Cálculo Numérico e Teoria dos Grafos aplicados em solução computacional de circuitos elétricos

Lucas Carvalho e Rafael Marasca Martins

***Resumo* — O presente artigo apresenta uma abordagem teórico-prática na solução de circuitos elétricos resistivo, propondo uma maneira de solucioná-los computacionalmente, utilizando métodos baseados na teoria dos grafos e cálculo numérico. Além disso, demonstra-se a dificuldade em resolver tais circuitos manualmente e conclui-se que a solução exata pode ser obtida trivialmente pela simulação.**

Palavras-Chave — Circuitos, Grafos, Métodos numéricos, Simulação

I. INTRODUÇÃO

A análise de circuitos elétricos é um tópico de interesse das engenharias, uma vez que, o mundo moderno depende da eletricidade e suas aplicações. Contudo, é inegável que realizar esta análise manualmente torna-se um processo laborioso ao passo que o número de componentes aumenta, visto que são muitas variáveis a se considerar. A fim de facilitar este processo, o presente artigo apresenta um método computacional para se encontrar a solução de circuitos elétricos resistivos, isto é, compostos apenas por fontes de tensão de corrente contínua e resistores. Além disso, é apresentada uma revisão bibliográfica dos conceitos necessários.

O artigo é dividido em três partes: na primeira, discutem-se os fundamentos matemáticos dos *sistemas lineares* e métodos de resolução. Na segunda, é apresentado o conceito de *grafos*, matematica e graficamente, junto com algumas de suas propriedades. Na última parte, introduz-se o conceito de *circuitos elétricos* e sintetiza-se tudo o que foi mostrado anteriormente na forma de um algoritmo solucionador de circuitos resistivos alimentados por fontes de tensão de corrente contínua.

II. MATRIZES

A. *Definição*

De acordo com (ANTON; RORRES, 2010, p. 26), uma *matriz* é um agrupamento retangular de números com tamanho definido por *linhas* e *colunas*, sendo estas, respectivamente, o número de fileiras horizontais e verticais. Uma matriz A com m linhas e n colunas $(m \times n)$ é representada da seguinte forma:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

Onde a_{ij} é o elemento na posição (i, j).

B. *Soma de Matrizes*

Sejam A e B matrizes com igual número de linhas e colunas na forma:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \dots & b_{mn} \end{bmatrix}$$

A soma dessas duas matrizes é dada por:

$$A + B = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \dots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \dots & a_{2n}+b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}+b_{m1} & a_{m2}+b_{m2} & \dots & a_{mn}+b_{mn} \end{bmatrix}$$

C. *Igualdade de Matrizes*

Duas matrizes são ditas iguais se, e somente se, possuírem o mesmo número de linhas e colunas e todas as suas entradas forem iguais. Ou seja, dadas A e B:

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ a_{21} & \dots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix}, B = \begin{bmatrix} b_{11} & \dots & b_{1n} \\ b_{21} & \dots & b_{2n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \dots & b_{mn} \end{bmatrix}$$

Tem-se que:

$$A = B \iff \begin{bmatrix} a_{11} = b_{11} & \dots & a_{1n} = b_{1n} \\ a_{21} = b_{21} & \dots & a_{2n} = b_{2n} \\ \vdots & \ddots & \vdots \\ a_{m1} = b_{m1} & \dots & a_{mn} = b_{mn} \end{bmatrix}$$

D. *Multiplicação de Matrizes*

Seja A uma matriz $(m \times r)$ e B uma matriz $(r \times n)$, onde M e N não são necessariamente iguais, na seguinte forma:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1r} \\ a_{21} & a_{22} & \dots & a_{2r} \\ a_{31} & a_{32} & \dots & a_{3r} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mr} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1n} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{r1} & b_{r2} & b_{r3} & \dots & b_{rn} \end{bmatrix}$$

O produto matricial entre A e B, denotado AB, será uma matriz $(m \times n)$ seguindo a regra (ANTON; RORRES, 2010, p. 30):

$$AB = C = \begin{bmatrix} c_{11} & c_{12} & c_{13} & \dots & c_{1n} \\ c_{21} & c_{22} & c_{23} & \dots & c_{2n} \\ c_{21} & c_{22} & c_{23} & \dots & c_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & c_{m3} & \dots & c_{mn} \end{bmatrix}$$

Onde

$$c_{ij} = \sum_{n=1}^{r} a_{in} b_{nj}$$

Vale ainda ressaltar que o produto matricial não é comutativo, ou seja, $AB \neq BA$.

E. *Matriz Transposta*

Se A for uma matriz $(m \times n)$, sua matriz transposta A^T será uma matriz $(n \times m)$ obtida pela troca das linhas pelas colunas de A (ANTON; RORRES, 2010, p. 34). Ou seja, se:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ a_{31} & a_{32} & \dots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

Então:

$$A^T = \begin{bmatrix} a_{11} & a_{21} & a_{31} & \dots & a_{m1} \\ a_{12} & a_{22} & a_{32} & \dots & a_{m2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & a_{3n} & \dots & a_{mn} \end{bmatrix}$$

III. SISTEMAS DE EQUAÇÕES LINEARES

A. *Definição*

Segundo (ANTON; RORRES, 2010, p. 2), uma *equação linear de n variáveis* é uma equação que pode ser escrita na forma:

$$a_1 x_1 + a_2 x_2 + \cdots + a_n x_n = b$$

Onde a_k, $k = 1, 2, \dots, n$ são constantes e x_k, $k = 1, 2, \dots, n$ são variáveis. Note que todos os termos são monômios de grau 1, ou seja, todas os termos são *lineares* e, portanto, a soma de todos os termos é linear.

Então, tem-se que uma equação linear é uma equação na qual figuram apenas constantes e variáveis lineares.

Pode-se expandir esse conceito. Se as variáveis x_k, $k = 1, 2, \dots, n$ forem organizadas em m equações lineares, cada uma com novos coeficientes a_k, $k = 1, 2, \dots, n$ e uma nova constante b, essa organização será chamada de *sistema de equações lineares*. Matematicamente:

$$\begin{cases} a_{11}x_1 + & a_{12}x_2 + \cdots + & a_{1n}x_n = & b_1 \\ a_{21}x_1 + & a_{22}x_2 + \cdots + & a_{2n}x_n = & b_2 \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1}x_1 + & a_{m2}x_2 + \cdots + & a_{mn}x_n = & b_m \end{cases}$$

Um sistema de equações lineares pode ter uma, nenhuma ou infinitas soluções (ANTON; RORRES, 2010, p. 4) e sua solução, caso exista, pode ser expressa por $(s_1, s_2, \ldots, s_n)$, onde $s_1 = x_1, s_2 = x_2, \ldots, s_n = x_n$.

B. *Forma Matricial*

É conveniente transformar o sistema anterior no produto de matrizes $Ax = b$, onde A é a matriz $(m \times n)$ dos coeficientes, b é a *matrix coluna* $(m \times 1)$ dos termos constantes e x é a *matriz coluna* das incógnitas (ANTON; RORRES, 2010, p. 33). Então, o sistema pode ser representado da seguinte forma:

$$Ax = b \iff$$
$$\begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ a_{31} & a_{32} & \ldots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_m \end{bmatrix}$$

Existem diversos métodos, diretos e iterativos, que podem ser utilizados para resolver o sistema na forma matricial. Neste artigo, são abordados os métodos de Gauss-Jordan e Gauss-Seidel.

C. *Método de Gauss-Jordan*

Dado um sistema linear determinado, é possível encontrar sua solução pelo metodo (direto) de Gauss-Jordan. Seja a matriz aumentada do sistema:

$$(A|B) = \begin{bmatrix} a_{11} & a_{21} & \ldots & a_{n1} & | & b_1 \\ a_{12} & a_{22} & \ldots & a_{n2} & | & b_2 \\ \vdots & \vdots & \ddots & \vdots & | & \vdots \\ a_{1n} & a_{2n} & \ldots & a_{nn} & | & b_n \end{bmatrix}$$

Onde a_{ij} e b_i são entradas das matrizes dos coeficientes e das constantes, respectivamente.

O método de Gauss-Jordan consiste em realizar operações elementares sobre a matriz aumentada a fim de reduzi-la à forma escalonada reduzida por linhas (ANTON; RORRES, 2010, p. 15) como segue:

$$\begin{bmatrix} 1 & 0 & \ldots & 0 & | & s_1 \\ 0 & 1 & \ldots & 0 & | & s_2 \\ \vdots & \vdots & \ddots & \vdots & | & \vdots \\ 0 & 0 & \ldots & 1 & | & s_n \end{bmatrix}$$

As operações elementares permitidas são:

1. multiplicação da linha por escalar;
2. troca de linhas;
3. adição de uma linha com um múltilo de outra.

A fase direta (ou eliminação gaussiana) é aquela em que os números abaixo do elemento da diagonal principal são zerados, enquanto na fase inversa são zerados os elementos acima (ANTON; RORRES, 2010, p.15).

De modo geral, a linha i é multiplicada pelo recíproco do elemento na diagonal principal a_{ii}, que transforma o elemento na diagonal principal (pivô) em 1. Então, as linhas L_j abaixo e acima de i serão somadas com $-a_{ji} \cdot L_i$, sendo L_i a linha de partida, o que zera todos os elementos acima e abaixo do pivô, já que o este passou a ser unitário.

Um dos problemas do passo a passo descrito é o caso do pivô ser nulo, o que resulta em uma divisão por zero.

Para garantir a generalidade e tornar o algoritmo computacionalmente viável, é preciso adicionar um passo extra: antes de efetuar as divisões, buscar na coluna i o maior elemento em valor absoluto e, se necessário, trocar sua linha com L_i. Esse passo é chamado de Pivoteamento Parcial e é descrito por (RUGGIERO; LOPES, 1988, p. 127).

O seguinte algoritmo descreve uma implementação computacional do método, assumindo um sistema determinado de n equações e n variáveis:

Algoritmo 1 Gauss-Jordan

função PIVOPARCIAL(A, b, i)
  maior $\leftarrow Aii$
  indice $\leftarrow i$
  para j $\leftarrow$ i até N **faça**
    se |maior| $< |A_{ji}|$ **então**
      maior $\leftarrow A_{ji}$
      indice $\leftarrow j$
    fim se
  fim para
  se indice $\neq$ i **então**
    troque A_i por A_{indice}
    troque b_i por b_{indice}
  fim se

fim função
função GAUSSJORDAN(A, b)
  para i $\leftarrow$ 1 até N **faça**
    pivoParcial(A, b, i)
    div $\leftarrow A_{ii}$
    para j $\leftarrow i$ até N **faça**
      $A_{ij} \leftarrow A_{ij} \cdot \frac{1}{div}$
    fim para
    $b_i \leftarrow b_i \cdot \frac{1}{div}$
    para k $\leftarrow$ 1 até N **faça**
      se $k = i$ **então**
        continue
      fim se
      $mult \leftarrow A_{ki}$
      para l $\leftarrow i$ até N **faça**
        $A_{kl} \leftarrow A_{kl} - mult \cdot A_{il}$
      fim para
      $b_k \leftarrow b_k - mult \cdot b_i$
    fim para
  fim para
fim função

D. *Método de Gauss-Seidel*

Muitas vezes, a utilização de métodos diretos leva a erros significativos de arredondamento (PIRES, 2015, p. 173), além de serem computacionalmente custosos. Uma possível solução para isso é o emprego de *métodos iterativos*, que consistem em gerar uma nova aproximação à solução real a cada *passo iterativo* com base em uma *função de iteração*.

Então, dado um sistema de equações lineares qualquer $Ax = b$, é necessário transformá-lo em um sistema do tipo $x = Cx + g$ para resolvê-lo iterativamente (RUGGIERO; LOPES, 1988, p. 167). Feita a transformação, toma-se o x do lado esquerdo da equação como o vetor solução aproximada no passo $k + 1$ e o x do lado direito como o vetor solução aproximada no passo k. Ou seja:

$$x^{(k+1)} = Cx^{(k)} + g$$

Então, dado um "chute" inicial $x^{(0)}$, define-se $x^{(k+1)}$ como a função de iteração $\varphi(x)$ do sistema:

$$\varphi(x) = Cx^{(k)} + g$$

Com $k = 0, 1, 2, \ldots, n; n \in \mathbb{N}$.

Existem diversos meios de encontrar as matrizes C e g. Um deles é o método de Gauss-Seidel, que consiste em separar a matriz A em três matrizes: L (triangular inferior), D (diagonal) e R (triangular inferior), ou seja:

$$A = L + D + R \iff$$

$$\begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ a_{21} & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & 0 \end{bmatrix}$$

$$+ \begin{bmatrix} a_{11} & 0 & \ldots & 0 \\ 0 & a_{22} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & a_{nn} \end{bmatrix} + \begin{bmatrix} 0 & a_{12} & \ldots & a_{1n} \\ 0 & 0 & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{bmatrix}$$

Então, (RUGGIERO; LOPES, 1988, p. 164) dá a seguinte demonstração:

$$Ax = b \iff (L + D + R)x = b$$

Seria conveniente ainda fatorar a matriz D da expressão $A = (L + D + R)$. Para tal, é necessário dividir cada linha pelo elemento na diagonal principal, o que resulta nas seguintes matrizes:

$$D_1 = I = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{bmatrix}$$

e

$$R_1 = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ \frac{a_{21}}{a_{22}} & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n1}}{a_{nn}} & \frac{a_{n2}}{a_{nn}} & \ldots & 0 \end{bmatrix}$$

e

$$L_1 = \begin{bmatrix} 0 & \frac{a_{12}}{a_{11}} & \ldots & \frac{a_{1n}}{a_{11}} \\ 0 & 0 & \ldots & \frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{bmatrix}$$

Logo, $Ax = b$ pode ser reescrito como:

$$Ax = b \iff D(L_1 + I + R_1)x = b$$
$$\iff (L_1 + I + R_1)x = D^{-1}b$$
$$\iff L_1x + Ix + R_1x = D^{-1}b$$
$$\iff Ix = -L_1x - R_1xD^{-1}b$$
$$\iff x = -L_1x - R_1x + D^{-1}b$$

Portanto, o método de Gauss-Seidel é dado por:

$$x^{(k+1)} = \varphi(x) = -L_1x^{(k+1)} - R_1x^{(k)} + D^{-1}b$$

Ou ainda:

$$x^{(k+1)} + L_1x^{(k+1)} = -R_1x^{(k)} + D^{-1}b$$

$$\iff (L_1 + I)x^{(k+1)} = -R_1x^{(k)} + D^{-1}b$$

$$\iff x^{(k+1)} = -(L_1+I)^{-1}R_1x^{(k)}+(L_1+I)^{-1}D^{-1}b$$

Se $C = -(L_1 + I)^{-1}R_1$ e $G = (L_1 + I)^{-1}D^{-1}b$, fica demonstrado que o método de Gauss-Seidel obedece a hipótese $x^{(k+1)} = \varphi(x) = Cx^{(k)} + g$.

Para o cálculo computacional do vetor solução aproximada $x^{(k+1)}$ é dada pela seguinte fórmula (GOLUB; LOAN, 2015, p. 513):

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \cdot \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} \right)$$

Onde x_i é a i-ésima linha do vetor solução aproximada, b_i é a i-ésima linha do vetor das constantes e a_{ij} é a entrada da matriz dos coeficientes na posição (i, j).

A convergência do método de Gauss-Seidel é garantida para qualquer aproximação inicial, contanto que seja cumprido o critério de Sassenfeld (RUGGIERO; LOPES, 1988, p. 171). Sejam:

$$\beta_1 = \frac{1}{|a_{11}|} \cdot \sum_{j=2}^{n} |a_{1j}|$$

e

$$\beta_i = \frac{1}{|a_{ii}|} \cdot \left(\sum_{j=1}^{i-1} \beta_j \cdot |a_{ij}| - \sum_{j=i+1}^{n} |a_{ij}| \right)$$

O critério de Sassenfeld diz que o método convergirá se $\beta = max_{1\leq i\leq n}(\beta_i) < 1$ (RUGGIERO; LOPES, 1988, p. 173).

Além disso, dada uma tolerância ε, as iterações do método de Gauss-Seidel devem ser interrompidas se $max_{1\leq i\leq n}|x_i^{(k)} - x_i^{(k+1)}| < \varepsilon$, ou seja se a maior distância entre os elementos do vetor solução aproximada nos passos k e $k + 1$ for menor do que a tolerância (RUGGIERO; LOPES, 1988, p. 155).

Assumindo um sistema com número igual de incógnitas e equações que cumpe o critério de Sassenfeld, tem-se o seguinte algoritmo para solução automática:

Algoritmo 2 Gauss-Seidel

```
função GAUSSSEIDEL(A, b, x^(0), ε, maxiter)
    x^(k) ← x^(0)
    x^(k+1) ← x^(k)
    para k ← 1 até maxiter faça
        para i ← 1 até N faça
            S_1 ← 0
            S_2 ← 0
            para j ← 1 até i − 1 faça
                S_1 ← S_1 + A_ij * x_j^(k+1)
            fim para
            para j ← i+1 até N faça
                S_2 ← S_2 + A_ij * x_j^(k)
            fim para
            x_i^(k+1) ← (1/A_ii) * (b_1 − S_1 − S_2)
        fim para
        se (max_{1≤i≤n} |x_i^(k) − x_i^(k+1)| < ε) então
            Pare
        fim se
        x^(k) ← x^(k+1)
    fim para
fim função
```

Assim com discutido por (C.CHAPRA; CANALE, 2016), a grande vantagem da utilização do método de Gauss-Seidel está na economia de memória do método, se comparado aos métodos diretos. A economia de recursos computacionais é extremamente importante para que se tenha eficiência na computação dos resultados.

IV. TEORIA DOS GRAFOS

A. *Definição*

Segundo (BONDY; MURTY, 1976, p. 1), um *grafo* é o terno ordenado $G = (V(G), E(G), \psi_G)$, onde $V(G) = v_1, v_2, \ldots, v_n, n \in \mathbb{N}$ é chamado de conjunto (não vazio) de *vértices*, $E(G) = e_1, e_2, \ldots, e_m, m \in \mathbb{N}$ é chamado de conjunto de *arestas* e ψ_G é chamada de *função de incidência*, que associa a cada aresta um par de vértices. Dados dois vértices u e v, a aresta e conecta u e v se $\psi(e) = uv$.

O nome *grafo* se deve à possibilidade de representar tais objetos matemáticos *graficamente* (BONDY; MURTY, 1976, p. 2). Por exemplo, seja um grafo G com 4 vértices e 5 arestas $G = (V(G), E(G), \psi_G)$, onde:

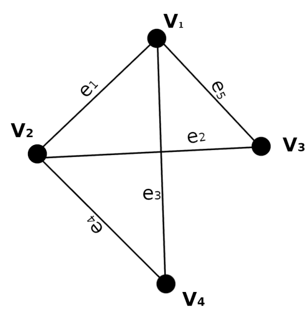
$$V(G) = v_1, v_2, v_3, v_4$$

$$E(G) = e_1, e_2, e_3, e_4, e_5$$

$$
\begin{aligned}
\psi(e_1) &= v_1v_2, \\
\psi(e_2) &= v_2v_3, \\
\psi(e_3) &= v_1v_4, \\
\psi(e_4) &= v_2v_4, \\
\psi(e_5) &= v_1v_3
\end{aligned}
$$

G pode facilmente ser representado graficamente pelo seguinte diagrama, onde cada vértice representa um ponto e cada aresta representa uma linha:

Figura 1: Exemplo (1) de um grafo



Fonte: Compilação dos autores.

Um *caminho* é dado por uma sequência de vértices. Um *ciclo* é um caminho através do qual é possível "sair" de um vértice e "voltar" para o mesmo por uma aresta diferente (SEDGEWICK; WAYNE, 2011, p. 519). Se este caminho for impossível, para todos os vértices, o grafo é dito *acíclico*.

Um grafo é *conexo* se existir pelo menos um caminho de qualquer vértice à qualquer outro vértice (SEDGEWICK; WAYNE, 2011, p. 519).

B. Matriz de Incidência

Uma forma conveniente de representar um grafo é pela chamada *matriz de incidência*. Seja G um grafo com v vértices e e arestas. G pode ser escrito em termos de uma matriz $v \times e$, isto é, $M(G) = [m_{ij}]$, onde m_{ij} (a entrada da matriz na i-ésima linha e na j-ésima coluna) representa quantas vezes (entre 0 e 2) v_i e e_j são *incidentes*, ou seja, quantas vezes a aresta e_j se "encontra" com o vértice v_i (BONDY; MURTY, 1976, p. 7). Dois vértices v_a e v_a possuem uma aresta e_c entre si se e somente se $m_{ac} = m_{bc} \neq 0$.

A matriz de incidência que representa o grafo do exemplo anterior é, nesse caso:

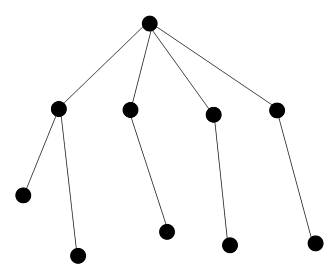
$$
M(G) = \begin{bmatrix}
 & e_1 & e_2 & e_3 & e_4 & e_5 \\
 & - & - & - & - & - \\
v_1| & 1 & 0 & 1 & 0 & 1 \\
v_2| & 1 & 1 & 0 & 1 & 0 \\
v_3| & 0 & 1 & 0 & 0 & 1 \\
v_4| & 0 & 0 & 1 & 1 & 0
\end{bmatrix}
$$

Verifica-se que, na matriz de incidência apresentada acima, $m_{11} = m_{21}$, o que indica que e_1 é uma conexão (aresta) entre v_1 e v_2. Nota-se também que três arestas incidem com v_1.

C. Árvore Geradora

Por definição, uma *árvore* é um grafo conexo acíclico (SEDGEWICK; WAYNE, 2011, p. 520), como explicitado na figura 2:

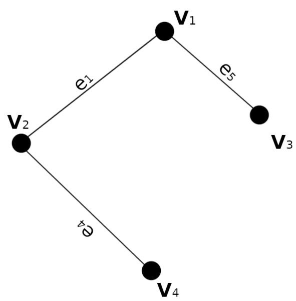
Figura 2: Exemplo de árvore



Fonte: Compilação dos autores.

Uma *árvore geradora* de um grafo conexo G é um subgrafo que contém todos os vértices (e não necessariamente todas as arestas) de G mas nenhum de seus ciclos. Uma árvore geradora do exemplo (1) pode ser:

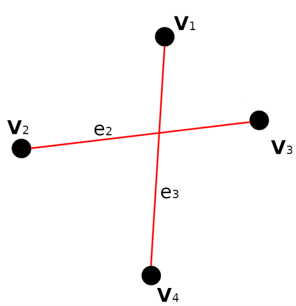
Figura 3: Árvore geradora de (1)



Fonte: Compilação dos autores.

Uma *árvore co-geradora* de um grafo conexo G é uma árvore formada pelas arestas (e seus respectivos vértices) que não aparecem na árvore geradora (THULASIRAMAN, 2002, p. 834). Suas arestas são chamadas de *acordes*. A árvore co-geradora do exemplo anterior é:

Figura 4: Árvore co-geradora de (1)

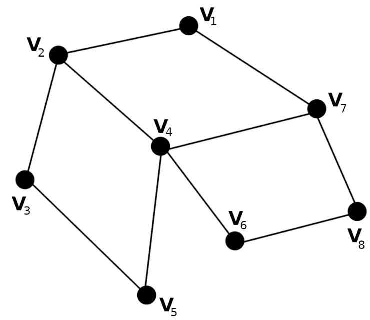


Fonte: Compilação dos autores.

É possível ainda encontrar uma árvore geradora para qualquer grafo conexo computacionalmente. Para tal, basta realizar uma *busca em profundidade* no grafo e então a árvore geradora será o caminho percorrido pela busca.

A busca em profundidade (ou DFS) consiste em dois passos simples: visitar um vértice inicial e recursivamente visitar todos os seus vizinhos ainda não visitados (SEDGEWICK; WAYNE, 2011, p. 531). Seja, por exemplo, o seguinte grafo G:

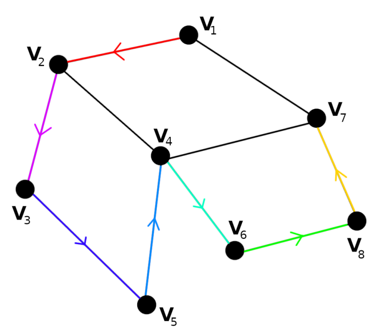
Figura 5: Exemplo (2) de um grafo



Fonte: compilação dos autores.

A seguinte figura mostra o percurso da DFS partindo de v_1, onde cada cor representa um passo do algoritmo:

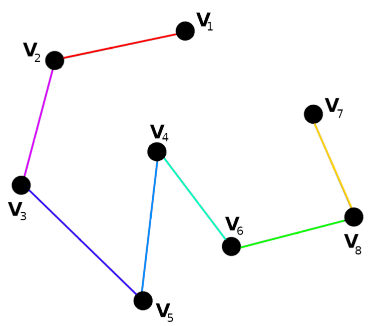
Figura 6: DFS sobre (2)



Fonte: Compilação dos autores.

É fácil perceber que o subgrafo formado pelas arestas destacadas é uma árvore geradora de G:

Figura 7: Arvore Gerada pela DFS



Fonte: Compilação dos autores.

O seguinte algoritmo representa uma implementação computacional do tópico discutido, assumindo um grafo conexo G e uma árvore inicialmente vazia T:

Algoritmo 3 Árvore Geradora

```
função ARVOREGERADORADFS(G, T, v)
    Marque v como visitado
    Insira v em T
    para Todos os vizinhos v_i não visitados de v faça
        Insira a aresta que conecta v e v_i em T
        arvoreGeradoraDFS(G, T, v_i)
    fim para
fim função
```

D. *Encontrando o Conjunto de Ciclos Fundamentais de um Grafo*

Assim como descrito por (THULASIRAMAN, 2002), ao adicionar um dos arcordes à árvore geradora de um grafo, obtém-se um grafo com exatamente um ciclo. Ao conjunto das arestas que formam este ciclo, dá-se o nome de ciclo fundamental. O conjunto dos ciclos obtidos ao se adicionar, separadamente, cada acorde à árvore geradora é denotado por conjunto fundamental de ciclos.

A seguir, é apresentado um algoritmo para a obtenção do conjunto de ciclos fundamentais de um grafo, com base nas ideias apresentadas por (THULASIRAMAN, 2002). A ideia geral é encontrar uma árvore geradora T do grafo G e a cada passo inserir um *acorde* $e_i \in (E(G) - E(T))$ em T, encontrar o ciclo gerado pela adição de e_i (este processo pode ser facilmente implementado através de pequenas modificações no algoritmo de busca em profundidade), então, remover e_i de T, até que todos os *acordes* tenham passado pelo processo.

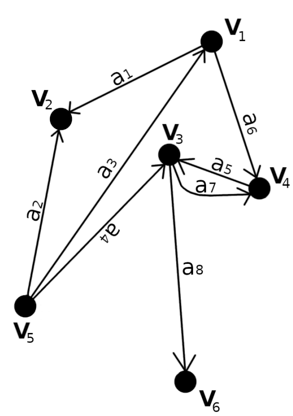
Algoritmo 4 Encontra Ciclos

```
função ENCONTRACICLOS(G)
    T ← arvoreGeradoraDFS(G, T, v_1)
    para Todas as arestas e_i em E(G) faça
        Insira e_i em T
        Encontre o único ciclo presente em T
        Remova e_i de T
    fim para
fim função
```

E. *Digrafo e Multigrafo*

Um grafo direcionado (ou digrafo) é um grafo no qual os vértices extremos de uma aresta podem (mas não necessariamente devem) estar conectados por uma via de mão única (SEDGEWICK; WAYNE, 2011, p. 566). Ou seja, dados dois vértices v_1 e v_2, uma aresta e_1 pode representar um caminho de v_1 à v_2 mas não necessariamente o contrário. Nesse caso, a aresta é chamada de *arco*. Graficamente, os arcos são representados por setas que ligam dois vértices:

Figura 8: Exemplo de digrafo



Fonte: Compilação dos autores.

Convencionando $m_{ij} = 1$ se o arco a_j *sai* do vértice v_i e $m_{ij} = -1$ se o arco a_j *chega* ao vértice v_i, a seguinte matriz de incidência representa o digrafo anterior:

$$M = \begin{array}{cc} & \begin{array}{cccccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 \end{array} \\ \begin{array}{c} v_1| \\ v_2| \\ v_3| \\ v_4| \\ v_5| \\ v_6| \end{array} & \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \end{array}$$

Note que há mais de um arco entre v_3 e v_4. A um grafo que permite esse tipo de comportamento se dá o nome de *multigrafo*.

V. Circuitos Elétricos

A. *Definição*

Segundo (SADIKU; ALEXANDER, 2013, p. 4), um *circuito elétrico* é basicamente uma conexão entre componentes elétricos. Estes componentes são chamados de *elementos* do circuito. Nesse trabalho são abordados apenas *fontes de tensão constante* e *resistores*.

Fontes de tensão constante são dispositivos que geram uma *força eletromotriz* (ou diferença de potencial), ou seja, realizam *trabalho* para deslocar *cargas elétricas* (SADIKU; ALEXANDER, 2013, p. 8). A unidade da diferença de potencial (DDP) é o *volt* (V) (SADIKU; ALEXANDER, 2013, p. 9). A esse deslocamento de cargas se dá o nome de *corrente elétrica*, medida em *ampères* (A) (SADIKU; ALEXANDER, 2013, p. 6).

Resistores são elementos que se submetidos à uma DDP, *opõem-se* à passagem de corrente elétrica (SADIKU; ALEXANDER, 2013, p. 28) de acordo com sua *resistência*. Essas três quantidades são relacionadas pela *Lei de Ohm*:

$$V = R \cdot I$$

Onde V é a diferença de potencial, R é a resistência do elemento resistivo e I é a corrente que atravessa o elemento (SADIKU; ALEXANDER, 2013, p. 28).

B. *Diagramas*

É conveniente, ainda, representar os componentes elétricos graficamente para análise, sem ser necessário que se construa um circuito concreto.

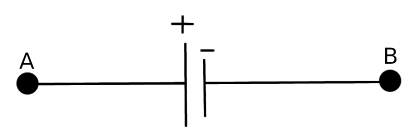
A seguinte figura mostra como um resistor inserido entre dois pontos A e B pode ser representado:

Figura 9: Diagrama de um resistor

Fonte: Compilação dos autores.

A próxima figura expõe a representação gráfica de uma fonte de tensão constante entre A e B:

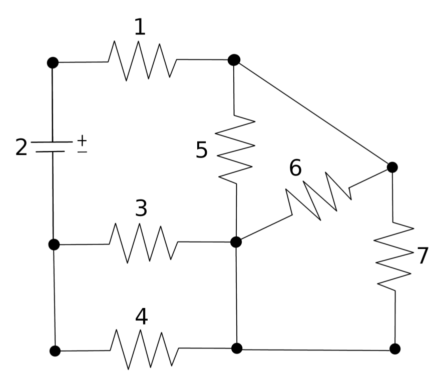
Figura 10: Diagrama de uma fonte de tensão



Fonte: Compilação dos autores.

Então, estes elementos podem ser combinados a fim de gerar circuitos mais complexos. A seguir, expõe-se um exemplo desse tipo:

Figura 11: Diagrama de um circuito resistivo

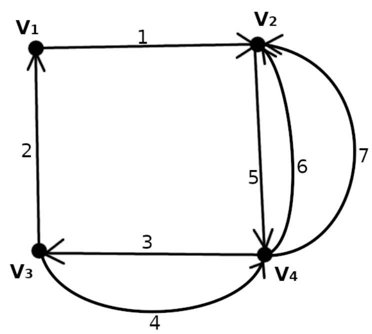


Fonte: Compilação dos autores.

C. *Representação por Grafos*

Apesar dos diagramas serem convenientes para seres humanos visualizarem o comportamento de circuitos elétricos, a interpretação direta deles é muito difícil computacionalmente. Note, porém, que o diagrama é composto basicamente por vértices e arestas. Então, é possível redesenhar o diagrama como um digrafo, tomando os vértices como pontos sob diferentes potenciais e os arcos (com a mesma numeração) como os componentes elétricos, assim como mostrado por (THULASIRAMAN, 2002, p. 838). Então, o seguinte grafo representa o circuito anterior:

Figura 12: Grafo do circuito resistivo



Fonte: Compilação dos autores.

O sentido dos arcos representa o sentido da corrente que será assumido na simulação. Então, a matriz de incidência resultante é:

$$M = \left[\begin{array}{c|ccccccc} & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 \\ & - & - & - & - & - & - & - \\ v_1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ v_2 & -1 & 0 & 0 & 0 & 1 & -1 & -1 \\ v_3 & 0 & 1 & -1 & 1 & 0 & 0 & 0 \\ v_4 & 0 & 0 & 1 & -1 & -1 & 1 & 1 \end{array}\right]$$

Os arcos que se incidem nos mesmos dois vértices são componentes em *paralelo*. Dois arcos que formam um caminho $v_i - v_j$, $v_j - v_k$, sendo os únicos arcos incidentes entre v_i, v_j e v_j, v_k, são dois componentes em *série*.

D. Solução Computacional

Resolver um circuito elétrico consiste em utilizar o *princípio de conservação da carga elétrica* e o *princípio de conservação de energia* para encontrar todas as variáveis desconhecidas dos componentes. Comumente são conhecidas apenas a tensão da fonte e as resistências dos resistores.

O método utilizado neste artigo consiste em uma adaptação dos métodos expostos em (THULASIRAMAN, 2002) e (BORUAH; GOGOI; CHUTIA, 2017).

A *Lei das Tensões de Kirchhoff* é a aplicação do princípio de conservação de energia aos circuitos elétricos e diz que (BORUAH; GOGOI; CHUTIA, 2017, p. 945):

$$\sum_{k=1}^{e} b_{rk} V_k = 0$$

Onde e é o número de arcos do grafo, b_{rk} é a entrada na posição rk da matriz B dos ciclos de G e V_k é a tensão através de um arco. Em outras palavras, diz que a soma das tensões em um ciclo é nula.

Já a *Lei das Correntes de Kirchhoff* enuncia o emprego do princípio da conservação de carga elétrica aos circuitos elétricos da seguinte forma (BORUAH; GOGOI; CHUTIA, 2017, p. 945):

$$\sum_{k=1}^{e} a_{rk} I_k = 0$$

Onde a_{rk} é a entrada na posição rk da matriz de incidência A do grafo G, I_k é a corrente que flui pela aresta k e e é o número de arcos do grafo. Em outras palavras, diz que a soma das correntes em um nodo é nula.

A matriz B dos ciclos fundamentais é uma matriz onde cada linha representa um ciclo fundamental do grafo G e cada coluna representa um arco presente no ciclo de acordo com as seguintes regras (BORUAH; GOGOI; CHUTIA, 2017, p. 948):

- $b_{ij} = 1$ se o galho b_j pertence ao ciclo fundamental c_i e sua referência têm a mesma orientação;
- $b_{ij} = -1$ se o galho b_j pertence ao ciclo fundamental c_i e sua referência têm orientações opostas;
- $b_{ij} = 0$ se o galho b_j não pertence ao ciclo fundamental c_i.

A matriz R das resistências é uma matriz diagonal quadrada $e \times e$, onde e é o número de arcos no grafo, cujos elementos r_{ii} representam a resistência no arco i.

Seja V a matriz coluna que representa as tensões das fontes de tensão de corrente contínua do circuito, R a matriz das resistências e I_b a matriz coluna que representa as correntes em cada um dos galhos (arestas) do circuito, tem-se, pela lei das tensões de kirchoff:

$$B(V - RI_b) = 0$$

$$\iff BV = BRI_b$$

Mas, sabemos que $I_b = B^t I_L$, onde I_L é a matriz de correntes em cada ciclo fundamental e B^t é a transposta de B. Então:

$$BV = BRB^t I_L$$

Como B, R e B^t são conhecidos e formam uma matriz quadrada, I_L é uma matriz coluna de incógnitas e BV é uma matriz coluna de constantes, a equação anterior é equivalente a um sistema linear na forma $Ax = b$. Logo, basta resolver o sistema e substituir em $I_b = B^t I_L$ para encontrar todas as correntes.

Desta forma, neste artigo, utilizou-se o seguinte algoritmo para a solução computacional de circuitos elétricos, assumindo A como a matriz de incidência do grafo, R como a matriz de resistência, B como a matriz dos ciclos fundamentais, I como a matriz das correntes e V como a matriz das tensões das fontes:

Algoritmo 5 Solucionador de Circuitos

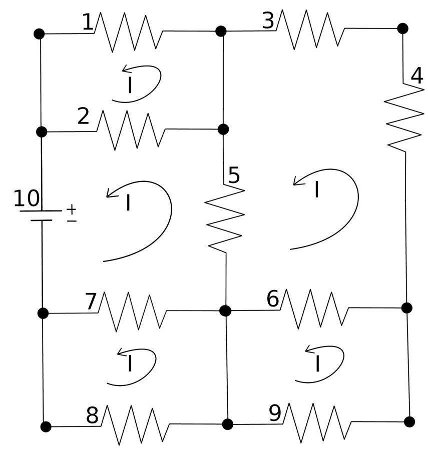
função RESOLVECIRCUITO(A, R, B, I, V)
  $B \leftarrow encontraCiclos(A)$
  $M \leftarrow BR_bB^t$
  $n \leftarrow BV_s$
  $x^{(0)} \leftarrow$ vetor nulo
  se M atende ao critério de Sassenfeld **então**
    $I_L \leftarrow gaussSeidel(M, n, x^{(0)}, 10^{-8}, 10^3)$
  senão
    $I_L \leftarrow gaussJordan(M, n)$
  fim se
  $I_b \leftarrow B^t I_L$
fim função

Ao fim do algoritmo, são conhecidas todas as correntes e tensões e, portanto, o circuito passa a ser determinado.

VI. RESULTADOS

Considere o seguinte circuito, onde a tensão na fonte é de $5V$ e todos os resistores possuem resistência de 100Ω:

Figura 13: Exemplo (1) de circuito



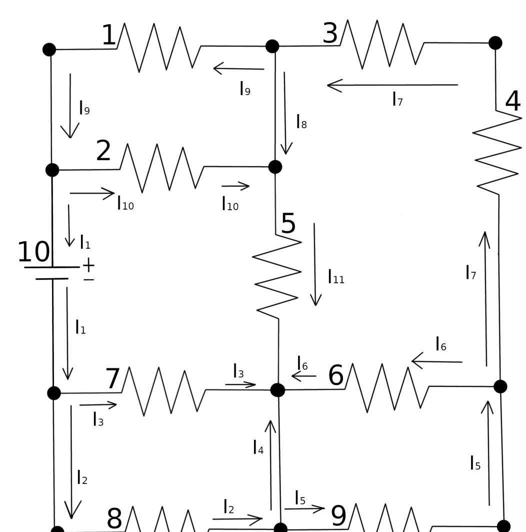
Fonte: Compilação dos autores.

É possível aplicar a lei das Tensões de Kirchhoff para cada malha, que resulta no seguinte sistema:

$$\begin{cases} V_1 + V_2 = 0 \\ V_{10} + V_7 + V_5 + V_2 = 0 \\ V_3 + V_4 + V_5 + V_6 = 0 \\ V_7 + V_8 = 0 \\ V_6 + V_9 = 0 \end{cases}$$

As correntes do circuito podem ser desenhadas, assumindo um sentido arbitrário e o sentido anti-horário como positivo, da seguinte forma:

Figura 14: Correntes de (1)



Fonte: Compilação dos autores.

Então, pode-se aplicar a lei das correntes de Kirchhoff ao circuito. Tem-se, então, o seguinte sistema:

$$\begin{cases} I_1 - I_2 - I_3 = 0 \\ I_2 - I_4 - I_5 = 0 \\ I_5 - I_6 - I_7 = 0 \\ I_3 + I_4 + I_6 + I_{11} = 0 \\ I_7 - I_8 - I_9 = 0 \\ -I_1 + I_9 - I_{10} + = 0 \\ I_8 + I_{10} - I_{11} = 0 \end{cases}$$

É possível ainda reescrever o primeiro sistema em função da Lei de Ohm:

$$\begin{cases} I_9 R + I_{10} R = 0 \\ V_{10} - I_3 R + I_{10} R + I_{11} R = 0 \\ -I_7 R - I_7 R + I_6 R - I_{11} R = 0 \\ -I_3 R + I_2 R = 0 \\ I_5 R + I_6 R = 0 \end{cases}$$

Que pode ser juntado com o segundo sistema:

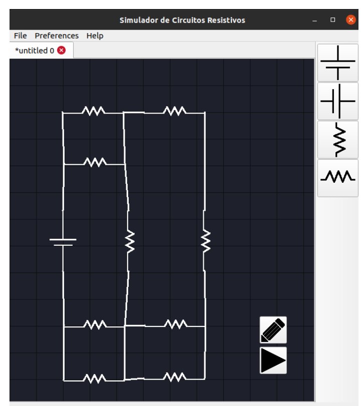
$$\begin{cases} I_9 R + I_{10} R = 0 \\ V_{10} - I_3 R + I_{10} R + I_{11} R = 0 \\ -I_7 R - I_7 R + I_6 R - I_{11} R = 0 \\ -I_3 R + I_2 R = 0 \\ I_5 R + I_6 R = 0 \\ I_1 - I_2 - I_3 = 0 \\ I_2 - I_4 - I_5 = 0 \\ I_5 - I_6 - I_7 = 0 \\ I_3 + I_4 + I_6 + I_{11} = 0 \\ I_7 - I_8 - I_9 = 0 \\ -I_1 + I_9 - I_{10} + = 0 \\ I_8 + I_{10} - I_{11} = 0 \end{cases}$$

Substituindo V_{10} e R e resolvendo para o vetor I, notando que os sinais negativos indicam uma escolha equivocada de sentido inicial para a corrente, temos:

$$I = \begin{bmatrix} 0,0291\bar{6} \\ 0,01458\bar{3} \\ 0,01458\bar{3} \\ 0,010417 \\ 0,0041\bar{6} \\ -0,0041\bar{6} \\ 0,008\bar{3} \\ -0,00625 \\ 0,01458\bar{3} \\ -0,01458\bar{3} \\ -0,0208\bar{3} \end{bmatrix}$$

Este mesmo circuito pode ser resolvido computacionalmente de forma trivial. Basta inserí-lo no programa desenvolvido para este artigo e checar os resultados, da seguinte forma:

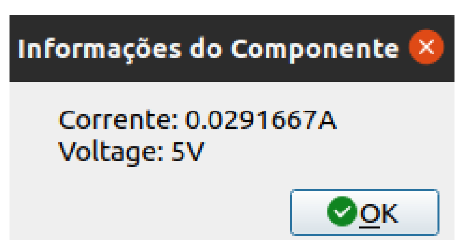
Figura 15: Simulação de (1) feita no programa desenvolvido pelos autores



Fonte: Compilação dos autores.

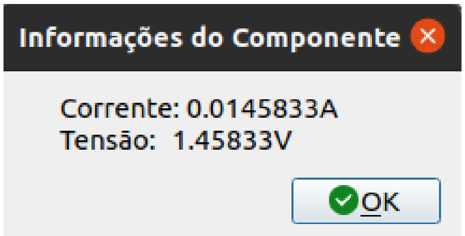
Com o mesmo sentido de corrente assumido, são dadas as seguintes correntes nos componentes:

Figura 16: Corrente na fonte



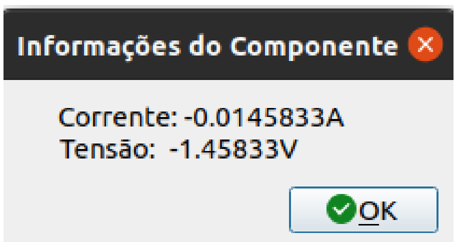
Fonte: Compilação dos autores.

Figura 17: Corrente em R_1



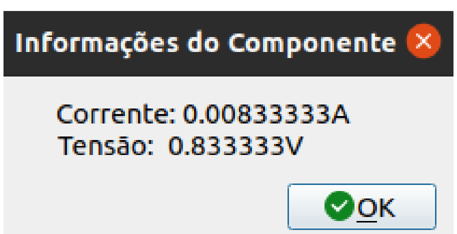
Fonte: Compilação dos autores.

Figura 18: Corrente em R_2



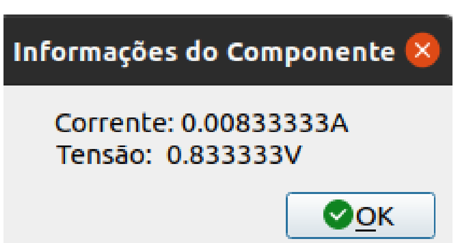
Fonte: Compilação dos autores.

Figura 19: Corrente em R_3



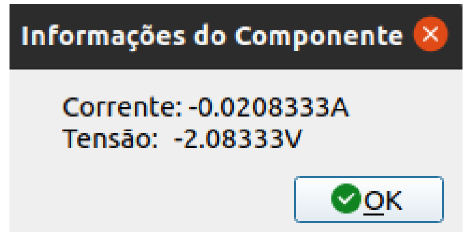
Fonte: Compilação dos autores.

Figura 20: Corrente em R_4



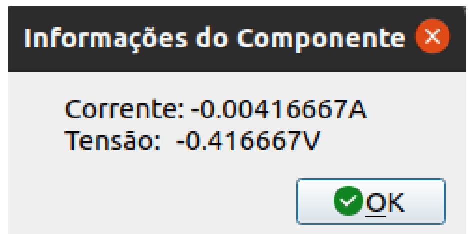
Fonte: Compilação dos autores.

Figura 21: Corrente em R_5



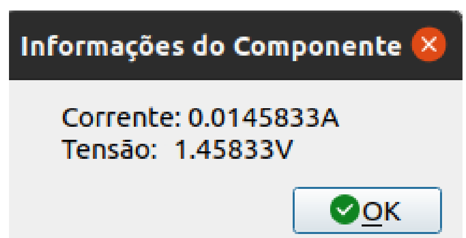
Fonte: Compilação dos autores.

Figura 22: Corrente em R_6



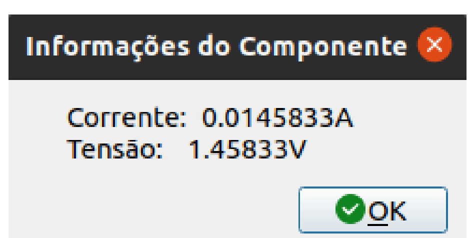
Fonte: Compilação dos autores.

Figura 23: Corrente em R_7



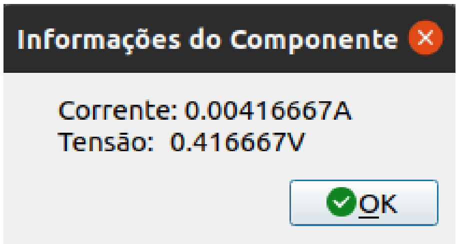
Fonte: Compilação dos autores.

Figura 24: Corrente em R_8



Fonte: Compilação dos autores.

Figura 25: Corrente em R_9



Fonte: Compilação dos autores.

As correntes negativas se devem ao fato de o sentido assumido para cada componente durante a resolução (ou montagem do circuito no simulador) ser contrário ao sentido real da corrente no circuito.

Note que apesar de não haver diferenças entre os resultados obtidos pela simulação e os resultados obtidos manualmente, a simulação pode gerar erros de arredondamento devido à representação numérica de ponto flutuante utilizada pelos computadores. Conforme (C.CHAPRA; CANALE, 2016), a fonte destes erros decorre, principalmente, da limitação de quantidades que podem representadas pelo sistema de ponto flutuante.

VII. CONCLUSÃO

Então, conclui-se que, a abordagem gera um resultado final satisfatório, já que apresenta a solução aproximadamente exata do problema proposto, facilitando e acelerando a obtenção dos resultados desejados.

Ademais, verifica-se que a utilização de métodos numéricos é de extrema importância na resolução de circuitos eletrônicos por meios computacionais, uma vez que estes podem, muitas vezes, convergir para o resultado de forma mais rápida que os métodos diretos, deste modo, diminuindo os custos computacionais envolvidos na resolução de sistemas lineares que, por sua vez, são cruciais para a obtenção dos resultados.

Vale ressaltar ainda que seria possível expandir os assuntos aqui discutidos a fim de produzir um solucionador que abranja mais tipos de componentes.

REFERÊNCIAS

ANTON, H.; RORRES, C. *Álgebra Linear*: com aplicações. 10. ed. São Paulo: BOOKMAN COMPANHIA EDITORA LTDA., 2010. 762 p. ISBN 978-04-704-3205-1.

BONDY, J. A.; MURTY, U. S. R. *Graph Theory*: With applications. 1. ed. Ontario: North Holland, 1976. 264 p. ISBN 0-444-19451-7.

BORUAH, C.; GOGOI, K.; CHUTIA, C. Analysis of some electrical circuits with the help of graph theory using network equilibrium equations. *International Journal of Innovative Research in Science, Engineering and Technology*, India, v. 6, n. 1, p. 944–953, jan. 2017.

C.CHAPRA, S.; CANALE, R. P. *Métodos Numéricos para Engenharia*. 7. ed. Porto Alegre: Amgh, 2016. ISBN 9788580555684.

GOLUB, G. H.; LOAN, C. F. V. *Matrix Computations*. 3. ed. Baltimore: The Johns Hopkins University Press, 2015. 694 p. ISBN 978-0-8018-5414-9.

PIRES, A. de A. *Cálculo Numérico*: Prática com algoritmos e planilhas. 1. ed. São Paulo: Editora Atlas S.A., 2015. 225 p. ISBN 978-85-224-9882-6.

RUGGIERO, M. A. G.; LOPES, V. L. da R. *Cálculo Numérico*: Aspectos teóricos e computacionais. 2. ed. São Paulo: Makron Books, 1988. 424 p. ISBN 978-85-346-0204-4.

SADIKU, M. N. O.; ALEXANDER, C. K. *Fundamentos de Circuitos Elétricos*. 5. ed. São Paulo: AMGH Editora Ltda., 2013. 874 p.

SEDGEWICK, R.; WAYNE, K. *Algorithms*. 4. ed. Nova Jersey: Addison-Wesley, 2011. 956 p. ISBN 978-0-321-57351-3.

THULASIRAMAN, K. Circuit theory. *Encyclopedia of Physical Science and Technology*, Oklahoma, v. 2, p. 831–841, 2002.