circuito elétrico. Cada linha do arquivo descreve um elemento do circuito no seguinte formato:

elemento, numeração, nó1, nó2, valor

onde elemento pode ser R (resistor), I (fonte de corrente) e V (fonte de tensão); numeração é um índice que identifica cada elemento; nó1 e nó2 são os terminais entre os quais o elemento está conectado e valor é o valor numérico do elemento.

A saída de dados do FCA pode ser selecionada pelo usuário. Dentre as opções mais relevantes estão: tensões nodais, corrente nos elementos e descrição em português do circuito elétrico. Esta saída pode ser salva em um arquivo texto.

A interface gráfica foi desenvolvida utilizando-se a *wxPython*, um wrapper da biblioteca *wxWindows*, que permite que as aplicações sejam portáteis e tenham a aparência de uma aplicação nativa. A Figura 2 mostra a interface do FCA.

Para a utilização do FCA no sistema operacional MS Windows foi utilizado o módulo *py2exe* objetivando não necessitar do interpretador Python instalado na máquina.

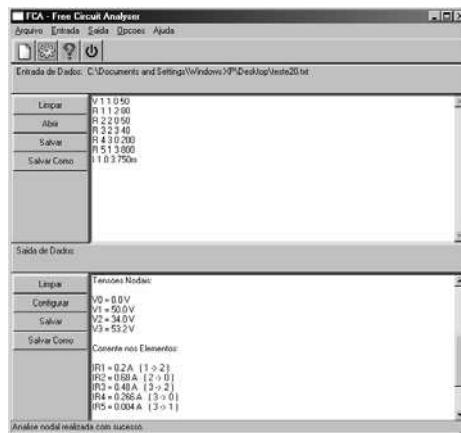


Figura 2. Interface do FCA (KDE)

4. Exemplo de Análise Realizada pelo FCA

Para ilustrar uma análise realizada pelo FCA, considere o circuito elétrico introduzido na primeira coluna da Tabela 1. A segunda coluna desta tabela mostra a representação do respectivo circuito no FCA. Os resultados obtidos com a análise nodal aplicada ao circuito, executada pela ferramenta, são as tensões nodais e a corrente nos elementos, mostrados na Tabela 2.

Tabela 1. Representação de um circuito elétrico

Circuito elétrico em sua representação gráfica	Representação no FCA
	<pre> R 1 1 4 1 R 2 1 2 2 R 3 2 0 2 R 4 3 0 1 R 5 4 5 1 R 6 5 0 1 V 1 3 2 12 V 2 3 4 6 I 1 0 1 4 I 2 4 2 2 </pre>

Tabela 2. Resultados obtidos pelo FCA

Tensões Nodais:	Corrente nos Elementos:
V0 = 0.0 V	IR1 = 0.666666666667 A (1 -> 4)
V1 = 1.166666666667 V	IR2 = 3.333333333333 A (1 -> 2)
V2 = -5.5 V	IR3 = 2.75 A (0 -> 2)
V3 = 6.5 V	IR4 = 6.5 A (3 -> 0)
V4 = 0.5 V	IR5 = 0.25 A (4 -> 5)
V5 = 0.25 V	IR6 = 0.25 A (5 -> 0)
	II1 = 4.0 A (0 -> 1)
	II2 = 2.0 A (4 -> 2)
	IV1 = 8.083333333333 A (2 -> 3)
	IV2 = 1.583333333333 A (3 -> 4)

5. Conclusão e Trabalhos em Andamento

Com a realização de vários testes comparativos com o software PSpice foi possível concluir que o FCA gera resultados em menos tempo e com maior precisão.

A utilização do software pelos alunos do curso de Engenharia Elétrica possibilita concluir que o software teve uma boa aceitação e é de fácil uso, servindo como alternativa aos softwares comerciais utilizados na área de análise de circuitos elétricos.

Atualmente existem três linhas de trabalho relativas ao projeto FCA: (i) desenvolver uma interface gráfica mais amigável para o software, semelhante as dos programas comerciais utilizados na área de análise de circuitos elétricos; (ii) permitir a utilização de fontes controladas [Hilburn, Johnson e Johnson, 1994; Irwin, 2000], pois atualmente o sistema só contempla fontes independentes e resistores; e (iii) trabalhar com a possibilidade de tratar os valores dos elementos como intervalos e aplicar as técnicas da Matemática Intervalar [Moore, 1966; 1979] no controle automático dos erros de computação numérica.

Referências Bibliográficas

- Chun W. J. Core Python Programming. s.l: Prentice Hall PTR, 2001.
- Cormen T. H. et al. Algoritmos: Teoria e Prática. 1.ed. Rio de Janeiro: Campus, 2002.
- Hilburn J. L.; Johnson D. E.; Johnson J. R. Fundamentos de Análise de Circuitos Elétricos. 4.ed. Rio de Janeiro: LTC, 1994.
- Irwin J. D. Análise de Circuitos em Engenharia. 4.ed. São Paulo: Makron Books, 2000.
- Moore, R. E. Interval Analysis. Englewood: Prentice-Hall, 1966.
- Moore, R. E. Methods and Applications of Interval Analysis. Philadelphia: SIAM, 1979.