

Transientes Em Linhas De Transmissão Sem Perdas

J. E. Oliveira, V. R. Leonardi and L. R. d. Souza

Abstract—Esse relatório tem como objetivo explicar e detalhar o algoritmo de diferenças finitas no domínio do tempo (FDTD) e seu uso na análise computacional do comportamento de transientes de tensão e corrente em linhas de transmissão sem perdas, particularmente com o uso do programa Octave.

Keywords— Linhas de transmissão sem perdas; FDTD; transientes; Octave; visualização de tensão e corrente; simulação.

I. INTRODUÇÃO

LINHAS de transmissão são circuitos utilizados para a transmissão de sinais eletromagnéticos. Elas se tornam necessárias quando o comprimento das ondas transmitidas é curto o suficiente para que o comprimento da linha seja uma parcela relevante desse valor [1].

Transientes são eventos de curta duração causados por mudanças abruptas no estado do circuito. No caso das linhas de transmissão, transientes podem ocorrer quando uma nova carga é conectada ao circuito, quando um sinal aperiódico é injetado na ou se propaga através da linha, ou quando o gerador é ligado [2].

A análise feita nesse projeto se baseia no transiente gerado por sinais aperiódicos do tipo degrau. Para que seja computacionalmente possível tal análise, é necessário realizar uma discretização do sistema. O método FDTD permite transformar as equações do telegrafista (1) e (2) em diferenças finitas, criando uma malha de pontos discretos no espaço e no tempo que serão utilizados para os cálculos de $v(z,t)$ e $i(z,t)$.

$$\frac{-\partial v(z,t)}{\partial z} = Ri(z,t) + L \frac{\partial i(z,t)}{\partial t} \quad (1)$$

$$\frac{-\partial i(z,t)}{\partial z} = Gv(z,t) + C \frac{\partial v(z,t)}{\partial t} \quad (2)$$

O programa criado permite o cálculo e a representação gráfica da tensão e corrente ao longo do comprimento da linha, com uma visualização em forma de animação, destacando a mudança dos valores ao longo do tempo e do comprimento da linha.

Foi escolhido o programa Octave, que é open source, para realizar a simulação, devido à facilidade da representação gráfica da tensão e corrente para ilustrar o comportamento em relação ao tempo.

II. DADOS DA LINHA DE TRANSMISSÃO

Para a correta realização da simulação é preciso especificar as características da linha utilizada. Primeiramente, o sistema simulado é baseado no modelo de linha de transmissão na Fig. 1.

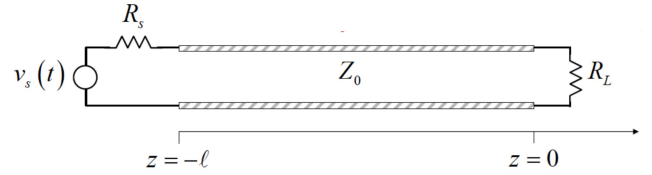


Figura 1. Modelo genérico de linha de transmissão[2].

Das especificações do projeto tem-se os seguintes valores:

$$Z_0 = 50 \Omega \quad R_s = 75 \Omega$$

A velocidade de propagação do sinal na linha é dada por:

$$u_f = 0.9c, \text{ onde } c \text{ é a velocidade da luz no vácuo.}$$

Como é o caso de uma linha sem perdas, os parâmetros R e G podem ser considerados $R = G = 0$. Os parâmetros L e C da

linha foram derivados de $Z_0 = \frac{1}{Cu_f}$ e $Z_0 = \sqrt{\frac{L}{C}}$.

Dessa forma, $C = 74,07 \text{ pF/m}$ e $L = 52,37 \text{ nH/m}$. O comprimento total da linha foi considerado $z = 10\text{m}$. Em questão dos valores da fonte e da carga acoplados, foram feitos testes para 2 fontes de tensão:

$$v_{s1} = 2u(t), v_{s2} = u(t) - u\left(t - \frac{z}{10u_f}\right), \text{ onde } u(t) \text{ é a}$$

função degrau. Para cada uma das fontes foram testados 3 valores de carga: $R_{L1} = 0$, $R_{L2} =$ e $R_{L3} = 100 \Omega$.

III. ALGORITMO FDTD

Para a linha de transmissão sem perdas, as equações do telegrafista (1) e (2) podem ser simplificadas com $R = G = 0$, resultando nas equações (3) e (4).

$$\frac{-\partial v(z,t)}{\partial z} = L \frac{\partial i(z,t)}{\partial t} \quad (3)$$

¹J. E. Oliveira, Universidade de São Paulo, São Carlos, São Paulo, Brasil, jeosvici@usp.br, NUSP 10716491

V. R. Leonardi, Universidade de São Paulo, São Carlos, São Paulo, Brasil, vitorleonardi@usp.br, NUSP 10716567

L. R. de Souza, Universidade de São Paulo, São Carlos, São Paulo, Brasil, leo_souza@usp.br, NUSP

Corresponding author: Julio Eiji Oliveira

$$\frac{-\partial i(z, t)}{\partial z} = C \frac{\partial v(z, t)}{\partial t} \quad (4)$$

A partir dessas equações pode-se aplicar o método FDTD para transformá-las em diferenças finitas e criar a malha dos pontos necessários.

Defini-se então um espaçamento fixo Δz no espaço e Δt no tempo, de tal modo que os pontos da malha podem ser descritos como:

$$z_k = k \Delta z \quad \text{e} \quad t_n = n \Delta t$$

Onde $1 \leq k \leq K, 1 \leq n \leq N$

Em seguida, aproxima-se o valor da derivada de tensão no espaço por uma diferença finita central:

$$\frac{\partial v(z, t)}{\partial z} \approx \frac{v(z + \Delta z, t) - v(z - \Delta z, t)}{2 \Delta z} \quad (5)$$

Utilizando uma notação mais conveniente para a tensão, tal que $v(z_k, t_n) = v_k^n$, temos:

$$\frac{\partial i_k^n}{\partial z} \approx \frac{i_{k+1}^n - i_{k-1}^n}{2 \Delta z} \quad (6)$$

De maneira similar, aplicando uma diferença finita no tempo, para a tensão temos:

$$\frac{\partial v_k^n}{\partial t} \approx \frac{v_k^{n+1} - v_k^{n-1}}{2 \Delta t} \quad (7)$$

É possível melhorar a precisão sem custo adicional de computação ao tomar-se meio passos entre cada ponto. Assim:

$$\frac{\partial i_k^n}{\partial z} \approx \frac{i_{k+1/2}^n - i_{k-1/2}^n}{\Delta z} \quad (8)$$

$$\frac{\partial v_k^n}{\partial t} \approx \frac{v_k^{n+1/2} - v_k^{n-1/2}}{\Delta t} \quad (9)$$

Utilizando as equações (8) e (9) nas equações (3) e (4) tem-se a aproximação numérica das equações do telegrafista que pode ser usada para o cálculo computacional:

$$-\frac{i_{k+1/2}^n - i_{k-1/2}^n}{\Delta z} = C \frac{v_k^{n+1/2} - v_k^{n-1/2}}{\Delta t} \quad (10)$$

$$-\frac{v_{k+1}^{n+1/2} - v_k^{n+1/2}}{\Delta t} = L \frac{i_{k+1/2}^{n+1} - i_{k+1/2}^n}{\Delta z} \quad (11)$$

O algoritmo FDTD consiste em encontrar o valor futuro de corrente e tensão a partir de valores passados conhecidos, então, rearranjando (10) e (11) tem-se como encontrar os valores futuros :

$$v_k^{n+1/2} = -\frac{\Delta t}{\Delta z C} (i_{k+1/2}^n - i_{k-1/2}^n) + v_k^{n-1/2} \quad (12)$$

$$i_{k+1/2}^{n+1} = -\frac{\Delta z}{\Delta t L} (v_{k+1}^{n+1/2} - v_k^{n+1/2}) + i_{k+1/2}^n \quad (13)$$

Então, para um tempo $n+1/2$, o algoritmo calcula os valores v para em todas as posições k e em seguida faz o mesmo para os valores de i na posição $k+1/2$ e tempo $n+1$. No final de uma iteração o n é aumentado e o ciclo continua até que $n > N$. Neste caso, N é o valor do tempo para ser atingido o regime estacionário: $N = \frac{10z}{u_f}$.

IV. ESTABILIDADE

Na simulação do algoritmo é preciso tomar cuidado com a estabilidade do sistema: a escolha de valores impróprios de Δx e Δt podem levar os resultados a divergirem, criando erros. Sendo assim, é necessário definir os valores que permitirão manter o sistema estável.

Analisando um caso simples na linha de transmissão sem perda, onde a onda gerada não é refletida, tem-se a seguinte solução para as equações do telegrafista, dadas no formato FDTD:

$$v_k^n = V_0^+ \cos(\omega n \Delta t - \beta k \Delta z) \quad (14)$$

$$i_k^n = \frac{V_0^+}{Z_0} \cos(\omega n \Delta t - \beta k \Delta z) \quad (15)$$

Nota-se que o termo de fase é zero. Tomando $V_0^+ = 1V$ e colocando as equações (14) e (15) em (12) chega-se em uma soma de cossenos. Assumindo um caso extremo onde $v_k^{n+1} = 1$ e $v_k^{n-1/2} = -1$ tem-se:

$$2 = -\frac{\Delta t}{\Delta z C Z_0} [\cos(A) - \cos(B)] \quad (16)$$

Onde:

$$A = \omega(n) \Delta t - \beta(k+1/2) \Delta z \quad \text{e}$$

$$B = \omega(n) \Delta t - \beta(k-1/2) \Delta z$$

Então, o caso máximo será aquele em que $\cos(A) - \cos(B) = 2$. A estabilidade do sistema dependerá apenas de $-\frac{\Delta t}{\Delta z C Z_0}$. Como é um caso máximo, deve-se definir um valor que garanta:

$$2 \leq -\frac{\Delta t}{\Delta z C Z_0} 2 \rightarrow 1 \leq -\frac{\Delta t}{\Delta z C Z_0} \quad (17)$$

Substituindo $Z_0 = \frac{1}{C u_f}$, onde u_f é a velocidade de propagação do sinal e resolvendo para o espaçamento de tempo:

$$\Delta t \leq \frac{\Delta z}{u_f} \quad (18a)$$

Este é o valor máximo que o espaçamento de tempo pode ter para garantir a estabilidade do sistema.

Para o espaçamento no tempo, o valor depende da frequência máxima da onda que viaja e, portanto, do comprimento de onda mínimo:

$$\Delta z \leq \frac{\lambda_{min}}{10} \quad (18b)$$

V. CONDIÇÕES DE FRONTEIRA

A tensão das extremidades da linha, na fonte e na carga, não podem ser calculados pelas equações (12) e (13). Sendo K o máximo valor de k pertencente a malha, e 1 o valor mínimo, para calcular esses valores seria necessário valores não pertencentes à malha. Então os valores de $v_K^{n+1/2}$ e $v_1^{n+1/2}$ devem ser definidos separadamente.

No caso da carga em curto-circuito:

$$v_K^{n+1/2} = 0 \quad (19)$$

No caso da carga em aberto:

$$v_K^{n+1/2} = v_{K-1}^{n+1/2} \quad (20)$$

Para calcular a influência da fonte, pode-se analisar o primeiro “nódo” da linha de transmissão, aplicando seu equivalente de Norton, como mostra a Fig. 2.

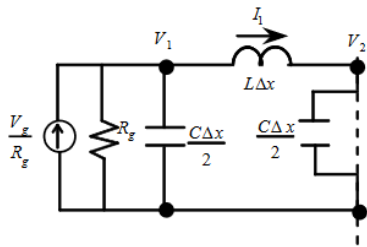


Figura 2. Equivalente de Norton do primeiro nódo da linha de transmissão[4].

Aplicando a LKC tem-se:

$$\frac{v_g^n}{R_g} = i_1^n + \frac{C \Delta x}{2} \frac{\partial v_1^n}{\partial t} + \frac{v_1^n}{R_g} \quad (21)$$

$$\frac{v_g^{n+1/2} + v_g^{n-1/2}}{2 R_g \Delta x} = \frac{i_1^n}{\Delta x} + C \frac{v_1^{n+1/2} - v_1^{n-1/2}}{2 \Delta t} + \frac{v_1^{n+1/2} - v_1^{n-1/2}}{2} \frac{1}{R_g \Delta x} \quad (22)$$

$$v_1^{n+1/2} = \left(\frac{R_g C \Delta x}{2 \Delta t} - \frac{1}{2} \right) v_1^{n-1/2} - \left(\frac{1}{\frac{C \Delta x}{2 \Delta t} + \frac{1}{2}} \right) \left(R_g i_n - \frac{v_g^{n+1/2} + v_g^{n-1/2}}{2} \right) \quad (23)$$

A equação (23) dá o valor de tensão para k=1.

Analogamente para a influência da carga temos o último “nódo” da linha na Fig. 3.

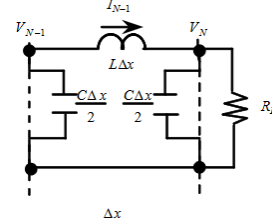


Figura 3. Carga e último componente de malha[4].

Aplicando a LKC:

$$v_K^{n+1/2} = \left(\frac{R_L C \Delta x}{2 \Delta t} - \frac{1}{2} \right) v_K^{n-1/2} - \left(\frac{1}{\frac{C \Delta x}{2 \Delta t} + \frac{1}{2}} \right) R_L i_{K-1}^n \quad (24)$$

A equação (24) dá o valor de tensão para k=K.

VI. IMPLEMENTAÇÃO

O programa foi implementado no Octave e é dividido em três partes. Ele está disponível em <https://github.com/VitorLeonardi/OndasEM>, junto com um arquivo README que contém instruções de uso. Primeiramente são definidas as constantes, como os dados da linha de transmissão e o passo de tempo e espaço utilizados. No caso da tensão da fonte e da carga no final da linha, o usuário deve indicar quais valores serão utilizados na caixa de diálogo inicial, como na Fig. 4.

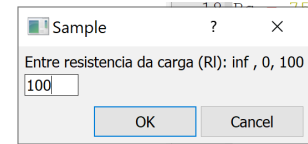


Figura 4. Caixa de escolha do valor da resistência.

A segunda parte consiste da loop principal da iteração. Aqui são calculados e atualizados os valores da corrente e tensão ao longo da linha.

for n = 2:Lt-1

#condicao de fronteira z = -Z (fonte)

$v(n,1) = (k1/k2) * v(n-1,1) - (Rs * i(n,1) - (Vs((n+1/2)*dt,m,uf,Z) + Vs((n-1/2)*dt,m,uf,Z))/2)/k2;$

#equacao de update tensão

```

#codigo vetorizado, linha sem perdas
v(n,2:Lz-1) = v(n-1,2:Lz-1) - dt*(i(n,2:Lz-1) - i(n,1:Lz-2))/(C*dz);

#condicao de fronteira z = 0 (carga)
if (Rl == 0)

    v(n,Lz) = 0;

    i(n+1,Lz) = i(n+1,Lz-1);

elseif (Rl == inf)

    v(n,Lz) = v(n,Lz-1);

    i(n+1,Lz) = 0;

else

    v(n,Lz) = (k1/k2)*v(n-1,Lz) + Rl*i(n,Lz-1)/k2;

    i(n+1,Lz) = v(n,Lz)/Rl;

endif

#equacao de update corrente

i(n+1,1:Lz-1) = i(n,1:Lz-1) - dt*(v(n,2:Lz)-v(n,1:Lz-1))/(L*dz);

endfor

```

A terceira parte é responsável pela representação gráfica dos resultados. A janela principal apresenta os gráficos de tensão e corrente em cada posição da linha. Estes gráficos são atualizados no tempo e representam uma animação do comportamento das ondas. A janela secundária apresenta o tempo decorrido da simulação.

```

figure('Name',[Rl = , num2str(Rl), ' m = ', num2str(m)], 'NumberTitle','off');

V = v(1,:);

I = i(1,:);

subplot(2,1,1);

plot(z, V, 'ydatasource','V');

title('tensão');

xlabel("z (m)");

ylabel("v (V)");

axis([-Z 0 min(min(v)) max(max(v))]);

subplot(2,1,2);

plot(z, I, 'ydatasource','I');

title('corrente');

xlabel("z (m)");

ylabel("i (A)");

axis([-Z 0 min(min(i)) max(max(i))]);

```

```

w = waitbar(0,'t = 0s');

#plot 1 a cada M, mais rapido e pouca perda de fidelidade visual, depende da precisao

M = round(p/100);

for n = 1:Lt/M

    I = i(M*n,:);

    V = v(M*n,:);

    if (mod(n,10) == 0)

        s = num2str(M*n*dt);

        waitbar(M*n/Lt,w,['t = 's','s']);

    endif

    refreshdata

    drawnow

endfor

s = num2str(M*n*dt);

waitbar(1,w,['Finalizado 't = 's','s']);

clear

```

VII. RESULTADOS

No total foram realizados 6 testes. Foram utilizados $Z = 10$ m, $\Delta z = 0.0001$ m e $\Delta t = 0,3086$, ps para todas as simulações.

Após a fonte $v_{st} = 2u(t)$ ser ligada, a onda viaja em direção à carga. A Fig. 4 apresenta o estado da linha antes de ocorrer reflexão.

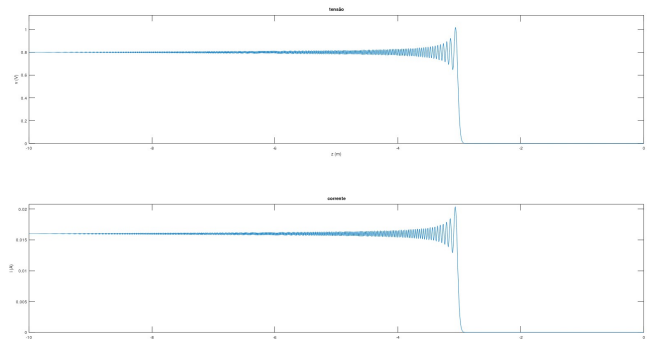


Figura 4. Ondas de tensão (acima) e corrente (abaixo) antes de ocorrer uma reflexão.

O valor da tensão é aproximadamente 0.8V após a passagem da frente de onda. Esse valor independe da carga acoplada a linha de transmissão, mas depende das características da linha e da resistência da fonte.

SIMULAÇÃO 1

Fonte: $v_{sI} = 2u(t)$ Carga: 0

A carga 0 representa o curto-circuito no sistema. A Fig. 5 representa a linha após a primeira reflexão.

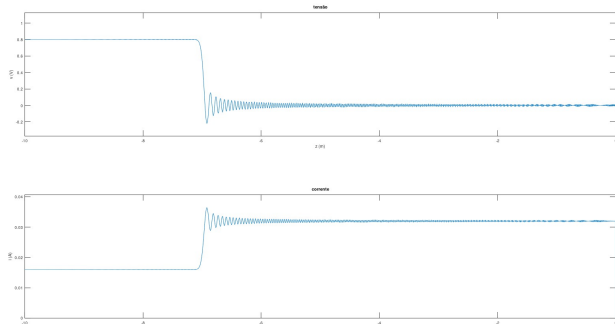


Figura 5. Linha após a primeira reflexão.

Aqui a onda de tensão refletida tem um comportamento destrutivo, de mesmo módulo, em relação a onda inicial, levando o valor da tensão a cair para 0. A onda de corrente, por outro lado, sofre uma interferência construtiva, aumentando seu valor.

No regime estacionário, Fig. 6, pode-se ver que a tensão tende à 0, o que é de se esperar para uma carga de curto-circuito, e que a corrente tende à um valor próximo de 0,03.

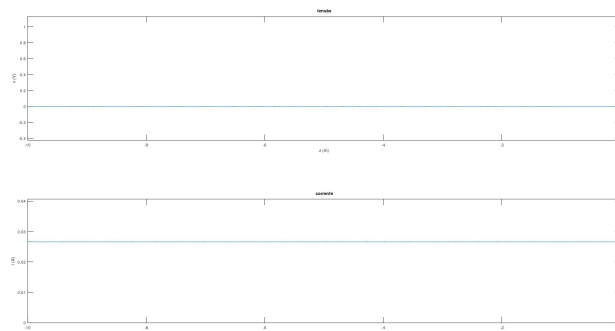


Figura 6. Linha no regime estacionário para $R_l = 0$.

SIMULAÇÃO 2

Fonte: $v_{sI} = 2u(t)$ Carga: ∞

A carga infinita representa o circuito aberto. Após a primeira reflexão, Fig. 7, observa-se um comportamento similar à carga de curto-circuito. Neste caso, porém, a corrente sofre interferência destrutiva da onda refletida, tendendo ao valor 0 e a tensão sofre interferência construtiva.

No regime estacionário, é de se esperar que o valor de tensão tenda ao próprio valor da fonte, 2V, e a corrente tenda a 0. A Fig. 8 mostra que, de fato, esses são os valores atingidos nessa simulação.

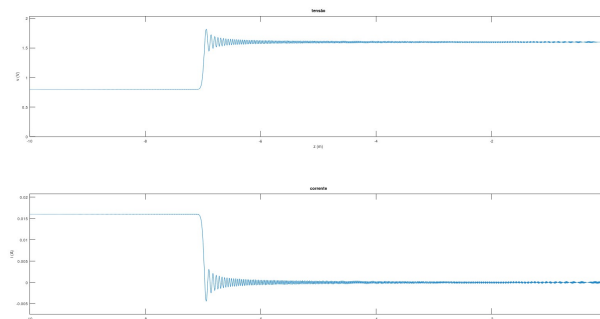


Figura 7. Linha após reflexão para carga de circuito aberto.

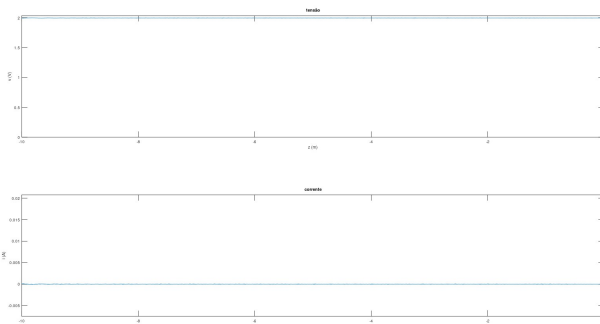


Figura 8. Linha no regime estacionário para carga de circuito aberto.

SIMULAÇÃO 3

Fonte: $v_{sI} = 2u(t)$ Carga: 100Ω

Nos casos anteriores as ondas refletidas não sofrem atenuação na carga, embora a tensão sofra alteração de sinal para carga 0 e a corrente para carga infinita.

Neste caso, a corrente ainda sofre alteração do sinal, porém ambas as ondas de tensão e corrente sofrem atenuação na carga, mostrado na Fig. 9, após a reflexão.

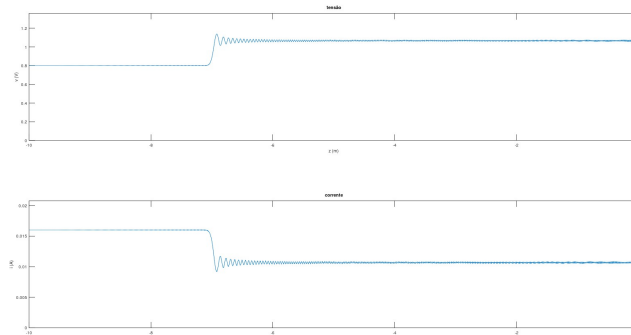


Figura 9. Linha após primeira reflexão para carga de 100 ohms.

Isso significa que no regime estacionário nenhuma das ondas será 0. A Fig. 10 mostra a convergência da tensão para um valor próximo de 1.2V e de corrente próximo de 0,01A

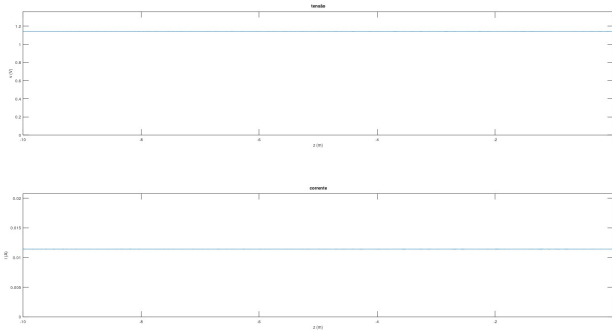


Figura 10. Regime estacionário para carga 100 ohms.

Nas próximas simulações foi utilizada a segunda fonte, que gera um pulso unitário de duração $\frac{Z}{10u_f}$ segundos. A Fig. 11 apresenta o pulso antes de qualquer reflexão.

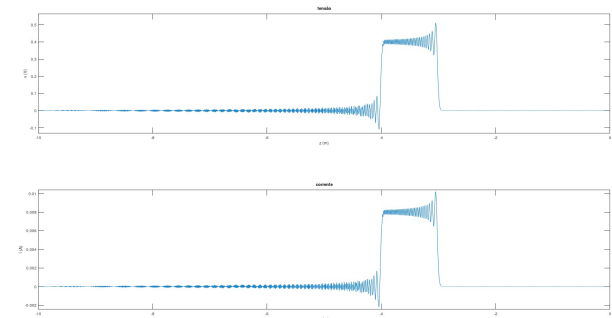


Figura 11. Pulso unitário antes da reflexão.

A tensão média no pulso é aproximadamente 0.4V, metade do valor encontrado nos testes da fonte 1, o que é de se esperar pois a fonte 2 apresenta metade da sua amplitude.

É importante notar que, como o sinal é um pulso, tanto a tensão quanto a corrente serão 0 no regime estacionário, já que as ondas sofrerão atenuação, ao menos na fonte, até se dissiparem, Fig. 12.

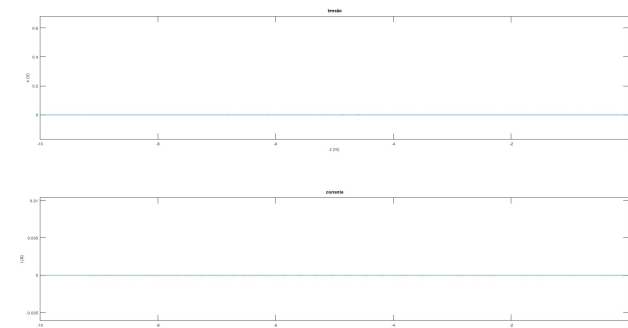


Figura 12. Regime estacionário para fonte 2. Independente da carga, as ondas tendem à zero.

SIMULAÇÃO 4

Fonte: $v_{s2} = u(t) - u\left(t - \frac{Z}{10u_f}\right)$ Carga: 0

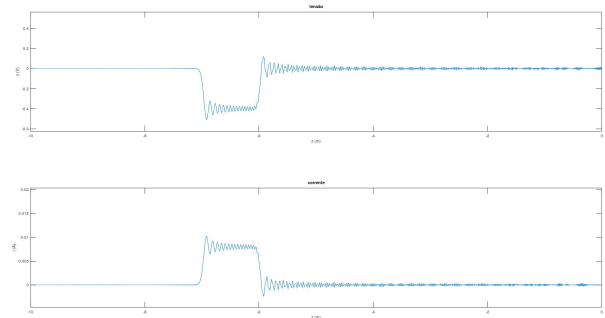


Figura 13. Pulso após a primeira reflexão.

Neste caso, observa-se a inversão do sinal da tensão, enquanto a corrente mantém o mesmo sinal. Nenhuma das ondas sofre atenuação na carga, como mostra a Fig.13.

SIMULAÇÃO 5

Fonte: $v_{s2} = u(t) - u\left(t - \frac{Z}{10u_f}\right)$ Carga: ∞

Assim como na simulação 2, aqui se observa um comportamento similar ao caso anterior, onde as ondas não sofrem atenuação na carga e a corrente tem seu sinal invertido enquanto a tensão se mantém (Fig.15).

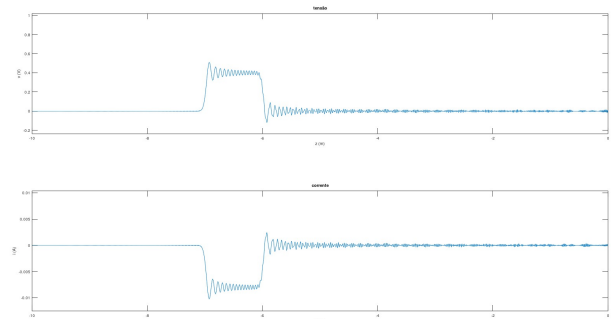


Figura 15. Pulso após primeira reflexão. Ocorre inversão do sinal da corrente.

SIMULAÇÃO 6

Fonte: $v_{s2} = u(t) - u\left(t - \frac{Z}{10u_f}\right)$ Carga: 100Ω

Finalmente, é de se esperar que tanto a corrente como a tensão sofram atenuação na carga neste caso.

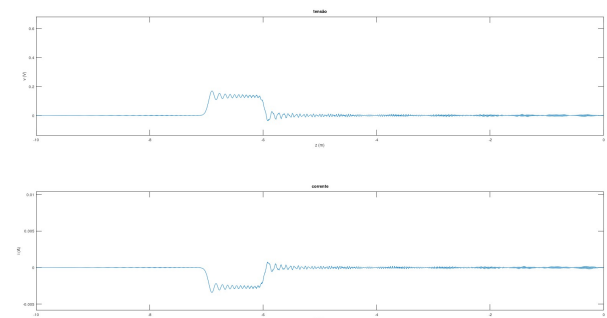


Figura 16. Pulso após primeira reflexão. Ocorre atenuação das ondas e inversão do sinal da corrente.

A Fig.16 mostra que, de fato, ocorre atenuação das ondas e também a inversão do sinal da corrente.

VIII. CONCLUSÃO

Analizando os resultados, é possível dizer que a implementação do algoritmo FDTD foi bem sucedida: o comportamento das ondas é similar ao esperado nas condições de carga utilizadas. Além disso, esses testes demonstram o valor que a simulação computacional tem para entender o comportamento de circuitos sem ser necessário o gasto de tempo e recursos para a montagem do circuito físico.

AGRADECIMENTOS

Agradecimento especial para o Monitor PAE Luiz Felipe Machado Votto para o Prof. Dr. Leonardo André Ambrosio e para o colega de turma Alex Marcelino Santee e ao Prof. Dr. Stephen D. Gedney.

REFERÊNCIAS

- [1] “Transmission line,” *Wikipedia*. May 03, 2020, Accessed: May 19, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Transmission_line&oldid=954668391.
- [2] F. Rana, “Time Domain Analysis of Transmission Lines,” Cornell University, 2007, [Online]. Available: <https://courses.cit.cornell.edu/ece303/Lectures/lecture24.pdf>.
- [3] L. Sevgi and C. Uluksik, “A MATLAB-based transmission-line virtual tool: finite-difference time-domain reflectometer,” *IEEE Antennas Propagation Magazine*, vol. 48, no. 1, pp. 141–145, Feb. 2006, doi: 10.1109/MAP.2006.1645595.
- [4] S. Gedney, *Computational Electromagnetics: The Finite-Difference Time-Domain*. University of Kentucky, 2003.
- [5] “Lab 4 -- The FDTD Method.” <https://utah.instructure.com/courses/297816/assignments/1750259> (accessed May 23, 2020).