

**UNIVERSIDADE ESTADUAL PAULISTA**  
**"JÚLIO DE MESQUITA FILHO"**  
**CAMPUS DE SÃO JOÃO DA BOA VISTA**

**CAIO VINÍCIUS COLOZZO GRILO**

**Simulações de Transitórios Eletromagnéticos em Ambiente Python**

São João da Boa Vista

2021

**Caio Vinícius Colozzo Grilo**

**Simulações de Transitórios Eletromagnéticos em Ambiente Python**

Trabalho de Graduação apresentado ao Conselho de Curso de Graduação em Engenharia Eletrônica e de Telecomunicações do Campus de São João da Boa Vista, Universidade Estadual Paulista, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia Eletrônica e de Telecomunicações .

Orientador: Prof<sup>o</sup> Dr. Afonso José do Prado

São João da Boa Vista

2021

G859s

Grilo, Caio Vinícius Colozzo

Simulações de transitórios eletromagnéticos em ambiente python /  
Caio Vinícius Colozzo Grilo. -- São João da Boa Vista, 2021  
61 p. : il., tabs.

Trabalho de conclusão de curso (Bacharelado - Engenharia de  
Telecomunicações) - Universidade Estadual Paulista (Unesp), Câmpus  
Experimental de São João da Boa Vista, São João da Boa Vista  
Orientador: Afonso José do Prado

1. Python (Linguagem de programação de computador). 2.  
Telecomunicações. 3. Transitórios (Eletricidade). I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Câmpus  
Experimental de São João da Boa Vista. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

**UNIVERSIDADE ESTADUAL PAULISTA**  
**“JÚLIO DE MESQUITA FILHO”**  
**CÂMPUS EXPERIMENTAL DE SÃO JOÃO DA BOA VISTA**  
**GRADUAÇÃO EM ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÕES**

**TRABALHO DE CONCLUSÃO DE CURSO**

**SIMULAÇÕES DE TRANSITÓRIOS ELETROMAGNÉTICOS EM AMBIENTE  
PHYTON**

Aluno: Caio Vinícius Colozzo Grilo  
Orientador: Prof. Dr. Afonso José do Prado

Banca Examinadora:

- Afonso José do Prado (Orientador)
- Elmer Mateus Gennaro (Examinador)
- Leonardo da Silva Lessa (Examinador)

A ata da defesa com as respectivas assinaturas dos membros encontra-se no prontuário do aluno (Expediente nº 010/2020)

Dedico esse trabalho aos meus pais, Gisele Aparecida Colozzo Grilo e Paulo Rogério Oliveira Grilo, e  
às pessoas que me auxiliaram na escalada profissional.

## **AGRADECIMENTOS**

Primeiramente, agradeço a Deus pela graça da vida e a oportunidade de realizar meus objetivos. Agradeço à minha família que sempre me apoiou nas minhas decisões, que me ajudou tanto em bons e maus momentos, especialmente, aos meus pais, Gisele Aparecida Colozzo Grilo e Paulo Rogério Oliveira Grilo, aos meus avós maternos, Irene Fabri Colozzo e João Batista Cardinal Colozzo, aos meus avós paternos, Maura Benedita Oliveira Souza e Rui Jesus Souza. Ao meu professor e orientador Afonso José do Prado, que durante minha formação contribuiu não só pelo crescimento acadêmico, mas também para o meu desenvolvimento como pessoa íntegra. Além disso, sinto-me grato, por todos que contribuíram direta e indiretamente para minha ascensão individual.

Este trabalho contou com o apoio da(s) seguinte(s) entidade(s):  
processo nº 2017/05995-1, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).  
processo nº 2019/23153-3, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

*“Se todos seguirem o roteiro, a peça será aplaudida no final.”  
(Nicollas Gabriel)*

## RESUMO

Neste trabalho, são apresentados rotinas numéricas de transitórios eletromagnéticos em linhas de transmissão. Tais rotinas podem substituir, considerando simulações simplificadas, programas do tipo *Electromagnetic Transients Program* (EMTP) para obter uma redução de custos em simulações, visto que esses programas normalmente possuem um custo elevado na obtenção de licenças. As rotinas numéricas normalmente são desenvolvidas em aplicativo matemático. Esses aplicativos podem ainda possuir custos na obtenção de licenças e é um ambiente fechado. O ambiente Python é uma linguagem de programação que está em desenvolvimento para áreas de simulações e rotinas numéricas. Por ser uma linguagem computacional simples, pode oferecer vantagem em tempos de simulação. Por ser de acesso livre, permite que colaboradores desenvolvam bibliotecas e novos recursos para sua expansão. É adequada para manipulação de matrizes na qual as rotinas são baseadas. Como desvantagem, exige que o usuário tenha conhecimento da linguagem. Contudo, será apresentado neste trabalho, uma ferramenta desenvolvida capaz de simular uma linha de transmissão apenas dispondo de seus parâmetros por unidade de comprimento aplicando as rotinas anteriormente desenvolvidas. Tal ferramenta é uma alternativa para abranger mais usuários e facilitar o uso sem que haja necessidade de conhecimentos adicionais.

**PALAVRAS-CHAVE:** Transitórios Eletromagnéticos. Rotinas Numéricas. Linhas de Transmissão. Python.

## **ABSTRACT**

In this work, numerical routines of electromagnetic transients are presented in transmission lines. Such routines can replace, considering simplified simulations, Electromagnetic Transients Program (EMTP) to achieve cost savings in simulations, these programs typically have a high cost of obtaining licenses. Numerical routines are typically developed in mathematical applications. These applications may still have license costs and are a closed environment. The Python environment is a programming language that is under development for simulation areas and numerical routines. Because it is a simple computational language, it can offer advantage in times of simulation. Because it is open access, it allows employees to develop libraries and new resources for their expansion. It is suitable for manipulating arrays on which routines are based. As a disadvantage, it requires the user to have knowledge in the language. However, it will be presented in this work, a developed tool capable of simulating a transmission line only by having its parameters per unit of length applying the previously developed routines. Such a tool is an alternative to cover more users and facilitate use without the need for additional knowledge.

**KEYWORDS:** Electromagnetic transients. Numerical Routines. Transmission lines. Python.

## LISTA DE ILUSTRAÇÕES

Figura 1	Linha representada por circuitos $\pi$ em cascata. . . . .	18
Figura 2	Representação dos parâmetros de uma linha de transmissão. . . . .	18
Figura 3	Simplificação das variáveis e aplicando análise de malha e de nó. . . . .	18
Figura 4	Aproximação da integral de uma função pela área de retângulos. . . . .	20
Figura 5	Aproximação da integral de uma função pela área de retângulos. . . . .	21
Figura 6	Fluxograma referente à rotina da matriz quadrada de ordem $2n$ . . . . .	22
Figura 7	Fluxograma da rotina da matriz de ordem 2. . . . .	23
Figura 8	Primeiro circuito $\pi$ para a aplicação de matrizes de ordem 2. . . . .	24
Figura 9	Circuito $\pi$ intermediário para a aplicação de matrizes de ordem 2. . . . .	24
Figura 10	Último circuito $\pi$ para a aplicação de matrizes de ordem 2. . . . .	24
Figura 11	Primeira unidade do modelo $\pi$ em cascata com resistência de amornecimento nos parâmetros longitudinais. . . . .	28
Figura 12	Unidades intermediárias do modelo $\pi$ em cascata com resistência de amorneci- mento nos parâmetros longitudinais. . . . .	28
Figura 13	Última unidade do modelo $\pi$ em cascata com resistência de amornecimento nos parâmetros longitudinais. . . . .	29
Figura 14	Circuitos $\pi$ em cascata com resistência de amornecimento nos parâmetros longi- tudinais. . . . .	30
Figura 15	Primeira unidade com resistência shunt. . . . .	30
Figura 16	Unidade intermediária com resistência shunt. . . . .	31
Figura 17	Última unidade com resistência shunt. . . . .	31
Figura 18	Unidade de circuito $\pi$ trabalhado. . . . .	32
Figura 19	Janela do <i>software</i> EMTP-RV e uma aplicação de unidades de circuito $\pi$ . . . . .	32
Figura 20	Modelo de linha de transmissão obtido pelo <i>software</i> EMTP-RV. . . . .	33
Figura 21	Parâmetros da linha do modelo da Figura 20. . . . .	33
Figura 22	Software desenvolvido. . . . .	34
Figura 23	Modelos de simulação disponíveis. . . . .	34
Figura 24	Resultado exibido pelo <i>software</i> . . . . .	35
Figura 25	Resultado ampliado exibido pelo <i>software</i> . . . . .	35
Figura 26	Simulação com matriz de grande ordem, tensão no terminal final da linha, $n =$ 200, $\Delta T = 50 \text{ ns}$ . . . . .	36
Figura 27	Simulação com matriz de grande ordem, detalhe da Figura 26. . . . .	37
Figura 28	Simulação com matriz de grande ordem, tensão no terminal final da linha, $n =$ 400, $\Delta T = 50 \text{ ns}$ . . . . .	37
Figura 29	Simulação com matriz de grande ordem, tensão no terminal final da linha, $n =$ 800, $\Delta T = 50 \text{ ns}$ . . . . .	38
Figura 30	Simulação matriz de grande ordem, 1000 segmentos $50 \text{ ns}$ . . . . .	38

Figura 31 Simulação com matriz de grande ordem, tensão no terminal final da linha, $n =$	
200, $\Delta T = 500 \text{ ns}$ .	39
Figura 32 Simulação com matriz de grande ordem, detalhe da Figura 31.	39
Figura 33 Simulação com matriz de grande ordem, tensão no terminal final da linha, $n =$	
200, $\Delta T = 50 \text{ ps}$ .	40
Figura 34 Simulação com matriz de grande ordem, detalhe da Figura 33.	40
Figura 35 Simulação com matriz de ordem 2, $n = 200$ , $\Delta T = 50 \text{ ns}$ , tensão no terminal	
final da linha.	41
Figura 36 Simulação com matriz de ordem 2, detalhe da Figura 35.	42
Figura 37 Simulação com matriz de ordem 2, $n = 205$ , $\Delta T = 50 \text{ ns}$ , tensão no terminal	
final da linha.	42
Figura 38 Simulação com matriz de ordem 2, detalhe da Figura 37.	43
Figura 39 Simulação com matriz de ordem 2, $n = 333$ , $\Delta T = 50 \text{ ns}$ , tensão no terminal	
final da linha.	43
Figura 40 Simulação com matriz de ordem 2, detalhe da Figura 39.	43
Figura 41 Simulação aplicando dois métodos de integração numérica, tensão no terminal	
final da linha, $n = 200$ , $\Delta T = 50 \text{ ns}$ .	44
Figura 42 Simulação aplicando dois métodos de integração numérica, detalhe da Figura 41.	45
Figura 43 Simulação aplicando dois métodos de integração numérica, tensão no terminal	
final da linha, $n = 333$ , $\Delta T = 50 \text{ ns}$ .	45
Figura 44 Simulação aplicando dois métodos de integração numérica, detalhe da Figura 43.	46
Figura 45 Simulação aplicando dois métodos de integração numérica, tensão no terminal	
final da linha, $n = 333$ , $\Delta T = 50 \text{ ps}$ .	46
Figura 46 Simulação aplicando dois métodos de integração numérica, detalhe da Figura 45.	47
Figura 47 Simulação aplicando dois métodos de integração numérica, tensão no terminal	
final da linha, $n = 400$ , $\Delta T = 50 \text{ ps}$ .	47
Figura 48 Simulação com esparsidade, tensão no terminal final da linha, $n = 200$ , $\Delta T =$	
50 ns.	48
Figura 49 Simulação com esparsidade, detalhe da Figura 48.	49
Figura 50 Simulação com esparsidade, tensão no terminal final da linha, $n = 400$ , $\Delta T =$	
50 ns.	49
Figura 51 Simulação com esparsidade, tensão no terminal final da linha, $n = 800$ , $\Delta T =$	
50 ns.	50
Figura 52 Simulação com esparsidade, tensão no terminal final da linha, $n = 1.000$ , $\Delta T =$	
50 ns.	50
Figura 53 Simulação com esparsidade, tensão no terminal final da linha, $n = 200$ , $\Delta T =$	
500 ns, em detalhe.	51
Figura 54 Simulação com esparsidade, tensão no terminal final da linha, $n = 200$ , $\Delta T =$	
500 ps, em detalhe.	51

Figura 55 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha, $n = 200$ , $\Delta T = 50 \text{ ns}$ .	53
Figura 56 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 55.	53
Figura 57 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha, $n = 400$ , $\Delta T = 50 \text{ ns}$ .	54
Figura 58 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 57.	54
Figura 59 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha, $n = 800$ , $\Delta T = 50 \text{ ns}$ .	55
Figura 60 Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 59.	55

## LISTA DE TABELAS

Tabela 1 – Comparação aplicando técnica de esparsidade usando aplicativo matemático. . . .	52
Tabela 2 – Comparação aplicando técnica de esparsidade usando Python. . . . .	52
Tabela 3 – Comparação do recurso computacional para armazenamento da matriz principal	
aplicando técnica de esparsidade. . . . .	52

## LISTA DE ABREVIATURAS E SIGLAS

TCC	Trabalho de Conclusão de Curso
UNESP	Universidade Estadual Paulista
EMTP	<i>Electromagnetic Transients Program</i>
EMTP-RV	<i>Electromagnetic Transients Program Restructured Version</i>

## LISTA DE SÍMBOLOS

$\pi$	Número PI
$n$	Número de circuitos $\pi$
$ns$	Nanosegundo
$ps$	Picossegundo
$v$	Tensão
$i$	Corrente
$R$	Resistência
$G$	Condutância
$C$	Capacitância
$L$	Indutância
$\Delta$	Delta
$\Delta t$	Passo de tempo
$R'$	Resistência por unidade de comprimento
$G'$	Condutância por unidade de comprimento
$C'$	Capacitância por unidade de comprimento
$L'$	Indutância por unidade de comprimento
$\Omega$	Ohm
$km$	Quilômetro
$H$	Henry
$F$	Farad
$S$	Siemens

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
<b>2</b>	<b>MODELAGEM DA LINHA</b>	<b>18</b>
<b>3</b>	<b>BASE MATEMÁTICA</b>	<b>20</b>
3.1	Integração de Euler	20
3.2	Integração trapezoidal	20
3.3	Matriz de grande ordem	21
3.4	Aplicação de matrizes de ordem 2	23
3.5	Aplicação de dois métodos de integração numérica	26
3.6	Matriz esparsa	27
3.7	Resistência de amortecimento nos parâmetros longitudinais	27
3.8	Resistência de amortecimento nos parâmetros transversais	30
3.9	EMTP-RV	31
<b>4</b>	<b>SOFTWARE DESENVOLVIDO</b>	<b>34</b>
4.1	Matriz de grande ordem	36
4.2	Método modificado	41
4.3	Aplicação de dois métodos de integração numérica	44
4.4	Matriz esparsa	48
4.5	Resistência de amortecimento nos parâmetros longitudinais	52
<b>5</b>	<b>CONCLUSÃO</b>	<b>56</b>
	<b>Referências</b>	<b>57</b>

## 1 INTRODUÇÃO

A análise de linhas de transmissão tem grande importância, visto que possibilita a análise de diversos parâmetros da rede, a determinação de nível de curto-circuito, modelagem de fenômenos eletromagnéticos, cálculo de fluxo de carga, dentre outros. Entretanto, quando analisada a modelagem das linhas de transmissão, encontra-se dificuldade na realização de testes em circuitos reais [1]. Com o avanço da tecnologia e avanço dos recursos computacionais, pode-se aplicar individualmente os modelos de cada elemento do sistema compondo funções matemáticas discretizadas utilizadas nos simuladores digitais [2]. Mesmo com os avanços dessas tecnologias, que fizeram boa aproximação em relação aos elementos reais, ainda há problemas ao se considerar, por exemplo, o efeito da frequência. Esse problema pode ser minimizado quando são aplicadas rotinas que utilizam matrizes de transformação como base [3]. Nesse caso, a matriz de transformação é representada por um conjunto de transformadores monofásicos ideais e as dependências com a frequência são representadas por cascatas de circuitos  $\pi$  modificados [1-5].

Quando a análise é relacionada a redes elétricas, o método mais comum para cálculos é a integração trapezoidal. Na integração trapezoidal, a aproximação se dá por trapézios infinitesimais, o que não ocorre no método de Euler, em que a área embaixo da curva é aproximada por retângulos infinitesimais. Essa mudança de aproximação não compromete significativamente o tempo de simulação e seu resultado é muito superior em termos de precisão dos resultados obtidos. Ao analisar os *softwares* do tipo EMTP, nota-se que eles também fazem uso da integração trapezoidal, considerando o tempo como variável independente [6, 7]. Esses *softwares* são bastante difundidos, sendo padrões para diversas análises de fenômenos em sistema de potência. Pelo fato de serem altamente difundidos, eles são frequentemente atualizados [8]. Por outro lado, é possível a implementação por meio da integração trapezoidal para análise simplificada de redes elétricas em qualquer *software* que trabalhe com cálculo matricial [9, 10].

Como mencionado, a grande dificuldade é a correta representação da linha de transmissão, visto que em geral as linhas de transmissão não podem ser representadas diretamente no domínio do tempo, pois elas são dependentes da frequência em seus parâmetros longitudinais.

Utilizando matrizes de grande ordem, o modelo abordado no início do projeto era representado por apenas uma matriz. Essa matriz descrevia toda linha de transmissão em variáveis de estado. A matriz de grande ordem é um dos modelos mais simples e difundido para representar circuitos  $\pi$  em cascata. Por sua vez, ela não é a mais eficaz. Na matriz de grande ordem, são encontrados problemas como: número elevado de elementos e os elementos não nulos só estão dispostos em sua diagonal principal, diagonais superior e inferior à diagonal principal. Dessa forma, os elementos nulos prejudicam o desempenho em termos de simulação, precisão e tempo computacional para a simulação.

Como citado, os valores que descrevem os circuitos  $\pi$  em cascata estão somente em torno da diagonal principal. A utilização do método pode trazer desvantagens em alguns modelos de simuladores tradicionais. Já no ambiente Python, o problema pode ser modelado por meio da programação de rotinas numéricas que descartam valores nulos. Esta abordagem pode ser denominada abordagem termo

a termo e possibilita redução computacional no tempo de simulação [20, 21]. Devido a simplicidade na representação numérica e aplicação em diversos *softwares* e aplicativos matemáticos, a rotina ainda é utilizada com fins de uma simulação rápida da linha.

O problema pode ser abordado de uma maneira mais específica: uma opção é reduzir a ordem da matriz e aumentar o número de matrizes. A redução da ordem da matriz promove uma diminuição de elementos nulos. Por sua vez, nesse método, há também uma redução do efeito de Gibbs, diminuindo erros numéricos. Quando comparada essa rotina com a rotina baseada em matriz de grande ordem, o tempo de simulação aumenta devido a inúmeros "loops" gerados por diminuição da ordem da matriz. Isso gera aumento na quantidade de matrizes utilizadas para descrever a cascata de circuitos  $\pi$ .

Outra alternativa é a utilização da técnica de esparsidade. Como mencionado, os elementos não nulos na matriz principal estão somente na diagonal principal e nas subdiagonais superior e inferior [22, 23]. A abordagem da técnica de esparsidade, então, permite a redução de memória, visto que elementos nulos não serão armazenados, bem como a redução no tempo de processamento, pois operações matemáticas matriciais também são simplificadas.

As oscilações numéricas encontradas no método de grande ordem podem ser amortecidas pela inserção de resistências de amortecimento ao longo da linha em seus parâmetros longitudinais. Contudo, além da utilização de resistências em paralelo aos parâmetros longitudinais, pode-se utilizar resistências de amortecimento nos parâmetros transversais da rede (resistências *shunts*).

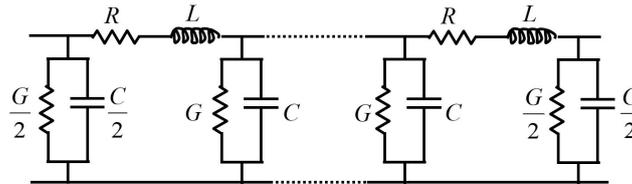
Já com auxílio do *software* EMTP-RV, é possível aplicar o circuito modelado para simulação pelos próprios métodos fornecidos pelo *software*. Por sua vez, para uma análise comparativa entre diferentes técnicas, pode-se aplicar uma linha de transmissão fornecida pelo EMTP [15], introduzindo as variáveis que definem a linha em aplicativos matemáticos ou ambientes de programação. Como resultado, circuitos similares podem ser simulados em diferentes plataformas e diversas comparações podem ser realizadas.

## 2 MODELAGEM DA LINHA

Para a realização do trabalho, o modelo utilizado foi uma cascata de circuitos  $\pi$ . Sua representação é mostrada na [Figura 1](#).

Em cada circuito  $\pi$  da cascata, cada saída está ligada a entrada do circuito seguinte, tornando o modelo viável para representação simplificada de uma linha de transmissão ([Figura 1](#)).

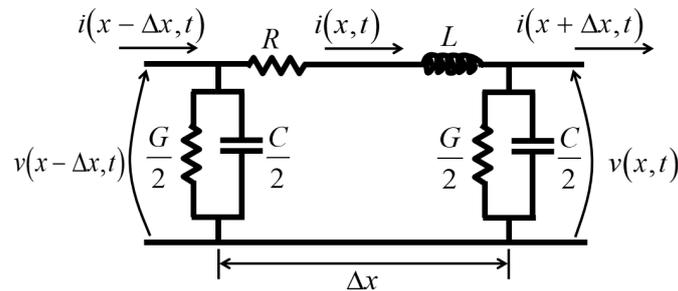
Figura 1 – Linha representada por circuitos  $\pi$  em cascata.



fonte: Produção do próprio autor.

A representação mostrada na [Figura 1](#) é baseada em unidades infinitesimais ( $\Delta x \rightarrow 0$ ) aplicadas para considerar a característica distribuída dos parâmetros de linhas de transmissão ([Figura 2](#)).

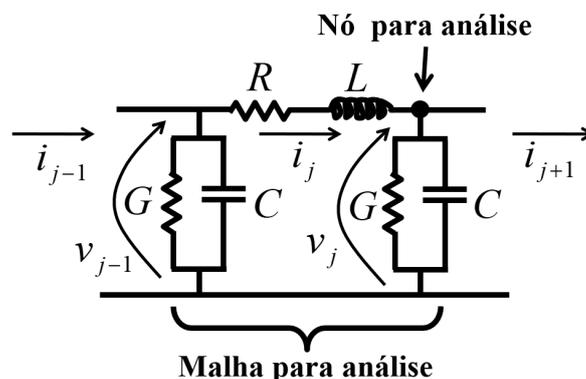
Figura 2 – Representação dos parâmetros de uma linha de transmissão.



fonte: Produção do próprio autor.

Simplificando a representação das variáveis e aplicando análise de malha e de nó, é possível obter equações de estado da unidade infinitesimal ([Figura 3](#)).

Figura 3 – Simplificação das variáveis e aplicando análise de malha e de nó.



fonte: Produção do próprio autor.

O parâmetro R representa a resistência da linha e pode ser calculado por (2.1).

$$R = R' \frac{d}{n} \quad (2.1)$$

Já o parâmetro L, representa a indutância da linha e pode ser calculado por (2.2).

$$L = L' \frac{d}{n} \quad (2.2)$$

De modo análogo, C e G representam respectivamente a capacitância e condutância da linha e podem ser calculados por (2.3) e (2.4).

$$C = C' \frac{d}{n} \quad (2.3)$$

$$G = G' \frac{d}{n} \quad (2.4)$$

Na malha da Figura 3, tem-se:

$$v_{j-1} - Ri_j - L \frac{di}{dt} - v_j = 0 \quad (2.5)$$

Representando a derivada da corrente em função do tempo por  $\dot{i}_j$ , tem-se:

$$\dot{i}_j = \frac{1}{L}v_{j-1} - \frac{R}{L}i_j - \frac{1}{L}v_j \quad (2.6)$$

No nó da Figura 3, tem-se:

$$i_j - C \frac{d\dot{v}}{dt} - Gv_j - i_{j+1} = 0 \quad (2.7)$$

A derivada da tensão em função do tempo é simplificada por  $\dot{v}_j$ . Dessa forma, tem-se:

$$\dot{v}_j = \frac{1}{C}i_j - \frac{G}{C}v_j - \frac{1}{C}i_{j+1} \quad (2.8)$$

### 3 BASE MATEMÁTICA

A base matemática para simulação de propagação de transitórios em linhas de transmissão é composta pela aplicação de integração numérica na solução de sistemas lineares.

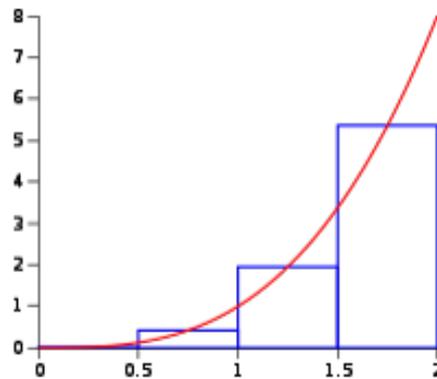
Para isso, dentre os métodos de integração numérica, destacam-se dois:

- Integração de Euler;
- Integração Trapezoidal.

#### 3.1 INTEGRAÇÃO DE EULER

O método mais comum de integração numérica é o método de Euler. Tal método visa à aproximação da integral sob uma curva por retângulos infinitesimais (Figura 4). A principal vantagem do método de Euler é a simplicidade de aplicação.

Figura 4 – Aproximação da integral de uma função pela área de retângulos.



fonte: Produção do próprio autor.

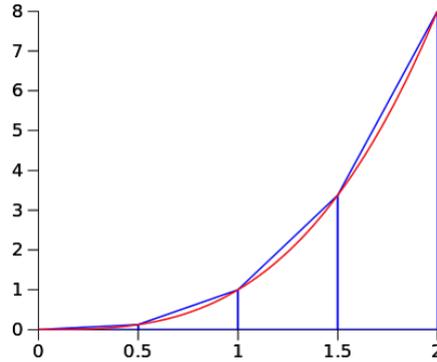
#### 3.2 INTEGRAÇÃO TRAPEZOIDAL

A integração trapezoidal é uma das ferramentas numéricas mais utilizadas nos *softwares* EMTP e rotinas para simulação de transitórios eletromagnéticos em sistemas de potência, cujo resultado é obtido por aproximação por meio de trapézios infinitesimais. Devido a utilização de trapézios infinitesimais (Figura 5), para o mesmo intervalo de tempo da integração de Euler, a integração trapezoidal se mostra mais precisa. Seu custo benefício é melhor em relação as rotinas mais simples e métodos mais complexos que, apesar de ter uma maior precisão, exigem um tempo de processamento muito elevado.

Em (3.1), é demonstrada a fórmula da equação trapezoidal, em que  $\Delta t$  representa o intervalo de tempo, no caso, a largura da base de cada trapézio e  $k$  representa o índice da posição analisada. Dessa forma, quanto menor a base desse trapézio, mais precisa será a aproximação.

$$X_{k+1} = X_k + \frac{\Delta t}{2} [\dot{X}_{k+1} + \dot{X}_k] \quad (3.1)$$

Figura 5 – Aproximação da integral de uma função pela área de retângulos.



fonte: Produção do próprio autor.

### 3.3 MATRIZ DE GRANDE ORDEM

Para simulação de transitórios eletromagnéticos, inicialmente, foi utilizada a manipulação de uma matriz de grande ordem. O tamanho da matriz é determinado pela quantidade de circuitos  $\pi$  em cascata empregados para representar a linha de transmissão (Figura 1). No caso, o tamanho da matriz quadrada é  $2n$ , em que  $n$  representa o número de circuitos  $\pi$  em cascata. Cada circuito  $\pi$  está relacionado à duas variáveis de estado.

Por exemplo, ao se utilizar uma matriz de 200 circuitos  $\pi$  em cascata, será necessária uma matriz quadrada de ordem 400.

A abordagem da manipulação matricial e da rotina numérica podem ser esquematizadas por meio do algoritmo da Figura 6. O modelo matemático simplificado pode ser expresso por (3.2).

Note que a estratégia de solução representada pelo fluxograma da Figura 6 consiste em um único "loop" de interação por cada instante de tempo.

O método de análise consiste em um sistema analítico, que pode ser expresso por (3.2).

$$\dot{x} = Ax + Bu \quad (3.2)$$

O vetor  $B$  é expresso por (3.3).

$$B = \left[ \frac{1}{L} \quad 0 \quad \dots \quad 0 \right]^T \quad (3.3)$$

Já o vetor  $x$  é expresso por (3.4).

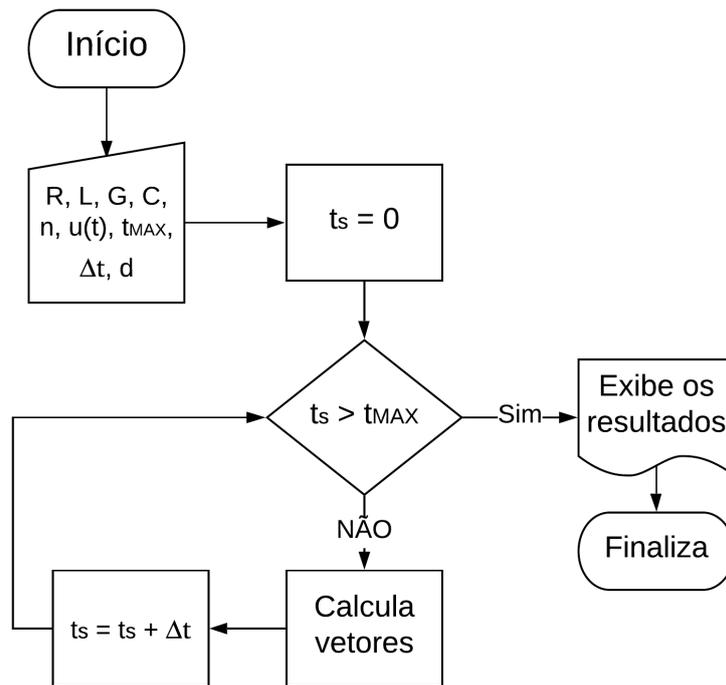
$$x = \left[ i_1 \quad v_1 \quad i_2 \quad v_2 \quad \dots \quad i_n \quad v_n \right]^T \quad (3.4)$$

O vetor  $u$  é expresso por (3.5).

$$u = \left[ u_{V1} \quad i_{C1} \quad u_{V2} \quad i_{C2} \quad \dots \quad u_{Vn} \quad i_{Cn} \right]^T \quad (3.5)$$

Os vetores  $x$  e  $B$  representam as variáveis de estado do sistema analisado e as variáveis relacionadas às entradas do circuito, respectivamente.

Figura 6 – Fluxograma referente à rotina da matriz quadrada de ordem  $2n$ .



fonte: Produção do próprio autor.

A matriz  $A$  é a matriz de ordem  $2n \times 2n$ , em que  $n$  é o número de circuitos  $\pi$  em cascata utilizados. A matriz  $A$  representa todos os elementos do circuito em cada nó de conexão, que são obtidos por meio das equações 2.5 à 2.8, e pode ser expressa por:

$$A = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} & 0 & \dots & \dots & \dots & 0 \\ \frac{1}{C} & -\frac{G}{C} & -\frac{1}{C} & 0 & \dots & \ddots & \vdots \\ 0 & -\frac{1}{L} & -\frac{R}{L} & -\frac{1}{L} & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0 & \frac{1}{C} & -\frac{G}{C} & \frac{1}{C} & 0 \\ \vdots & \ddots & \ddots & 0 & \frac{1}{L} & -\frac{R}{L} & -\frac{1}{L} \\ 0 & \dots & \dots & \dots & 0 & \frac{2}{C} & -\frac{G}{C} \end{bmatrix} \quad (3.6)$$

### 3.4 APLICAÇÃO DE MATRIZES DE ORDEM 2

Devido aos erros impostos pelo fenômeno de Gibbs, o modelo de grande ordem foi dividido em três partes. Dessa forma, foram analisados circuitos de maneira individualizada, com  $n$  matrizes quadradas de ordem 2.

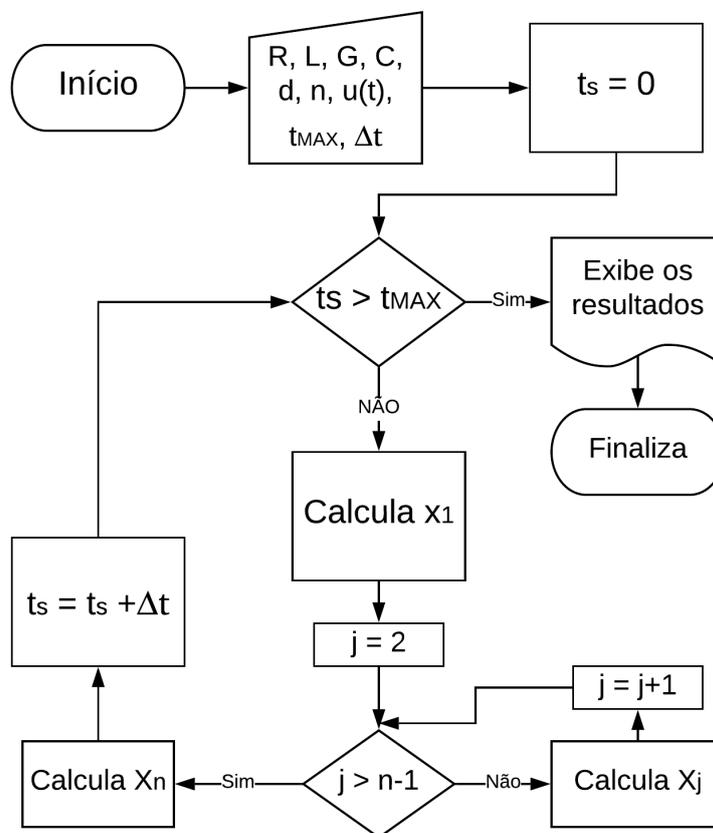
As três partes da cascata podem ser definidas como:

- Circuito inicial;
- Circuitos intermediários;
- Circuito final.

A rotina baseada nessa modelagem, pode ser expressa pelo fluxograma da [Figura 7](#). Nele, não há um único "loop", mas sim, diversos laços que serão utilizados para calcular o valor de cada matriz quadrada de ordem 2.

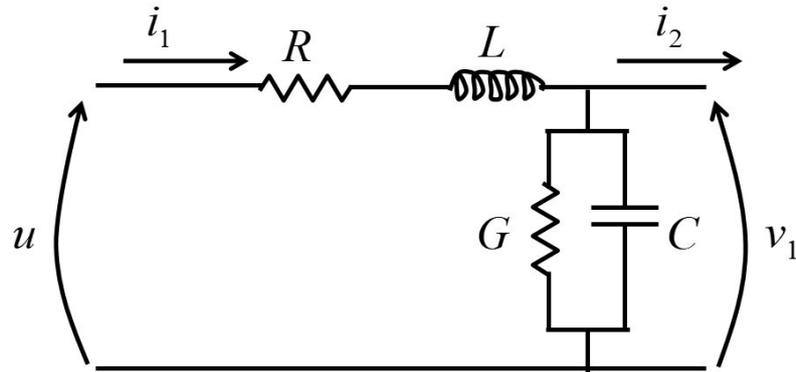
Nessa modelagem, o primeiro circuito é destinado à entrada. Ele pode ser representado pela [Figura 8](#). Os circuitos intermediários são todos os circuitos  $\pi$  intermediários, representando toda linha de transmissão até a última unidade. Eles podem ser representados pela [Figura 9](#). Já o último circuito é o que finaliza a cascata e pode ser representado pela [Figura 10](#).

Figura 7 – Fluxograma da rotina da matriz de ordem 2.



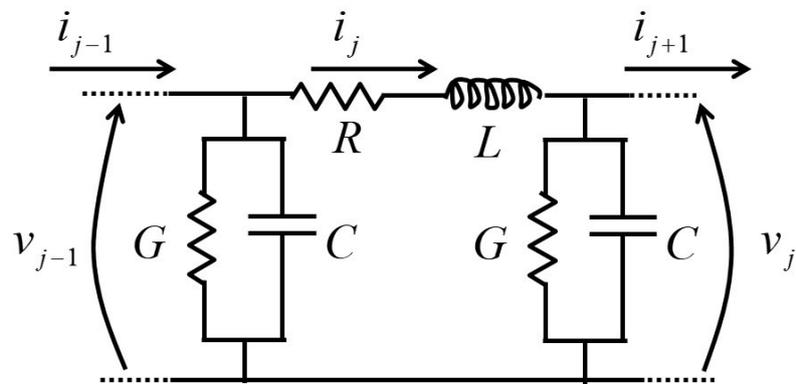
fonte: Produção do próprio autor.

Figura 8 – Primeiro circuito  $\pi$  para a aplicação de matrizes de ordem 2.



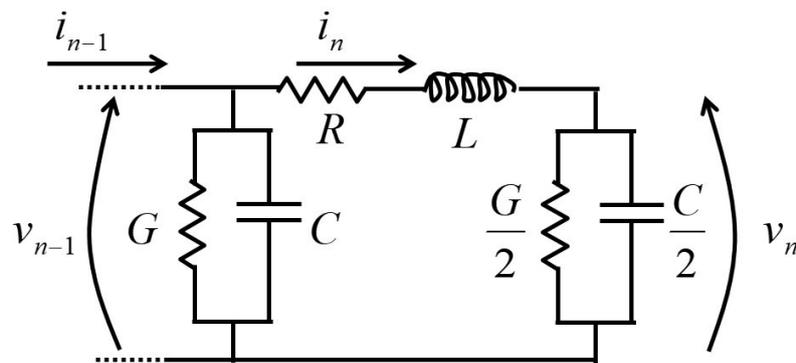
fonte: Produção do próprio autor.

Figura 9 – Circuito  $\pi$  intermediário para a aplicação de matrizes de ordem 2.



fonte: Produção do próprio autor.

Figura 10 – Último circuito  $\pi$  para a aplicação de matrizes de ordem 2.



fonte: Produção do próprio autor.

Para análise desses circuitos, são utilizadas as técnicas de análise de malha e análise nodal. Para a primeira unidade, tem-se a equação matricial representada por (3.7):

$$\begin{bmatrix} i_1(K+1) \\ v_1(K+1) \end{bmatrix} = A_{1A}^{-1} \cdot A_{1B} \cdot \begin{bmatrix} i_1(K) \\ v_1(K) \end{bmatrix} + A_{1A}^{-1} \cdot B_{1A} + A_{1A}^{-1} \cdot C_{1A} \cdot \begin{bmatrix} i_2(K) \\ v_2(K) \end{bmatrix} \quad (3.7)$$

A matriz  $A_{1A}$  é dada por (3.8):

$$A_{1A} = \begin{bmatrix} 1 + \frac{R\Delta t}{2L} & \frac{\Delta t}{2L} \\ -\frac{\Delta t}{2C} & 1 + \frac{G\Delta t}{2C} \end{bmatrix} \quad (3.8)$$

Já matriz  $A_{1B}$  pode ser representada por (3.9):

$$A_{1B} = \begin{bmatrix} 1 - \frac{R\Delta t}{2L} & -\frac{\Delta t}{2L} \\ \frac{\Delta t}{2C} & 1 - \frac{G\Delta t}{2C} - \frac{\Delta t^2}{2LC} \end{bmatrix} \quad (3.9)$$

A matriz  $B_{1A}$  é dada por (3.10):

$$B_{1A} = \begin{bmatrix} \frac{\Delta t}{2L} \\ 0 \end{bmatrix} \cdot \left[ u(K+1) + u(k) \right] \quad (3.10)$$

E a matriz  $C_{1A}$  pode ser dada por (3.11):

$$C_{1A} = \begin{bmatrix} 0 & 0 \\ \frac{\Delta t}{2C} \left[ \frac{R\Delta t}{L} - 2 \right] & \frac{\Delta t^2}{2LC} \end{bmatrix} \quad (3.11)$$

Fazendo a análise usando as mesmas técnicas para o circuito intermediário, obtém-se (3.12):

$$\begin{bmatrix} i_j(K+1) \\ v_j(K+1) \end{bmatrix} = A_{jA}^{-1} \cdot A_{jB} \cdot \begin{bmatrix} i_j(K) \\ v_j(K) \end{bmatrix} + A_{jA}^{-1} \cdot B_{jA} + A_{jA}^{-1} \cdot C_{jA} \cdot \begin{bmatrix} i_{j+1}(K) \\ v_{j+1}(K) \end{bmatrix} \quad (3.12)$$

Em que  $j$  corresponde ao circuito intermediário na cascata.

A matriz  $A_{jA}$  é dada por (3.13):

$$A_{jA} = \begin{bmatrix} 1 + \frac{R\Delta t}{2L} - \frac{\Delta t^2}{2LC} & \frac{\Delta t}{2L} \\ -\frac{\Delta t}{2C} & 1 + \frac{G\Delta t}{2C} \end{bmatrix} \quad (3.13)$$

Já matriz  $A_{jB}$  é representada por (3.14):

$$A_{jB} = \begin{bmatrix} 1 - \frac{R\Delta t}{2L} & -\frac{\Delta t}{2L} \\ \frac{\Delta t}{2C} & 1 - \frac{G\Delta t}{2C} - \frac{\Delta t^2}{2LC} \end{bmatrix} \quad (3.14)$$

A matriz  $B_{jA}$  apresenta-se como (3.15):

$$B_{jA} = \begin{bmatrix} -\frac{\Delta t^2}{2LC} & \frac{\Delta t}{2L} \left[ \frac{G\Delta t}{C} + 2 \right] \\ 0 & 0 \end{bmatrix} \quad (3.15)$$

E a matriz  $C_{jA}$  consiste em (3.16):

$$C_{jA} = \begin{bmatrix} 0 & 0 \\ \frac{\Delta t}{2C} \left[ \frac{R\Delta t}{L} - 2 \right] & \frac{\Delta t^2}{2LC} \end{bmatrix} \quad (3.16)$$

Por fim, o último circuito  $\pi$  é descrito por (3.17).

$$\begin{bmatrix} i_n(K+1) \\ v_n(K+1) \end{bmatrix} = A_{nA}^{-1} \cdot A_{nB} \cdot \begin{bmatrix} i_n(K) \\ v_n(K) \end{bmatrix} + A_{nA}^{-1} \cdot B_{nA} \cdot \begin{bmatrix} i_{n-1}(K+1) \\ v_{n-1}(K+1) \end{bmatrix} \quad (3.17)$$

Em que  $n$ , corresponde a posição do último circuito  $\pi$  na cascata.

A matriz  $A_{nA}$  é mostrada em (3.18):

$$A_{nA} = \begin{bmatrix} 1 + \frac{R\Delta t}{2L} - \frac{\Delta t^2}{2LC} & \frac{\Delta t}{2L} \\ -\frac{\Delta t}{C} & 1 + \frac{G\Delta t}{2C} \end{bmatrix} \quad (3.18)$$

Já matriz  $A_{nB}$  é dada por (3.19):

$$A_{nB} = \begin{bmatrix} 1 - \frac{R\Delta t}{2L} & -\frac{\Delta t}{2L} \\ \frac{\Delta t}{C} & 1 - \frac{G\Delta t}{2C} \end{bmatrix} \quad (3.19)$$

Por fim, a matriz  $B_{nA}$  é dada por (3.20):

$$B_{nA} = \begin{bmatrix} -\frac{\Delta t^2}{2LC} & \frac{\Delta t}{2L} \left[ \frac{G\Delta t}{C} + 2 \right] \\ 0 & 0 \end{bmatrix} \quad (3.20)$$

### 3.5 APLICAÇÃO DE DOIS MÉTODOS DE INTEGRAÇÃO NUMÉRICA

Uma nova proposta de trabalho foi aproximar o desempenho de simulação à rotina da matriz de grande ordem e manter a precisão obtida no método de aplicação de matrizes de ordem 2. Para isso, foi proposta a utilização de manipulação termo a termo, adotando parâmetros de tensão e corrente como variáveis independentes.

Na nova rotina, as aproximações são dadas por meio de dois tipos de integração: integração de base (integração trapezoidal) e uma integração de ordem inferior (integração de Euler). Nesse modo, a utilização da integração trapezoidal se torna obrigatória, visto que a manipulação da rotina exige que se tenha outra de ordem inferior para realização dos cálculos aproximados, caso contrário, chegaria em um sistema sem solução.

Nesse método, a cascata de circuitos  $\pi$  também é dividida em unidades, sendo a mesma divisão da rotina anterior.

A primeira unidade continua sendo considerada como o início da linha e o local onde estão situadas todas as fontes da cascata. Suas equações de tensão e corrente são descritas por (3.21) e (3.22) respectivamente.

$$\begin{aligned} \left[ 1 + \frac{G\Delta t}{2C} - \frac{\Delta t^2}{2LC} \right] v_1(k+1) &= \left[ 1 - \frac{G\Delta t}{2C} - \frac{\Delta t^2}{2LC} \right] v_1(k) + \frac{\Delta t}{2C} \left[ \frac{R\Delta t}{L} + 2 \right] i_1(k+1) + \\ &\quad - \frac{\Delta t^2}{2LC} u(k+1) + \frac{\Delta t}{2C} \left[ \frac{R\Delta t}{L} - 2 \right] i_2(k) + \frac{\Delta t^2}{2LC} v_2(k) \end{aligned} \quad (3.21)$$

$$i_1(k) = \left[ \frac{R\Delta t}{L} + 1 \right] i_1(k+1) - \frac{\Delta t}{L} u(k+1) + \frac{\Delta t}{L} v_1(k) \quad (3.22)$$

As unidades intermediárias são as que se estendem por toda linha e podem ser representadas por (3.23) e (3.24).

$$\left[ 1 + \frac{G\Delta t}{2C} - \frac{\Delta t^2}{LC} \right] v_n(k+1) = \left[ 1 - \frac{G\Delta t}{2C} \right] v_n(k) + \frac{\Delta t}{C} \left[ \frac{R\Delta t}{L} + 2 \right] i_n(k+1) + \frac{\Delta t^2}{LC} v_{n-1}(k+1) \quad (3.23)$$

$$i_n(k) = i_n(k+1) - \frac{\Delta t}{L} v_{n-1}(k+1) + \frac{R\Delta t}{L} i_n(k+1) + \frac{\Delta t}{L} v_n(k+1) \quad (3.24)$$

Por fim, a última unidade representa o término da linha. No caso, esta é colocada em aberto para as simulações. Sua equação pode ser descrita por (3.25) e (3.26).

$$\left[ 1 + \frac{G\Delta t}{2C} - \frac{\Delta t^2}{LC} \right] v_n(k+1) = \left[ 1 - \frac{G\Delta t}{2C} \right] v_n(k) + \frac{\Delta t}{C} \left[ \frac{R\Delta t}{L} + 2 \right] i_n(k+1) - \frac{\Delta t}{LC} v_{-1n}(k+1) \quad (3.25)$$

$$\left[ 1 + \frac{R\Delta t}{2L} - \frac{\Delta t^2}{2LC} \right] i_n(k+1) = \left[ 1 - \frac{R\Delta t}{2L} - \frac{\Delta t^2}{2LC} \right] i_1(k) + \frac{\Delta t}{2L} \left[ \frac{G\Delta t}{C} + 2 \right] v_{n-1}(k+1) - \frac{\Delta t}{2L} \left[ 2 - \frac{G\Delta t}{C} \right] v_n(k) - \frac{\Delta t^2}{2LC} i_{n-1}(k+1) \quad (3.26)$$

### 3.6 MATRIZ ESPARSA

A rotina da matriz esparsa é baseada no modelo de grande ordem. Em (3.6), a matriz A possui elementos não nulos somente em sua diagonal principal e nas subdiagonais superiores e inferiores à diagonal principal. A manipulação de valores nulos causa um grande fluxo computacional, prejudicando o desempenho da rotina em termos de tempo de processamento, já que o aumento de números de circuitos aumenta em quatro vezes o número de elementos.

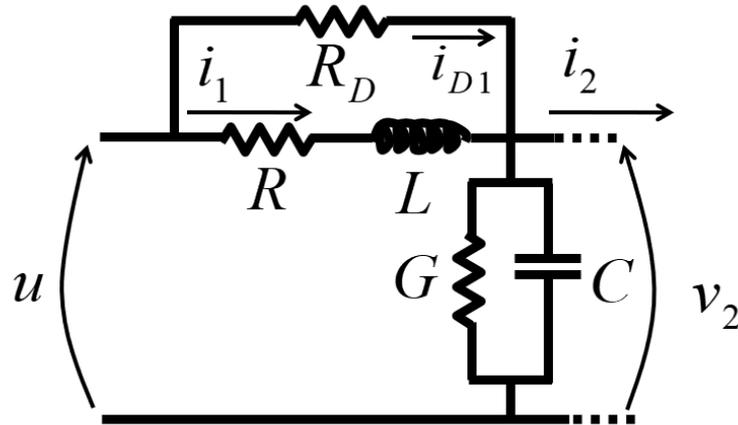
A técnica de esparsidade não afeta o resultado do cálculo matricial. A matriz é salva em uma lista com posições definidas de cada elemento. A multiplicação entre matrizes acaba sendo por posição e ao entrar um termo de uma posição que não existe na lista, o mesmo será nulo. Desse modo, a utilização da esparsidade torna-se mais viável em rotinas com um grande número de elementos nulos. De certa maneira, em uma matriz pequena, por exemplo de ordem 400, o ganho obtido se torna relativamente baixo. Contudo, em matrizes de ordem maior, esse ganho é relativamente alto, capaz de reduzir drasticamente o tempo de processamento.

### 3.7 RESISTÊNCIA DE AMORTECIMENTO NOS PARÂMETROS LONGITUDINAIS

Em maior ou menor grau, a aplicação de métodos de integração numérica gera erros numéricos, que em sua maioria podem ser associados a oscilações numéricas ou oscilações de Gibbs. No caso das simulações de transitórios em linhas de transmissão realizadas, uma alternativa para eliminar as

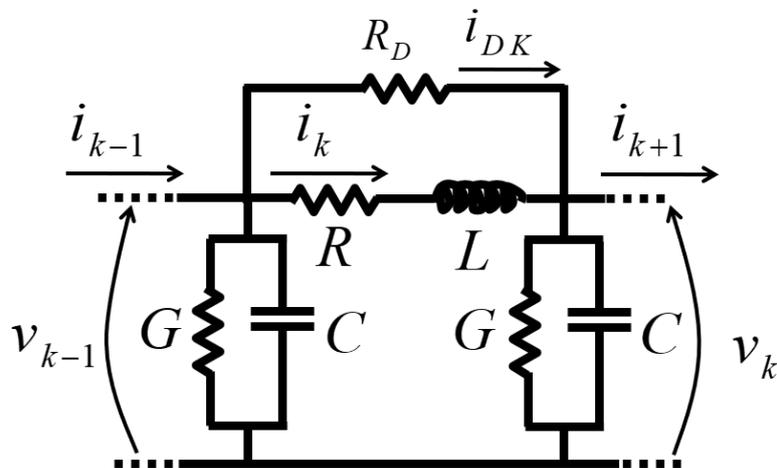
oscilações numéricas foi aplicar o uso de resistências de amortecimento. Na [Figura 11](#) à [Figura 13](#), foi colocada uma resistência  $R_d$  em paralelo com os parâmetros longitudinais para amortecimento de fenômenos relacionados às frequências das oscilações numéricas. O resultado obtido é similar ao esperado pelo método de Gear e pelos programas EMTP [\[22, 23\]](#).

Figura 11 – Primeira unidade do modelo  $\pi$  em cascata com resistência de amortecimento nos parâmetros longitudinais.



fonte: Produção do próprio autor.

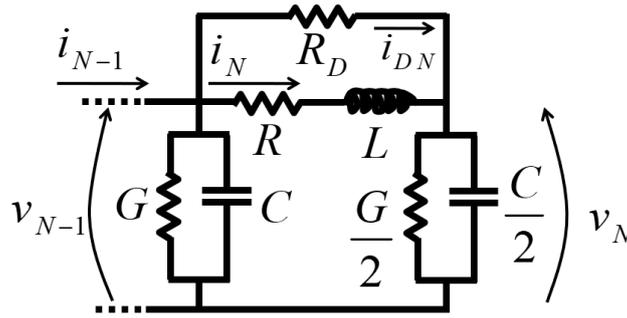
Figura 12 – Unidades intermediárias do modelo  $\pi$  em cascata com resistência de amortecimento nos parâmetros longitudinais.



fonte: Produção do próprio autor.

O equacionamento para tensão e corrente da primeira unidade ([Figura 11](#)), é mostrado por [\(3.27\)](#) e

Figura 13 – Última unidade do modelo  $\pi$  em cascata com resistência de amornecimento nos parâmetros longitudinais.



fonte: Produção do próprio autor.

(3.28).

$$\begin{aligned}
 \left[ 1 + \frac{G\Delta t}{2C} + \frac{G_D\Delta t}{C} \right] v_1(k+1) &= \left[ 1 - \frac{\Delta t^2}{2LC} + \frac{G_D^2\Delta t^2}{2C^2} - \frac{G\Delta t}{2C} - \frac{G_D\Delta t}{C} \right] v_1(k) + \\
 &+ \frac{\Delta t}{2C} i_1(k+1) + \frac{\Delta t}{2C} i_1(k) + \frac{G_D\Delta t}{2C} u(k+1) + \frac{G_D\Delta t}{2C} u(k) + \\
 &+ \frac{\Delta t}{2C} \left[ G_D \left[ 2 - \frac{(G+2G_D)\Delta t}{C} + \frac{\Delta t}{L} \right] v_2(k) + \right. \\
 &\quad \left. + \frac{\Delta t}{2C} \left[ \frac{R\Delta t}{L} + \frac{G_D\Delta t}{C} - 2 \right] i_2(k) + \right. \\
 &\quad \left. + \frac{G_D^2\Delta t^2}{2C} v_3(k) - \frac{G_D\Delta t^2}{2C^2} i_3(k) \right] \quad (3.27)
 \end{aligned}$$

$$\begin{aligned}
 \left[ 1 + \frac{R\Delta t}{2L} \right] i_1(k+1) &= \left[ 1 - \frac{R\Delta t}{2L} \right] i_1(k) - \frac{2\Delta t}{2L} v_1(k+1) - \frac{\Delta t}{2L} v_1(k) + \\
 &\quad + \frac{\Delta t}{2L} u(k+1) + \frac{\Delta t}{2L} u(k) \quad (3.28)
 \end{aligned}$$

Já o equacionamento para tensão e corrente das unidades intermediárias (Figura 12) pode ser expresso por (3.29) e (3.30).

$$v_n(k+1) = \left[ 1 - \frac{2G_D\Delta t}{C} - \frac{G\Delta t}{C} \right] v_n(k) + \frac{2\Delta t}{C} i_n(k) + \frac{2G_D\Delta t}{C} v_{n-1}(k) \quad (3.29)$$

$$i_n(k+1) = i_n(k) + \frac{\Delta t}{L} v_{n-1}(k) - \frac{R\Delta t}{L} i_n(k) - \frac{\Delta t}{L} v_n(k) \quad (3.30)$$

Por fim, o equacionamento para tensão e corrente para a última unidade (Figura 13) pode ser expresso por (3.31) e (3.32).

$$\begin{aligned}
 v_n(k+1) &= v_n(k) + \\
 + \frac{\Delta t}{C} \left[ i_n(k+1) + G_D v_{n-1}(k+1) - \left[ G_D + \frac{G}{2} \right] v_n(k+1) + i_n(k) + G_D v_{n-1}(k) - \left[ G_D + \frac{G}{2} \right] v_n(k) \right] \quad (3.31)
 \end{aligned}$$

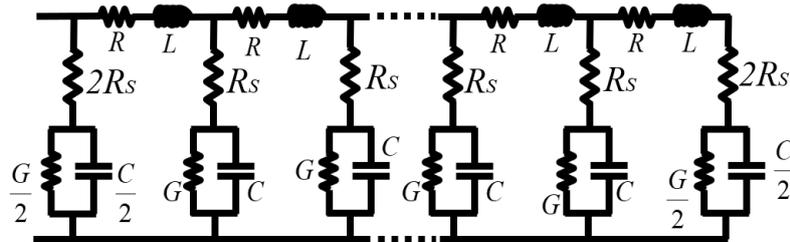
$$\left[ \frac{1}{2} + \frac{R\Delta t}{2L} \right] i_n(k+1) = \frac{1}{2} i_n(k) + \frac{\Delta t}{2L} v_{n-1}(k+1) - \frac{\Delta t}{2L} v_n(k+1) \quad (3.32)$$

### 3.8 RESISTÊNCIA DE AMORTECIMENTO NOS PARÂMETROS TRANSVERSAIS

As oscilações numéricas encontradas no método de grande ordem foram amortecidas pela inserção de resistências de amortecimento ao longo da linha em seus parâmetros longitudinais. Uma alternativa, além da utilização de resistências em paralelo aos parâmetros longitudinais, seria a implementação de resistências de amortecimento nos parâmetros transversais da rede.

A estrutura de uma linha de transmissão com essas resistências ficaria conforme a [Figura 14](#).

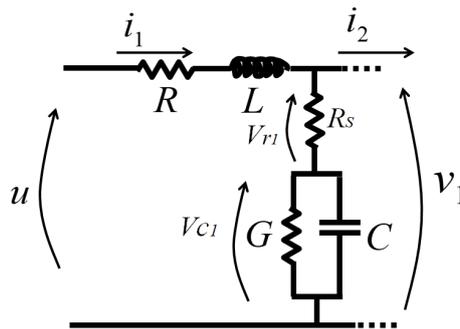
Figura 14 – Circuitos  $\pi$  em cascata com resistência de amortecimento nos parâmetros longitudinais.



fonte: Produção do próprio autor.

Na [Figura 15](#), é mostrado o exemplo da estrutura da primeira unidade do circuito.

Figura 15 – Primeira unidade com resistência shunt.



fonte: Produção do próprio autor.

Há alterações, considerando circuitos  $\pi$  não modificados, apenas nas relações diferenciais das tensões. Para a primeira unidade, tem-se:

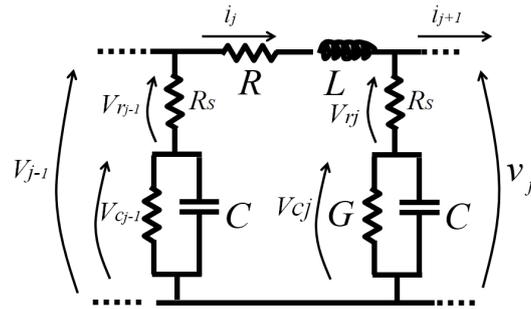
$$\begin{aligned} \dot{v}_1 = \frac{1}{C} \left[ 1 + R_s \left( G - \frac{RC}{L} \right) \right] i_1 - \frac{1}{C} \left[ G + \left( \frac{2CR_s}{L} \right) \right] v_1 + \\ - \frac{1}{C} \left[ 1 + R_s \left( G - \frac{RC}{L} \right) \right] i_2 + \frac{R_s}{L} u(t) + \frac{R_s}{L} v_2 \end{aligned} \quad (3.33)$$

Já na [Figura 16](#), é mostrado o exemplo da estrutura das unidade intermediárias do circuito.

Para as tensões das unidades intermediárias, tem-se:

$$\begin{aligned} \dot{v}_j = \frac{1}{C} \left[ 1 + R_s \left( G - \frac{RC}{L} \right) \right] i_j - \frac{1}{C} \left[ G + \left( \frac{2CR_s}{L} \right) \right] v_j + \\ - \frac{1}{C} \left[ 1 + R_s \left( G - \frac{RC}{L} \right) \right] i_{j+1} + \frac{R_s}{L} v_{j+1} + \frac{R_s}{L} v_{j-1} \end{aligned} \quad (3.34)$$

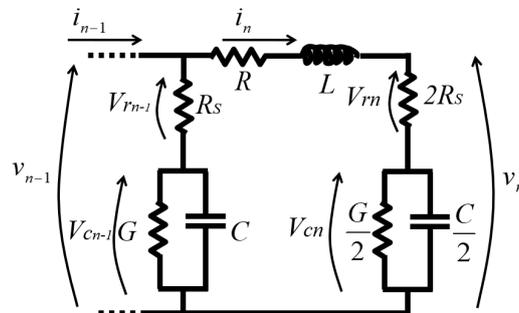
Figura 16 – Unidade intermediária com resistência shunt.



fonte: Produção do próprio autor.

Logo, na [Figura 17](#) é mostrado o exemplo da última estrutura do circuito.

Figura 17 – Última unidade com resistência shunt.



fonte: Produção do próprio autor.

Para a última unidade, tem-se:

$$v_n = \frac{2}{C} \left[ 1 - R_s \left( G - \frac{RC}{L} \right) \right] i_n - \frac{2}{C} \left( \frac{G}{2} + \frac{CR_s}{L} \right) v_n + \frac{2R_s}{L} v_{n-1} \quad (3.35)$$

### 3.9 EMTP-RV

Foi disponibilizada uma licença do *software* EMTP-RV para realização do trabalho durante o período de bolsa de pesquisa e estágio no exterior (BEPE/FAPESP, processo 2019/23153-3). Tal *software* é específico para simulação de transitórios eletromagnéticos. Com ele é possível aplicar o método desenvolvido numericamente para validação.

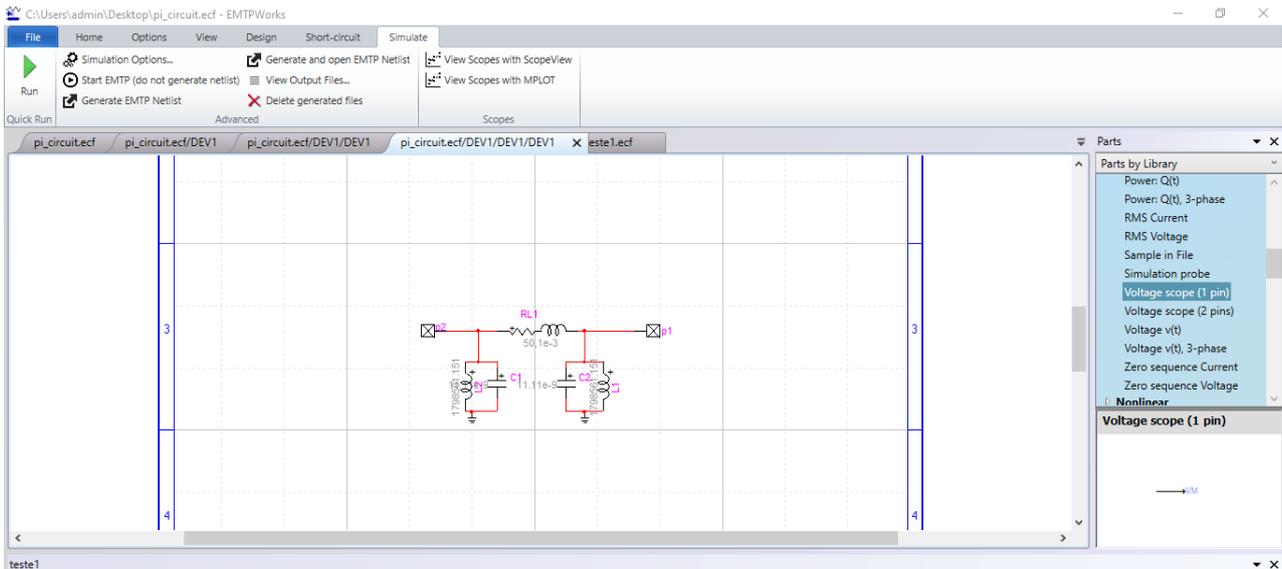
Para validação do método foram executadas duas etapas: constituição de inúmeros circuitos  $\pi$  em cascata para comprovar os resultados da rotina existente, bem como, a aplicação de um modelo de linha de transmissão direto no *software* EMTP-RV. Dessa maneira, foi analisada principalmente a solução numérica obtida para desenvolvimento da rotina em ambiente matemático.

A [Figura 18](#) representa uma unidade de circuito  $\pi$  aplicada no *software* EMTP-RV. Esse circuito dará origem a uma cascata de unidades servindo para validação da rotina desenvolvida em ambiente Python e aplicativo matemático.

Na [Figura 19](#), mostra-se como foram aplicados os inúmeros circuitos  $\pi$  em cascata para utilização no EMTP-RV.

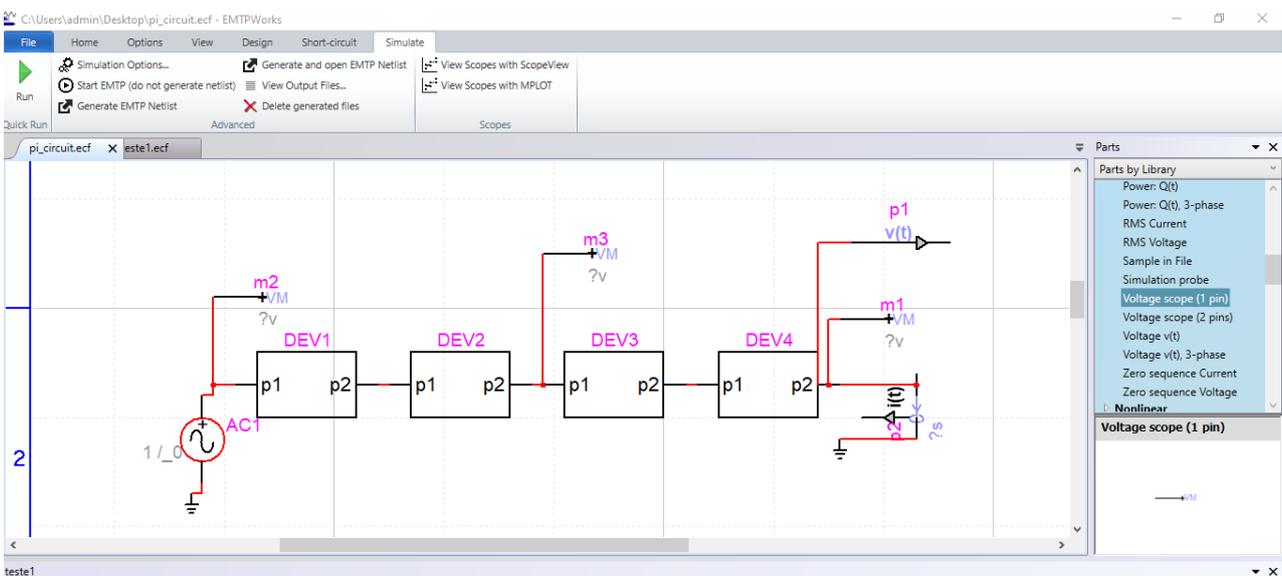
Em um segundo momento, no *software* EMTP-RV, foram aplicadas as características de uma linha de transmissão real no *software* para obtenção dos parâmetros da linha. A [Figura 21](#) mostra os parâmetros obtidos. Com esses valores, foi possível representar uma linha de transmissão no *software* EMTP-RV e realizar a simulação.

Figura 18 – Unidade de circuito  $\pi$  trabalhado.



fonte: Produção do próprio autor.

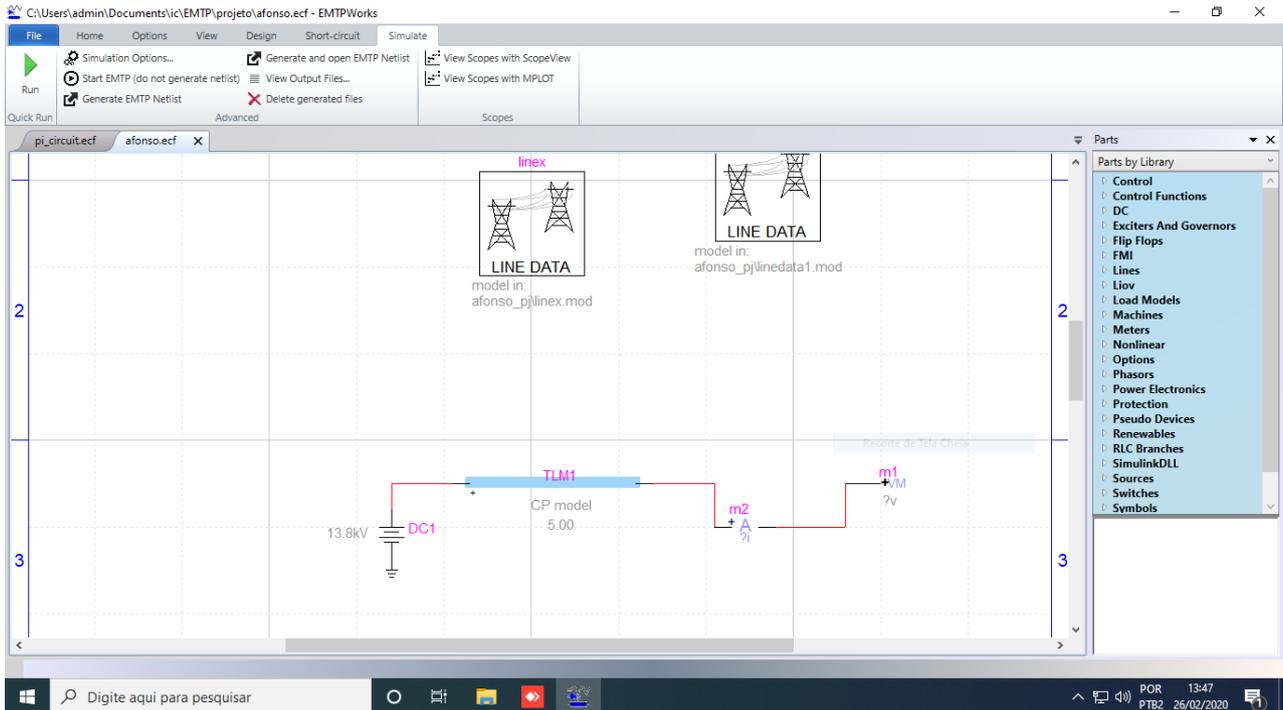
Figura 19 – Janela do *software* EMTP-RV e uma aplicação de unidades de circuito  $\pi$ .



fonte: Produção do próprio autor.

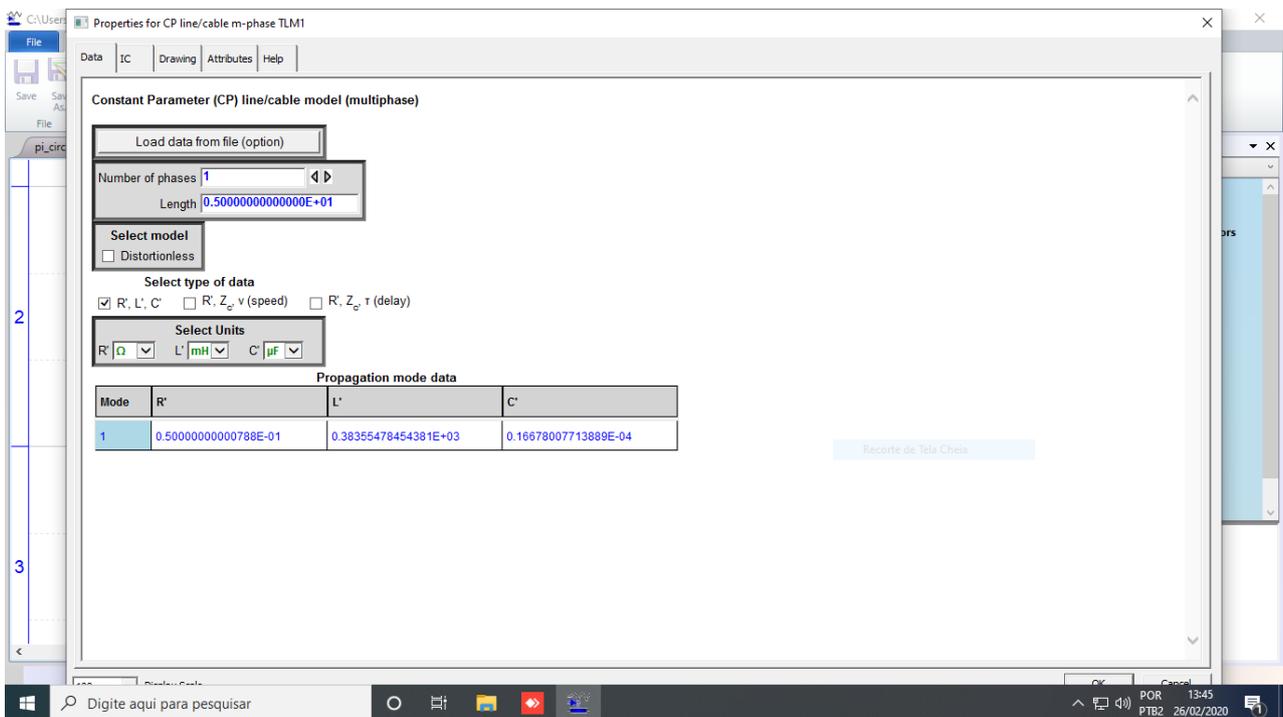
Ainda foi possível simular uma descarga atmosférica descrita por uma função dupla exponencial, aplicando descarga de corrente e tensão para analisar seu comportamento e a rede elétrica é afetada

Figura 20 – Modelo de linha de transmissão obtido pelo *software* EMTP-RV.



fonte: Produção do próprio autor.

Figura 21 – Parâmetros da linha do modelo da Figura 20.

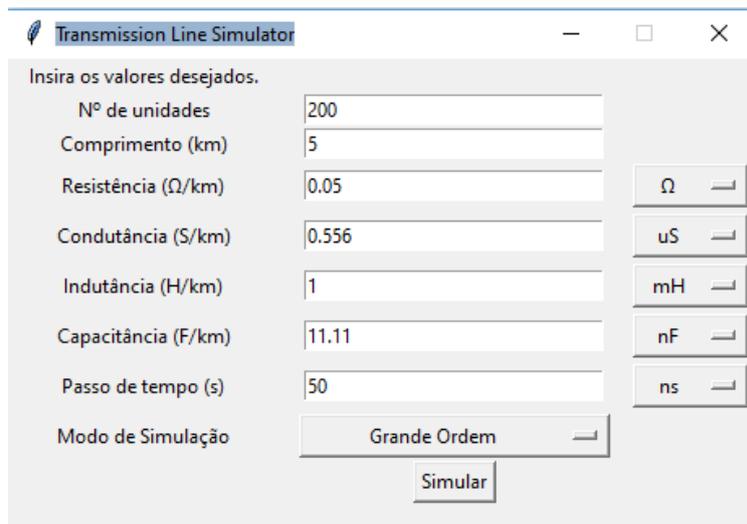


fonte: Produção do próprio autor.

## 4 SOFTWARE DESENVOLVIDO

Foi desenvolvido um *software* para simulações de transitórios eletromagnéticos. O *software*, por sua vez, foi desenvolvido inteiramente em ambiente Python, possibilitando utilização sem a necessidade de instalação. Além disso, permite portabilidade e fácil acesso. As rotinas incluídas no *software* foram desenvolvidas por um grupo de pesquisa composto por pesquisadores da UNESP, alunos de pós-graduação e alunos de graduação [24–50]. Na [Figura 22](#), pode-se observar a vista inicial desse programa, um ambiente simples e eficaz.

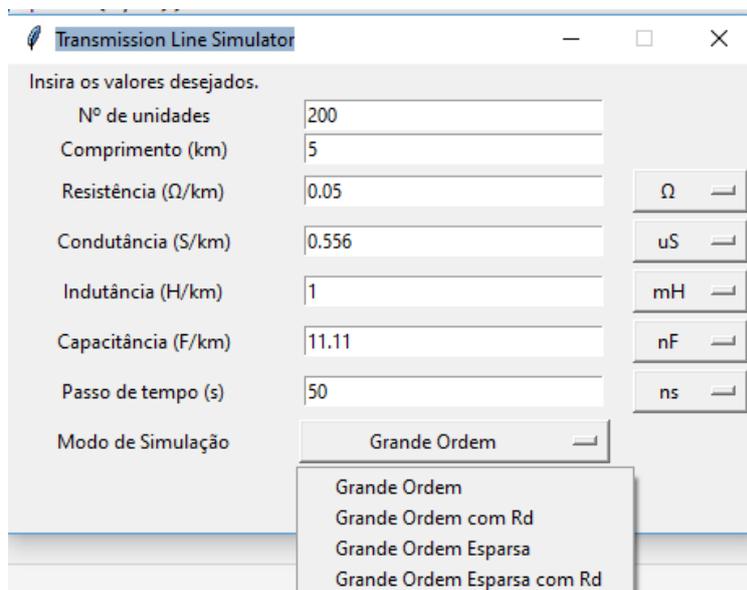
Figura 22 – Software desenvolvido.



fonte: Produção do próprio autor.

Atualmente, é possível selecionar entre os modos de operação como mostrado na [Figura 23](#).

Figura 23 – Modelos de simulação disponíveis.

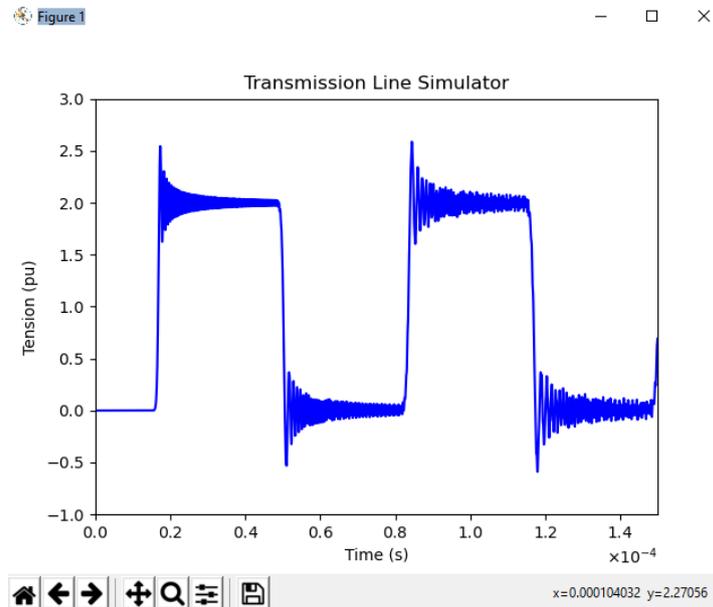


fonte: Produção do próprio autor.

A vantagem do *software* é permitir executar simulações sem a necessidade de instalar novos programas e sem depender de programas e/ou bibliotecas pré-instaladas. Outra vantagem é a versatilidade que o mesmo permite. Todos parâmetros da rotina podem ser alterados de maneira fácil e intuitiva.

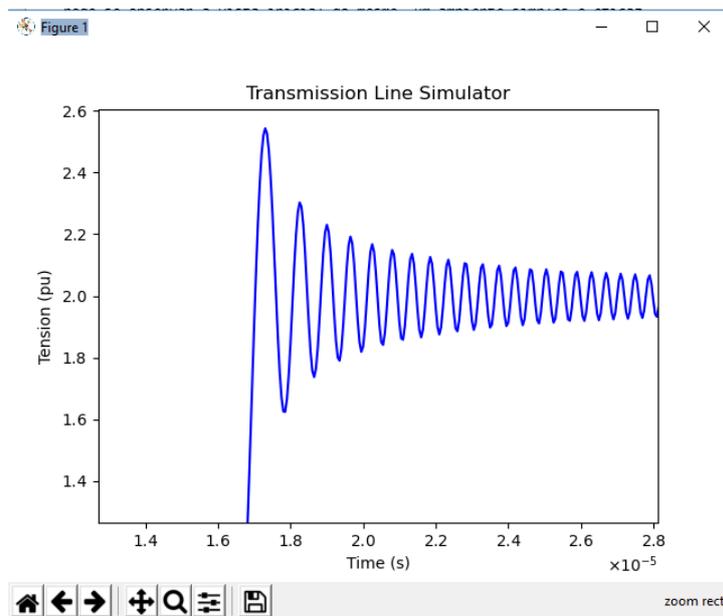
A plataforma que exibe os resultados (Figura 24) é bastante intuitiva, permitindo a manipulação das imagens, como ampliação (Figura 25), marcações de pontos e obtenções de medidas mais precisas.

Figura 24 – Resultado exibido pelo *software*.



fonte: Produção do próprio autor.

Figura 25 – Resultado ampliado exibido pelo *software*.



fonte: Produção do próprio autor.

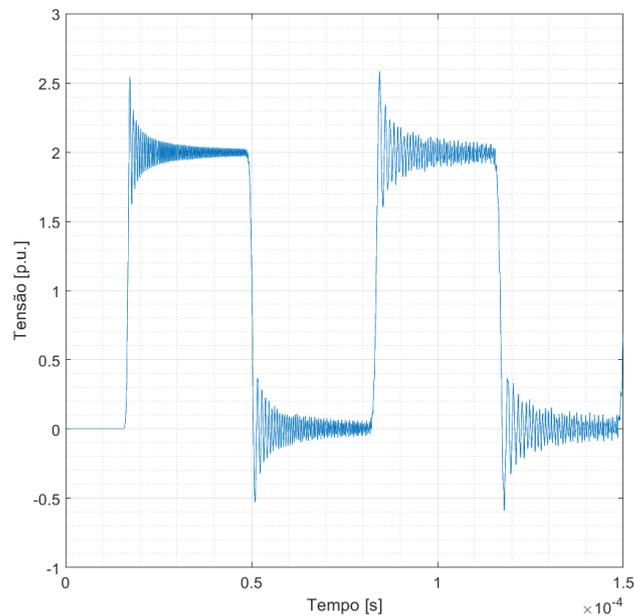
#### 4.1 MATRIZ DE GRANDE ORDEM

No método matriz de grande ordem, foram variados os parâmetros de quantidade de unidades e passo de tempo. Da [Figura 26](#) à [Figura 34](#), os parâmetros por unidade de comprimento da linha mantidos constantes foram:

- Comprimento: 5 km;
- $R' = 0,05 \Omega/km$ ;
- $G' = 0,556 \cdot 10^{-6} S/km$ ;
- $L' = 1 \cdot 10^{-3} H/km$ ;
- $C' = 11,11 \cdot 10^{-9} F/km$ .

Na [Figura 26](#), está representado o resultado para uma linha com 200 segmentos e um passo de tempo de 50 ns. Nesse caso, há oscilações de Gibbs que influenciam e geram um erro de aproximadamente 25 % no primeiro pico de tensão após a reflexão da onda de tensão no terminal final da linha de transmissão.

Figura 26 – Simulação com matriz de grande ordem, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 50 ns$ .

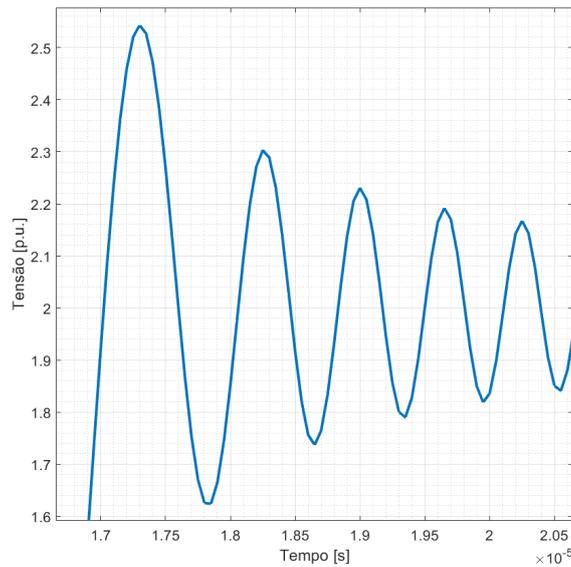


fonte: Produção do próprio autor.

Podemos ampliar o resultado obtido na [Figura 26](#), mostrando-o na [Figura 27](#) para observar as primeiras oscilações. Dessa forma, ficam visíveis as amplitudes e frequências dessas oscilações.

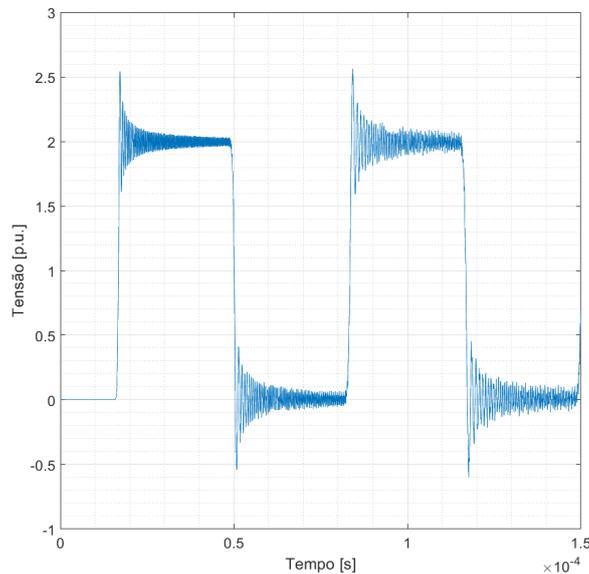
Já a [Figura 28](#) apresenta resultados da mesma linha, porém representada por 400 segmentos.

Figura 27 – Simulação com matriz de grande ordem, detalhe da [Figura 26](#).



fonte: Produção do próprio autor.

Figura 28 – Simulação com matriz de grande ordem, tensão no terminal final da linha,  $n = 400$ ,  $\Delta T = 50 \text{ ns}$ .



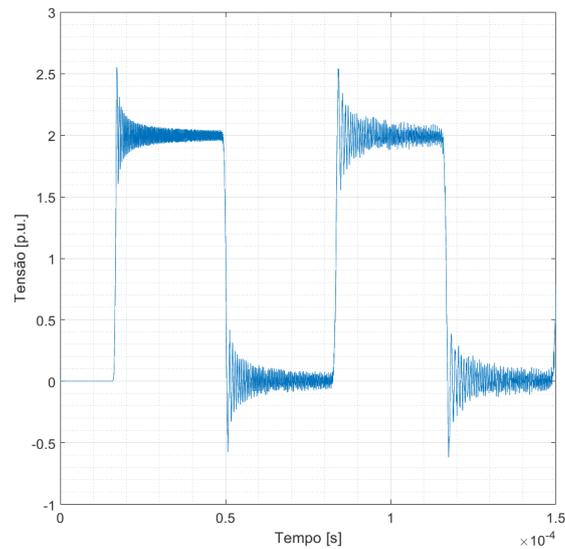
fonte: Produção do próprio autor.

Continuando a expansão, na [Figura 29](#) e [Figura 30](#) representa-se a linha por 800 e 1000 segmentos respectivamente. O modelo de grande ordem mostrou-se estável para simulação de transitórios independentemente do número de circuitos  $\pi$ .

Ao verificar-se que o aumento do número de circuitos  $\pi$  não afetou a simulação, o próximo passo foi variar o passo de tempo da integração trapezoidal. Tal parâmetro afeta a discretização da linha. A [Figura 31](#) mostra resultados para um tempo de integração de 500 ns.

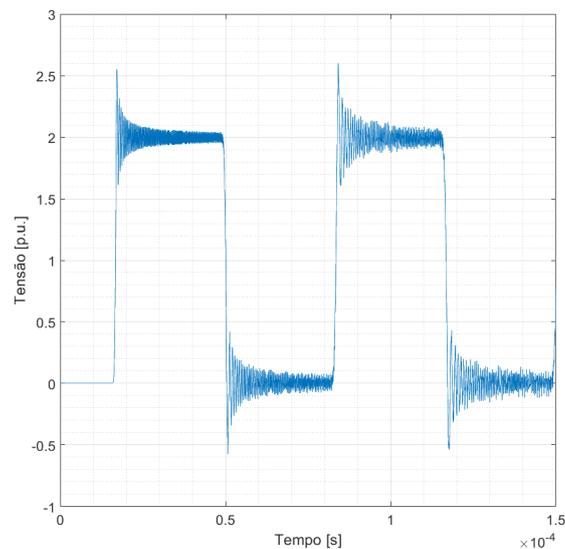
Ao observar as primeiras oscilações da [Figura 31](#), é notório pela [Figura 32](#) que as oscilações não

Figura 29 – Simulação com matriz de grande ordem, tensão no terminal final da linha,  $n = 800$ ,  $\Delta T = 50 \text{ ns}$ .



fonte: Produção do próprio autor.

Figura 30 – Simulação matriz de grande ordem, 1000 segmentos 50 ns.

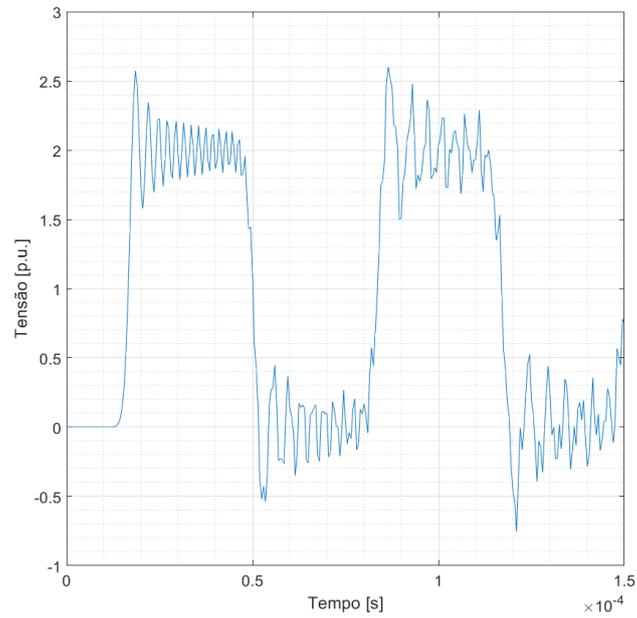


fonte: Produção do próprio autor.

apresentam características senoidais, mostrando que tal passo de tempo não é suficientemente baixo para realizar tal simulação.

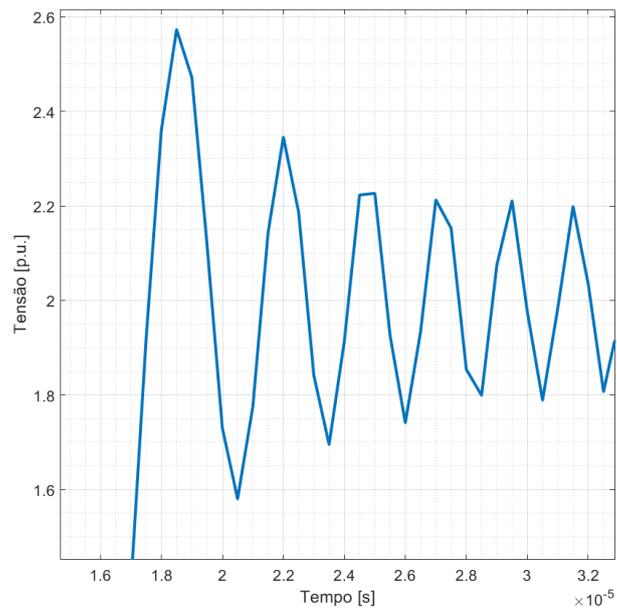
Já na [Figura 33](#), é mostrada a simulação para um passo de tempo de 50 ps. Quando ampliado e visto por meio da [Figura 34](#), o comportamento das oscilações são senoidais, contudo a diminuição desse passo de tempo implica diretamente no recurso computacional utilizado. Logo, há um compromisso entre o tempo computacional, passo de tempo e precisão.

Figura 31 – Simulação com matriz de grande ordem, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 500 \text{ ns}$ .



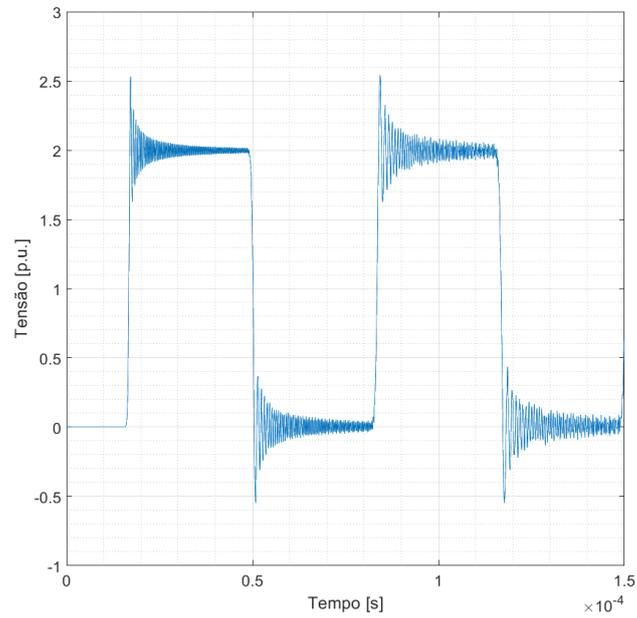
fonte: Produção do próprio autor.

Figura 32 – Simulação com matriz de grande ordem, detalhe da Figura 31.



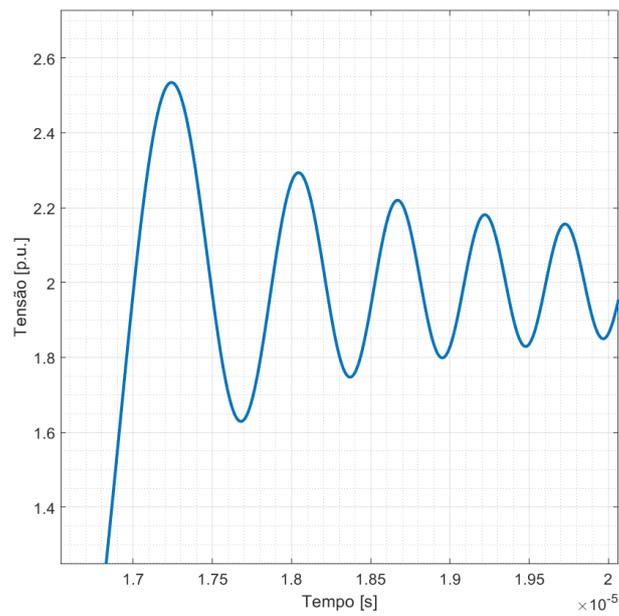
fonte: Produção do próprio autor.

Figura 33 – Simulação com matriz de grande ordem, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 50 \text{ ps}$ .



fonte: Produção do próprio autor.

Figura 34 – Simulação com matriz de grande ordem, detalhe da Figura 33.



fonte: Produção do próprio autor.

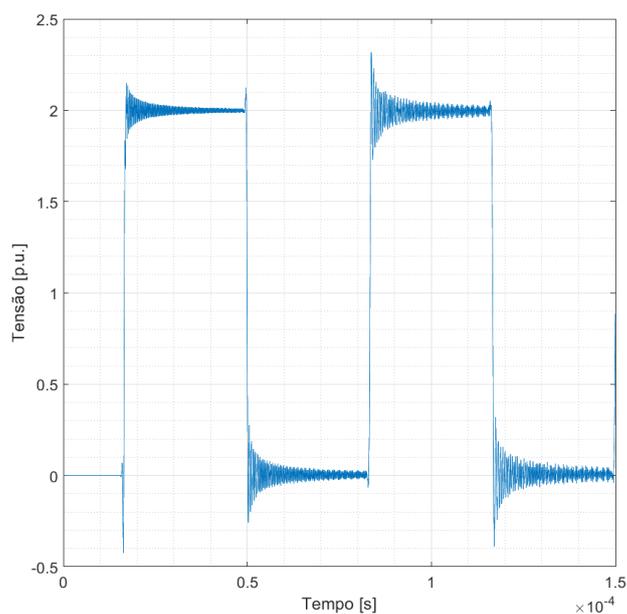
## 4.2 MÉTODO MODIFICADO

Já para o método modificado, também foram variados apenas os parâmetros de quantidade de circuitos ou segmentos  $\pi$  e passo de tempo. Da [Figura 35](#) à [Figura 40](#), estão exibidos os resultados obtidos mantendo os parâmetros por unidade de comprimento em:

- Comprimento: 5 km;
- $R' = 0,05 \Omega/km$ ;
- $G' = 0,556 \cdot 10^{-6} S/km$ ;
- $L' = 1 \cdot 10^{-3} H/km$ ;
- $C' = 11,11 \cdot 10^{-9} F/km$ .

Na [Figura 35](#), mostra-se o resultado obtido para os mesmos parâmetros da [Figura 26](#). O erro das oscilações de Gibbs neste modelo é inferior ao modelo de grande ordem.

Figura 35 – Simulação com matriz de ordem 2,  $n = 200$ ,  $\Delta T = 50 ns$ , tensão no terminal final da linha.

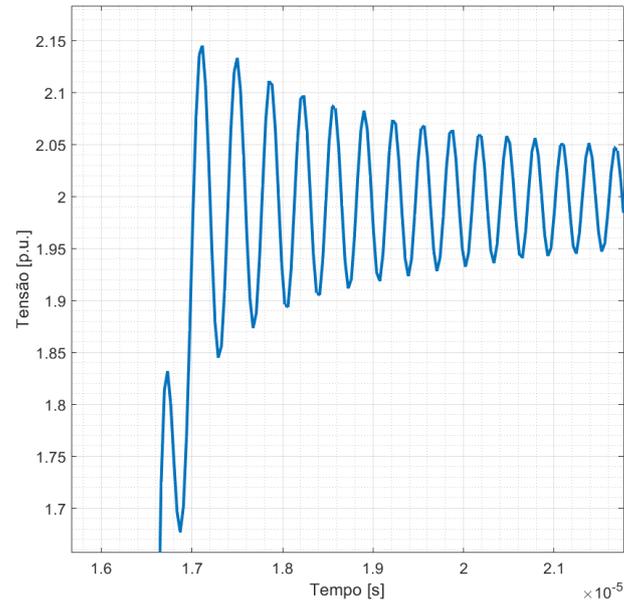


fonte: Produção do próprio autor.

Já na [Figura 36](#), é possível observar as primeiras oscilações do modelo, obtendo valores de amplitude e frequência dessas oscilações.

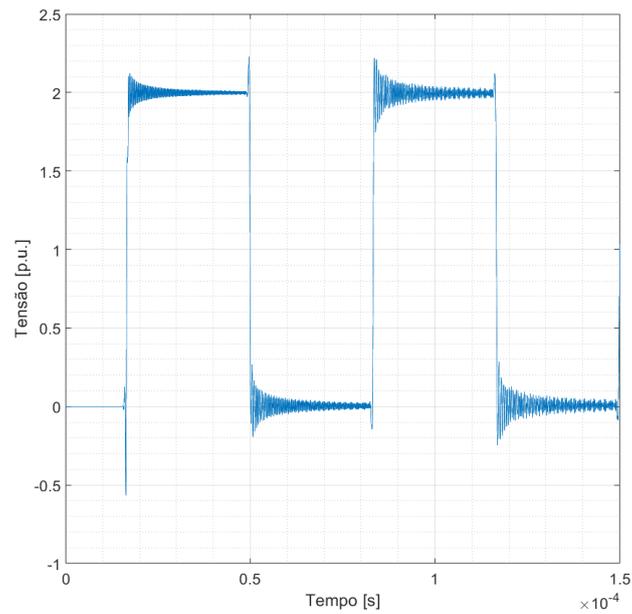
Quando variada a quantidade de circuitos  $\pi$ , os resultados obtidos mostram que a quantidade de unidades tem influência nas oscilações e valores obtidos como visto com 205 segmentos, valor obtido de modo experimental, na [Figura 37](#) e por 333 segmentos pela [Figura 39](#).

Figura 36 – Simulação com matriz de ordem 2, detalhe da Figura 35.



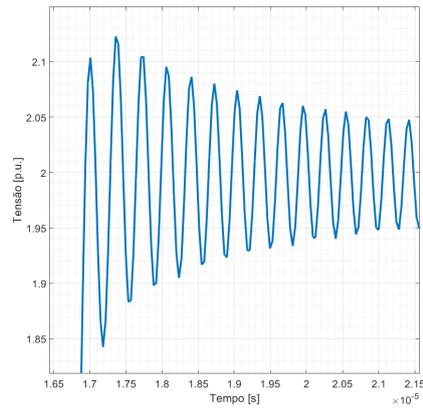
fonte: Produção do próprio autor.

Figura 37 – Simulação com matriz de ordem 2,  $n = 205$ ,  $\Delta T = 50 \text{ ns}$ , tensão no terminal final da linha.



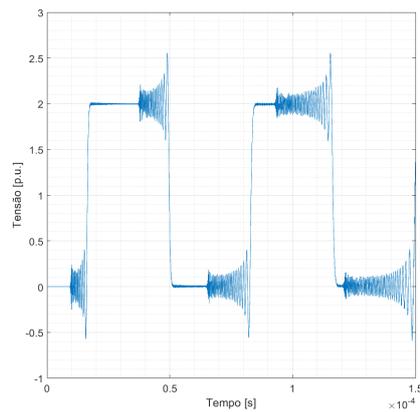
fonte: Produção do próprio autor.

Figura 38 – Simulação com matriz de ordem 2, detalhe da [Figura 37](#).



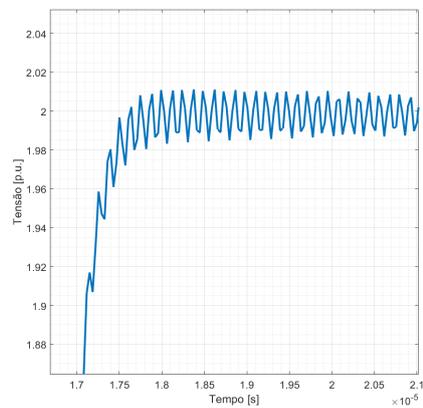
fonte: Produção do próprio autor.

Figura 39 – Simulação com matriz de ordem 2,  $n = 333$ ,  $\Delta T = 50 \text{ ns}$ , tensão no terminal final da linha.



fonte: Produção do próprio autor.

Figura 40 – Simulação com matriz de ordem 2, detalhe da [Figura 39](#).



fonte: Produção do próprio autor.

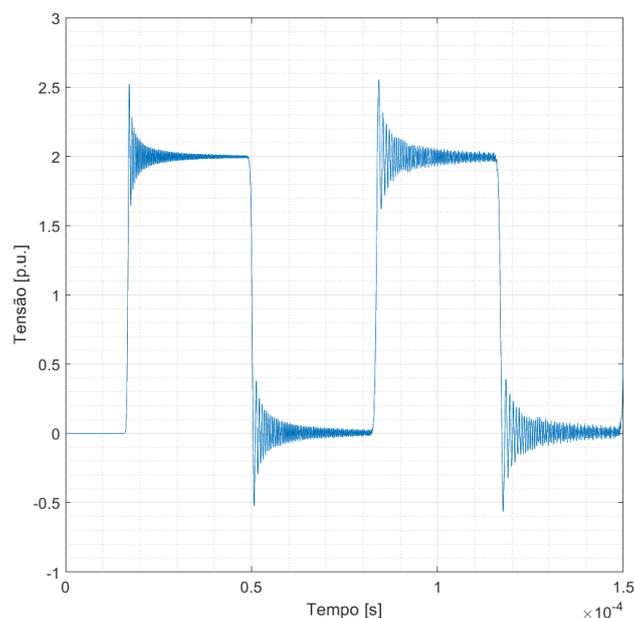
### 4.3 APLICAÇÃO DE DOIS MÉTODOS DE INTEGRAÇÃO NUMÉRICA

Para o método de sistemas lineares, foram variados os parâmetros de quantidade de segmentos e passo de tempo. Nas [Figura 41](#) à [Figura 47](#), estão exibidos os resultados obtidos mantendo os parâmetros por unidade de comprimento em:

- Comprimento:  $5 \text{ km}$ ;
- $R' = 0,05 \text{ } \Omega/\text{km}$ ;
- $G' = 0,556 \cdot 10^{-6} \text{ S}/\text{km}$ ;
- $L' = 1 \cdot 10^{-3} \text{ H}/\text{km}$ ;
- $C' = 11,11 \cdot 10^{-9} \text{ F}/\text{km}$ .

Já na [Figura 41](#), a simulação foi realizada com os mesmos parâmetros da [Figura 26](#). Os resultados obtidos foram similares ao modelo de grande ordem.

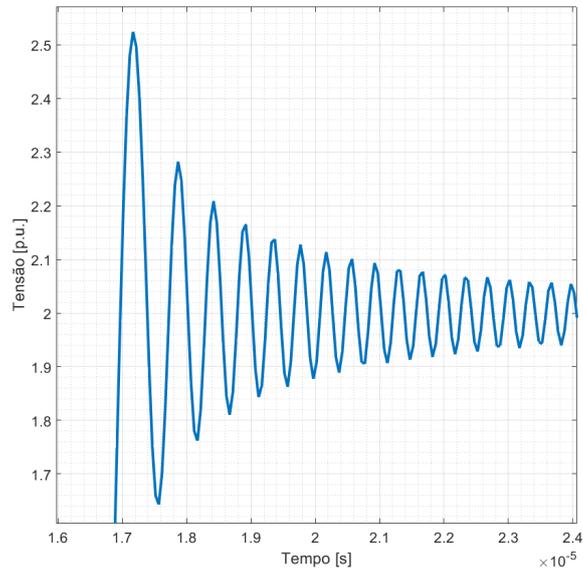
Figura 41 – Simulação aplicando dois métodos de integração numérica, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 50 \text{ ns}$ .



fonte: Produção do próprio autor.

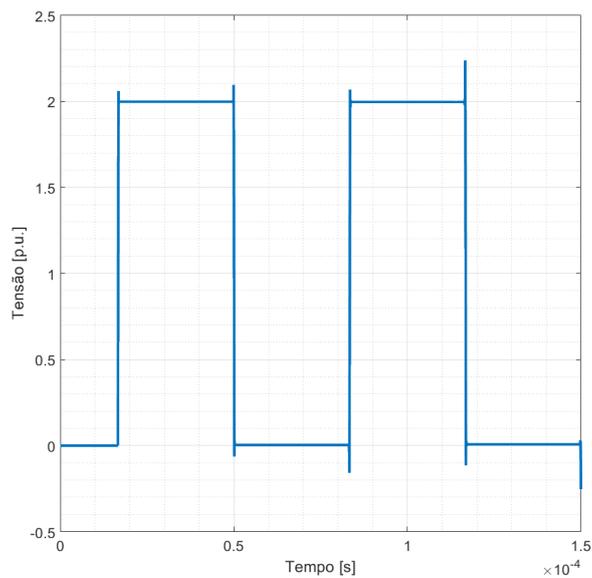
As primeiras oscilações mostradas na [Figura 42](#) também são semelhantes aos os resultados anteriores. O método possuiu um ponto ótimo, com 333 segmentos, no qual o erro do primeiro pico de tensão decaiu de 25 % para 2,5 %. Isso pode ser visto pela [Figura 43](#). Para analisar a primeira variação da [Figura 43](#), pode-se ampliar obtendo a [Figura 44](#). Contudo, o método mostrou problemas de estabilidade matemática. Para valores de passo de tempo maiores do que aquele utilizado na [Figura 43](#), os resultados voltaram a apresentar erros de 25 %, como mostrado pela [Figura 45](#) e [Figura 46](#). Já para quantidades maiores de circuitos  $\pi$ , o método não é viável e o resultado pode ser visto na [Figura 47](#).

Figura 42 – Simulação aplicando dois métodos de integração numérica, detalhe da Figura 41.



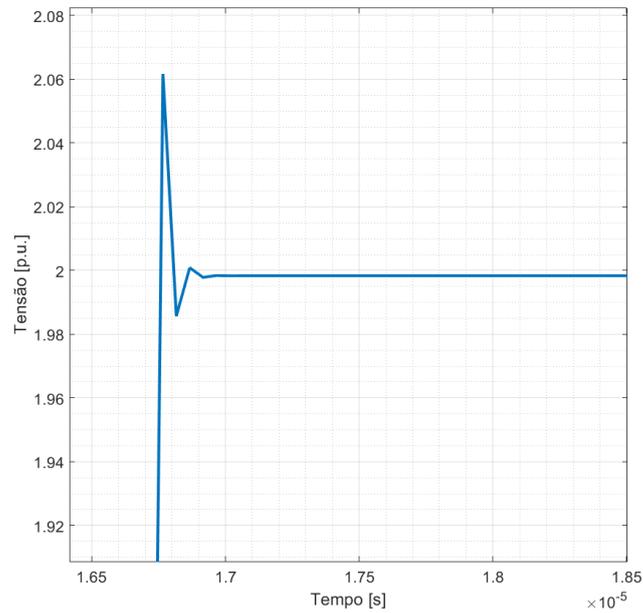
fonte: Produção do próprio autor.

Figura 43 – Simulação aplicando dois métodos de integração numérica, tensão no terminal final da linha,  $n = 333$ ,  $\Delta T = 50 \text{ ns}$ .



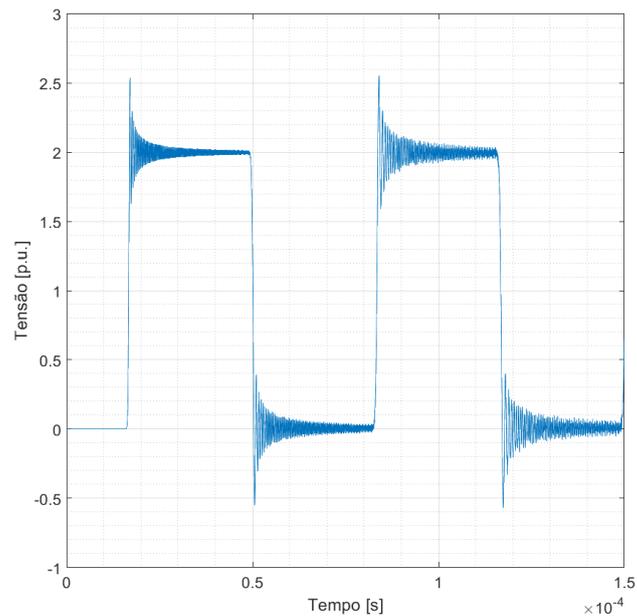
fonte: Produção do próprio autor.

Figura 44 – Simulação aplicando dois métodos de integração numérica, detalhe da Figura 43.



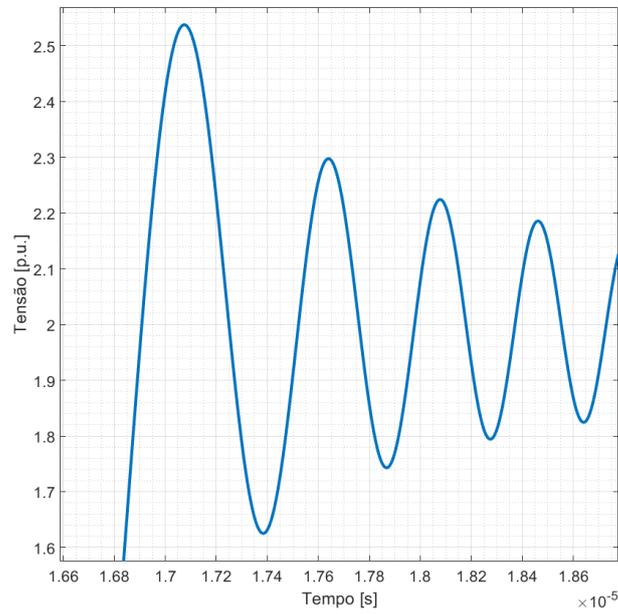
fonte: Produção do próprio autor.

Figura 45 – Simulação aplicando dois métodos de integração numérica, tensão no terminal final da linha,  $n = 333$ ,  $\Delta T = 50$  ps.



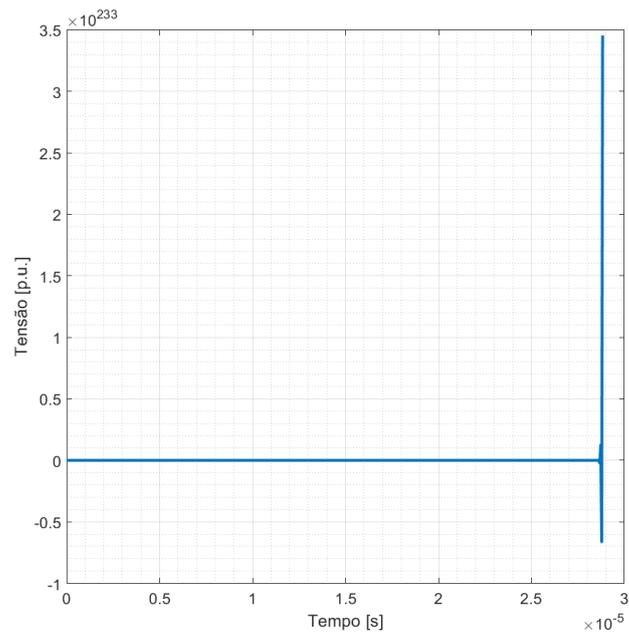
fonte: Produção do próprio autor.

Figura 46 – Simulação aplicando dois métodos de integração numérica, detalhe da Figura 45.



fonte: Produção do próprio autor.

Figura 47 – Simulação aplicando dois métodos de integração numérica, tensão no terminal final da linha,  $n = 400$ ,  $\Delta T = 50 \text{ ps}$ .



fonte: Produção do próprio autor.

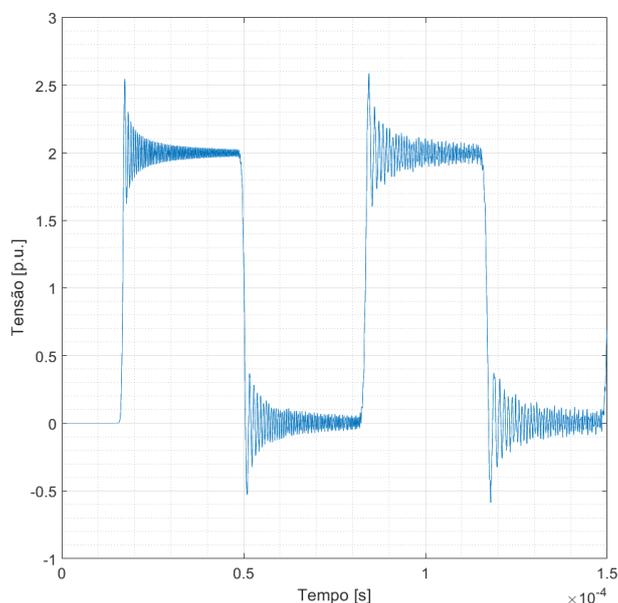
#### 4.4 MATRIZ ESPARSA

A matriz esparsa teve como base a matriz de grande ordem e as simulações realizadas tiveram como base a variação conforme o modelo matriz de grande ordem. Os resultados obtidos podem ser expressos pela [Figura 48](#) à [Figura 54](#), mantendo os parâmetros por unidade de comprimento em:

- Comprimento: 5 km;
- $R' = 0,05 \Omega/km$ ;
- $G' = 0,556 \cdot 10^{-6} S/km$ ;
- $L' = 1 \cdot 10^{-3} H/km$ ;
- $C' = 11,11 \cdot 10^{-9} F/km$ .

Novamente, para efeito de comparação a primeira simulação foi realizada com os mesmos parâmetros utilizados para obtenção da [Figura 26](#) e o resultado obtido é mostrado na [Figura 48](#), sendo igual ao método de grande ordem. Houve um tempo de simulação menor em ambiente matemático e um tempo superior em Python. Em ambos os casos, houve redução de recurso computacional para armazenamento.

Figura 48 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 50 ns$ .

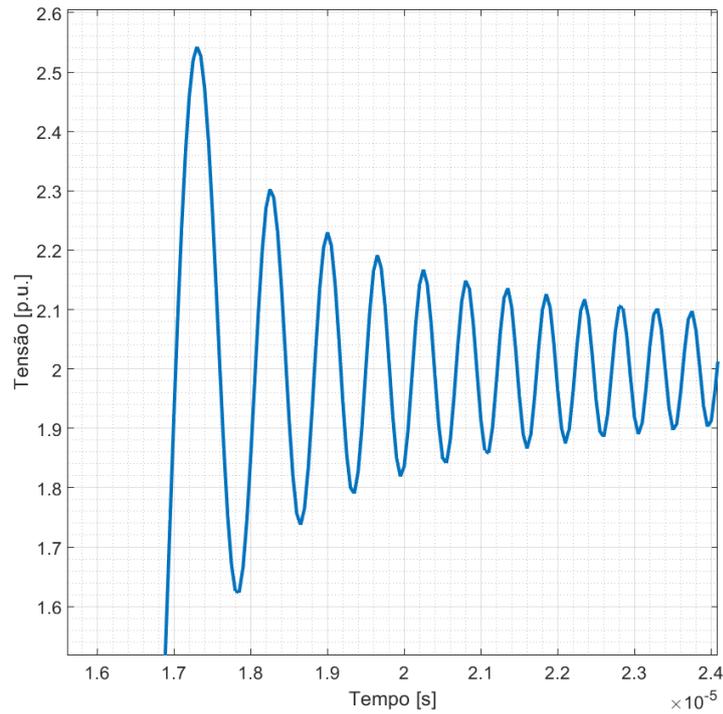


fonte: Produção do próprio autor.

Para comparar o desempenho em tempo de simulação e recurso dispensado, foram ampliados o número de segmentos para 400, 800, 100. Os resultados são mostrados nas [Figura 50](#), [Figura 51](#) e [Figura 52](#) respectivamente.

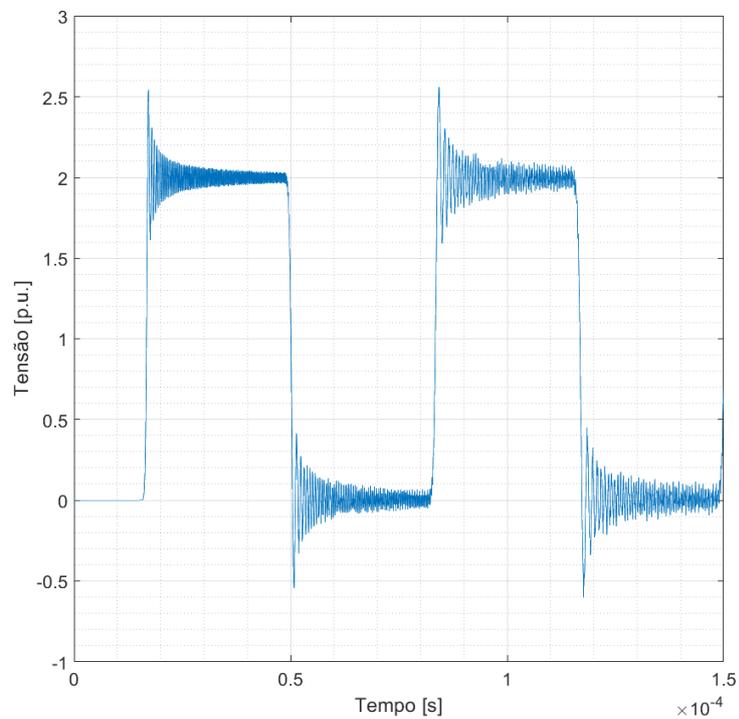
Já na [Figura 53](#) e [Figura 54](#), estão representadas as primeiras oscilações, para análise da influência do passo de tempo na simulação.

Figura 49 – Simulação com esparsidade, detalhe da Figura 48.



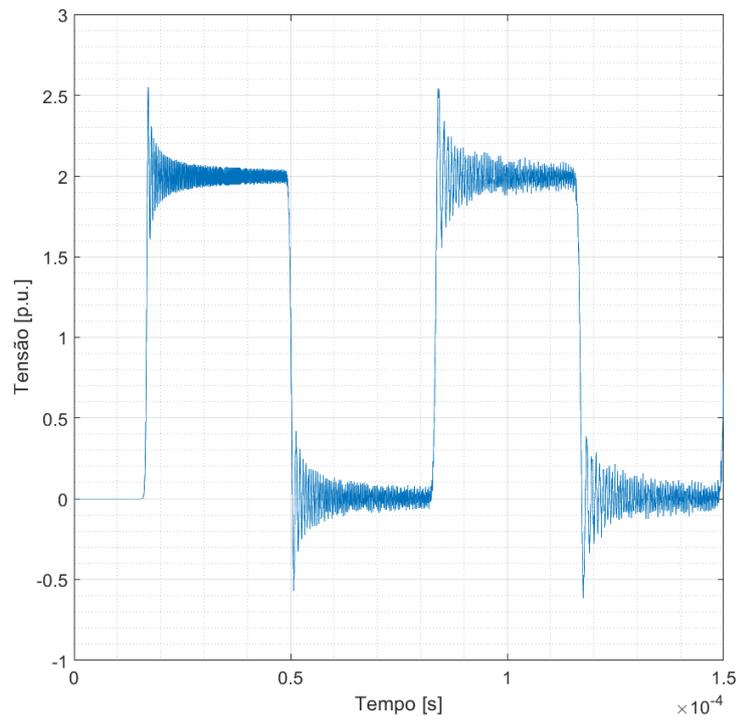
fonte: Produção do próprio autor.

Figura 50 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 400$ ,  $\Delta T = 50$  ns.



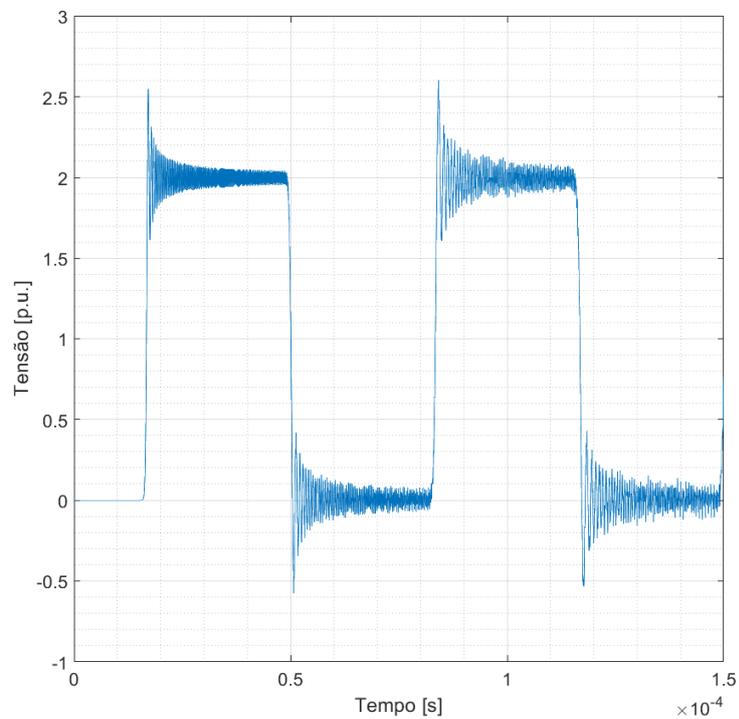
fonte: Produção do próprio autor.

Figura 51 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 800$ ,  $\Delta T = 50 \text{ ns}$ .



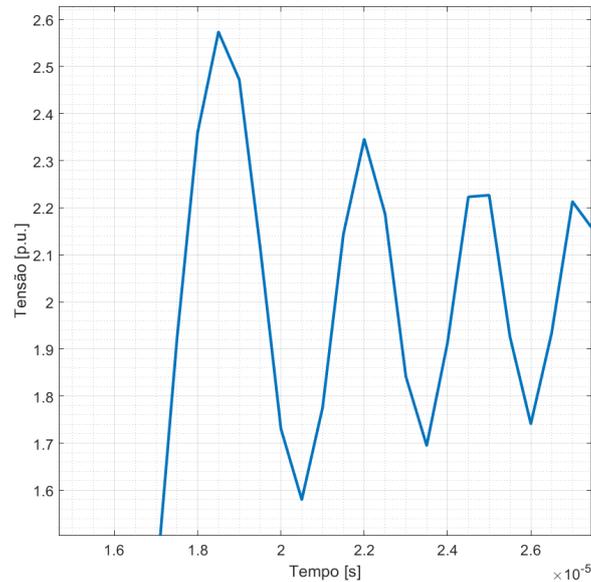
fonte: Produção do próprio autor.

Figura 52 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 1.000$ ,  $\Delta T = 50 \text{ ns}$ .



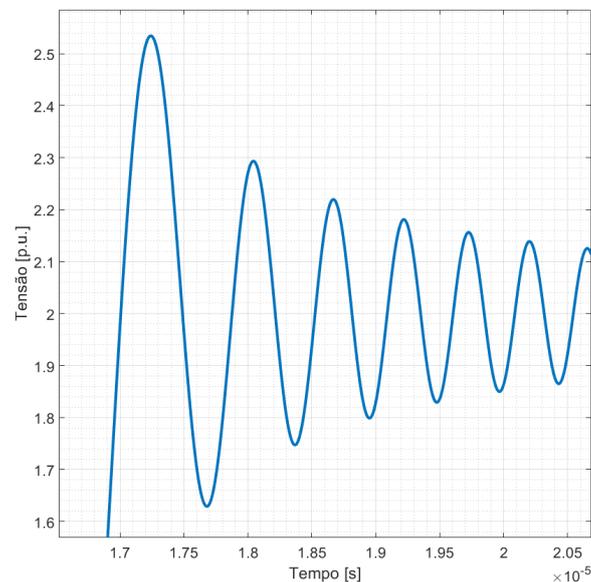
fonte: Produção do próprio autor.

Figura 53 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 500 \text{ ns}$ , em detalhe.



fonte: Produção do próprio autor.

Figura 54 – Simulação com esparsidade, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 500 \text{ ps}$ , em detalhe.



fonte: Produção do próprio autor.

Nas Tabelas [1](#) e [2](#), são apresentadas as comparações entre a rotina com a técnica de esparsidade e a rotina sem a técnica de esparsidade. A Tabela [1](#) mostra os resultados obtidos utilizando o aplicativo matemático e a Tabela [2](#) está relacionada ao ambiente Python. Os valores foram obtidos utilizando um processador Ryzen 2600X em 4 GHz, com memórias do tipo DDR4 2.400 MHz. Já na Tabela [3](#), mostra a redução de alocação de memória RAM necessária para as simulações.

Tabela 1 – Comparação aplicando técnica de esparsidade usando aplicativo matemático.

Nº de circuitos $\pi$	Tempo sem esparsidade (s)	Tempo com esparsidade (s)
200	0.11	0.09
800	6.93	4.11
1000	10.96	6.74
2000	52.51	37.11

fonte: Produção do próprio autor.

Tabela 2 – Comparação aplicando técnica de esparsidade usando Python.

Nº de circuitos $\pi$	Tempo sem esparsidade (s)	Tempo com esparsidade (s)
200	0.61	1.4
800	4.83	12.81
1000	7.85	21.96
2000	33.41	95.88

fonte: Produção do próprio autor.

Tabela 3 – Comparação do recurso computacional para armazenamento da matriz principal aplicando técnica de esparsidade.

Nº de circuitos $\pi$	Memória sem esparsidade (kB)	Memória com esparsidade (KB)
200	1.250	21,85
800	20.000	87,47
1000	31.250	109,35
2000	125.000	218,72

fonte: Produção do próprio autor.

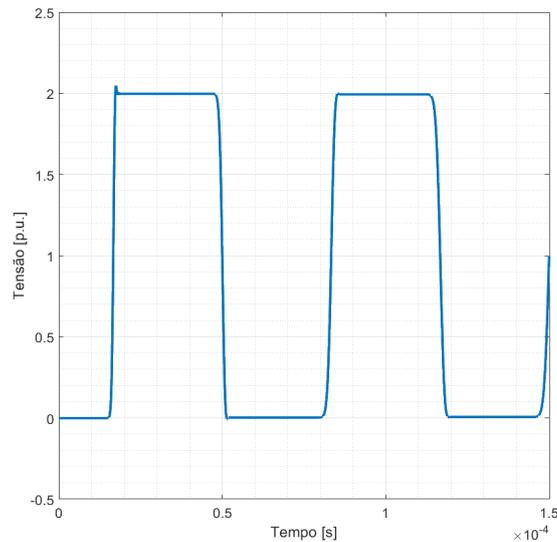
#### 4.5 RESISTÊNCIA DE AMORTECIMENTO NOS PARÂMETROS LONGITUDINAIS

Para eliminar as oscilações de Gibbs, foi proposta a utilização de resistências de amortecimento. As [Figura 55](#) à [Figura 60](#) mostram resultados aplicando tal modelo, mantendo os parâmetros por unidade de comprimento em:

- Comprimento: 5 km;
- $R' = 0,05 \Omega/km$ ;
- $G' = 0,556 \cdot 10^{-6} S/km$ ;
- $L' = 1 \cdot 10^{-3} H/km$ ;
- $C' = 11,11 \cdot 10^{-9} F/km$ .

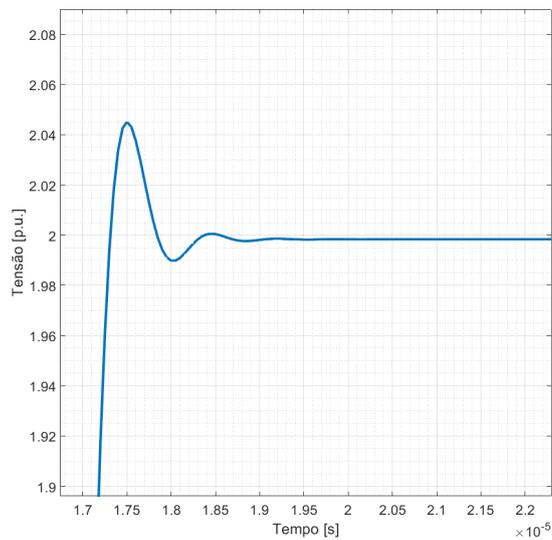
Seguindo o mesmo padrão, a [Figura 55](#) apresenta resultados para uma linha com 200 segmentos, onde as oscilações de Gibbs foram amortecidas. Por meio da [Figura 56](#) é possível verificar a primeira oscilação da [Figura 55](#) em detalhes.

Figura 55 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha,  $n = 200$ ,  $\Delta T = 50 \text{ ns}$ .



fonte: Produção do próprio autor.

Figura 56 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 55.

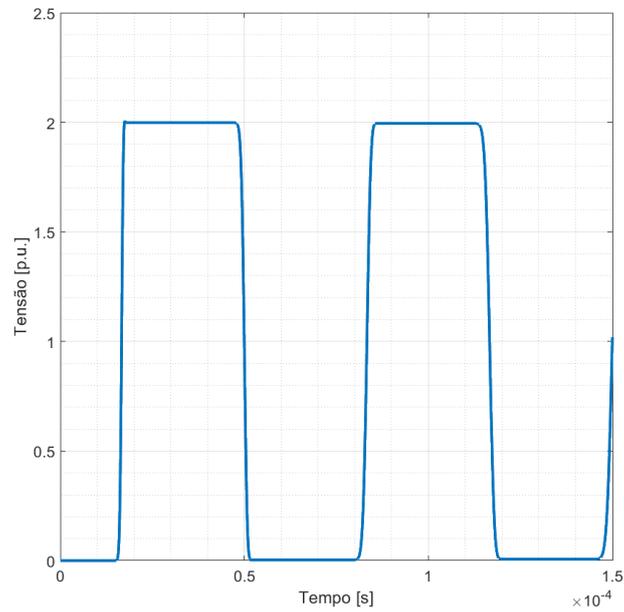


fonte: Produção do próprio autor.

Já a Figura 57, representa uma linha elevando o número de segmentos para 400, a oscilação decaiu ainda mais quando comparado pela Figura 58.

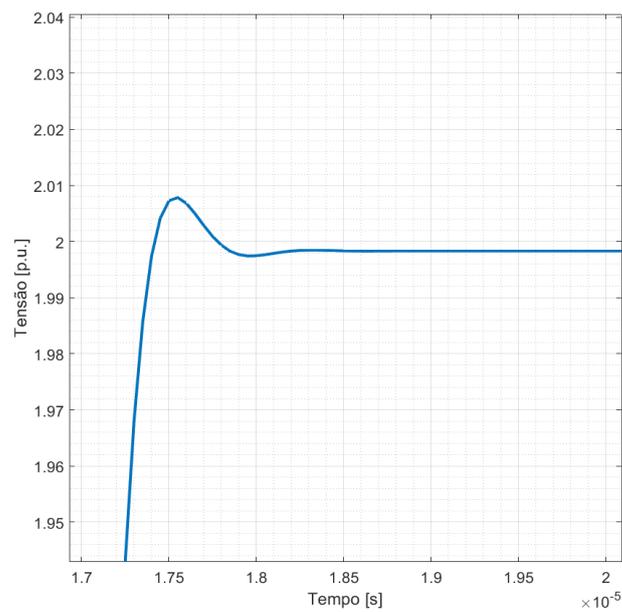
O mesmo fenômeno ocorre conforme se eleva o número de segmentos de circuitos  $\pi$  para 800, como visto pela Figura 59, e a oscilação pode ser vista por meio da Figura 60.

Figura 57 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha,  $n = 400$ ,  $\Delta T = 50 \text{ ns}$ .



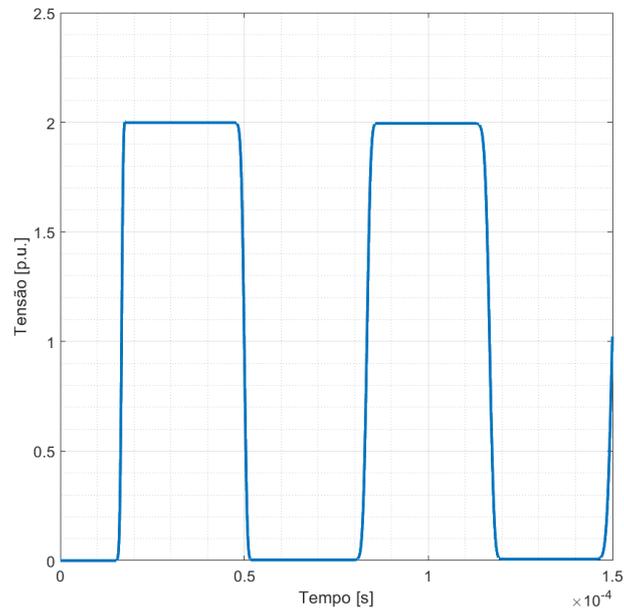
fonte: Produção do próprio autor.

Figura 58 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 57.



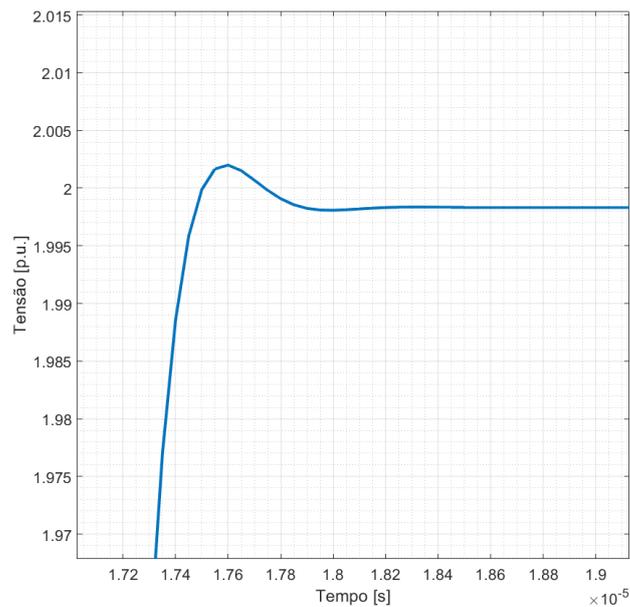
fonte: Produção do próprio autor.

Figura 59 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, tensão no terminal final da linha,  $n = 800$ ,  $\Delta T = 50 \text{ ns}$ .



fonte: Produção do próprio autor.

Figura 60 – Simulação aplicando resistência de amortecimento nos parâmetros longitudinais, detalhe da Figura 59.



fonte: Produção do próprio autor.

## 5 CONCLUSÃO

Neste trabalho, foram desenvolvidas rotinas numéricas para simulação de transitórios eletromagnéticos em linhas de transmissão. As primeiras rotinas trabalhadas foram com base em modelos simples, como a rotina baseada em matrizes de grande ordem. Tal rotina apresentou um bom desempenho computacional. Contudo, houve um alto erro percentual em relação ao primeiro pico de onda de tensão durante a primeira reflexão dessa onda no terminal de carga da linha, considerando a situação de linha em aberto. Ao implementar no ambiente Python a rotina demonstrou-se mais eficiente do que em *software* matemático. O segundo modelo abordado foi representar segmento por segmento da linha. A aplicação desse método em ambiente Python também se mostrou viável, contudo ao comparar o tempo de simulação o *software* matemático teve vantagem e o método utilizado teve influência do número de segmentos utilizados.

Já na aplicação de dois métodos de integração numérica, é possível reduzir as oscilações de Gibbs, contudo o método mostrou-se bastante instável, permitindo obter resultados de apenas certos modelos, a influência do tempo de integração ou número de segmentos afetou bruscamente o resultado. A aplicação no Python mostrou-se eficiente bem como em plataforma matemática.

Ao aplicar a técnica de esparsidade, é possível reduzir a alocação de memória RAM para ambas plataformas utilizadas, além de permitir uma redução no tempo de simulação em aplicativo matemático. Em relação ao Python, foi possível aplicar tal técnica. Resultados sem aplicar técnicas de esparsidade no Python se mostram mais eficientes do que resultados aplicando tal técnica em aplicativo matemático, quando o número de segmentos é elevado.

O último método abordado, foi com resistência de amortecimento. Esse método se mostrou eficaz em minimizar oscilações de Gibbs, reduzindo erros percentuais.

Foram comparados resultados obtidos utilizando as rotinas desenvolvidas em ambiente Python e resultados do *software* EMTP-RV, permitindo a validação do modelo desenvolvido e incluído no *software* desenvolvido. Dessa forma, é apresentado, ao final deste trabalho, um *software* com base na linguagem Python, capaz de simular uma linha de transmissão por meio dos seus parâmetros por unidades de comprimento. O programa permite também selecionar a rotina desejada, de forma a variar o modo de simulação para comparação dos resultados. Tal *software* é uma opção para pessoas leigas na linguagem Python que tenham interesse em realizar simulações numéricas simples de transitórios eletromagnéticos, possuindo assim uma interface amigável e instrutiva.

## REFERÊNCIAS

- [1] H. W. Dommel, A. Yan, R. J. Ortiz De Marcano, A. B. Miliani, Case Studies for Electromagnetic Transients, University of British Columbia, Vancouver, Canada, May, 1983.
- [2] H. W. Dommel, "Digital computer solution of electromagnetic transients in single and multiphase networks", IEEE Trans. on Power Apparatus and Systems, vol. PAS-88, pp. 388-399, April, 1969.
- [3] J. R. Marti, "Accurate model of frequency-dependent transmission lines in 105 electromagnetic transient simulations", IEEE Trans. on Power Apparatus and Systems, vol. PAS-101, n° 1, pp. 147-155, January, 1982.
- [4] T. Ingelbinck, P. R. Andrade, L. S. Lessa, A. J. Prado, L. F. Bovolato, J. Pissolato Filho, "Analyses of the frequency-dependent model's inductances' resistances for transmission lines", CLAGTEE 2013, CD-ROM, 5 pp., 6-9 de outubro de 2013, Viña del Mar, Chile.
- [5] P. R. Andrade, R. C. Monzani, L. S. Lessa, t. Ingelbinck, A. J. Prado, J. Pissolato Filho, L. F. Bovolato, "Physical model for representing transmission lines by undergraduate students", CLAGTEE 2013, CD-ROM, 5 pp., 6-9 de outubro de 2013, Viña del Mar, Chile.
- [6] J. A. R. Macías, A. G. Expósito, A. B. Soler, "A Comparison of Techniques for Statespace Transient Analysis of Transmission Lines", IEEE Transactions on Power Delivery, vol. 20, n° 2, pp. 894-903, April, 2005.
- [7] Lines. IEE Proceedings Generation, Transmission and Distribution, vol. 149, NO. 6; pp. 705-712.
- [8] M. S. Mamis, and A. Nacaroglu (2003). Transient Voltage and Current Distributions on Transmission
- [9] H.W. Dommel, Electromagnetic Transients Program. Reference Manual (EMTP Theory Book), Bonneville Power Administration, Portland, 1986.
- [10] Microtran - Reference Manual - Transients Analysis Program for Power and Power Electronic Circuits, Microtran Power System Analysis Corporation, Vancouver, B.C., Canada, September, 1992.
- [11] B. Alberto, M. Luca, "Python Data Science Essentials - Second Edition", October 2016.
- [12] F. Claus, S. E. Jan, V. Olivier, "Scientific Computing with Python 3", December 2016.
- [13] M. T. Correia de Barros, J. F. Borges da Silva, "A non-Linear Propagation Algorithm for Transmission Lines", Proceedings of the Eight Power Systems Computation Conference, pp 772-776, Butterworths, August 1984.
- [14] M. T. Correia de Barros, J. F. Borges da Silva, "Numerical Filtering of High Frequency Oscillations", EMTP Newsletter, Vol 7, n° 2, pp 23-32, June 1987.

- [15] ATPDRAW version 3, User Manual, TR A4389, EFI, Norway, 1996
- [16] R. M. Nelms, G. B. Sheble, S. R. Newton, L. L. Grigsby, "Using a personal computer to teach power system transients", IEEE Transactions on Power Systems, vol. 4, n° 3, 1293-1297, August, 1989.
- [17] M. S. Mamis, (2003). Computing of Electromagnetic Transients on Transmission Lines with Nonlinear Components. IEE Proceedings Generation, Transmission and Distribution, vol, 150, NO. 2; PP. 200-203.
- [18] H. W. Dommel, "Techniques For Analyzing Electromagnetic Transients", IEEE Computer Applications on Power, vol. 10, no, 3, pp. 18-21, July, 1997.
- [19] H. V. Nguyen, H. W. Dommel, J. R. Martí, "Modelling of single-phase nonuniform transmission lines in electromagnetic transient simulations", IEEE Transactions on Power Delivery, vol.12, no. 2, pp. 916-921, April, 1997.
- [20] PRADO, A. J.; JUS, L H., SANTOS, M. O.; GENNARO, E. M.; FERREIRA, A. A.; PEREIRA, T. G.; MOREIRA, A. C.; MENZINGER, J. S.; GRILO, C. V. C.; CARGNIN STIELER, M.; PISSOLATO FILHO, J.; "Application of numeric routine for simulating transients in power line communication (PLC) Systems" (DOI 10.5772/intechopen.74753) in Waveguide, Kok Yeow You, ISBN 978-1-78923-492-3 (Print) e 978-1-78923-493-0 (Online), InTech - Open Access Publisher, Rijeka, Croatia, 2018, pp. 351-371.
- [21] LESSA, L. S.; LUCA, C. C. S.; PEREIRA, T. G.; GRILO, C. V. C.; MOREIRA, A. C.; RONCHINI, C. M. B.; GENNARO, E. M.; AGOPIAN, P. G. D.; PRADO, A. J.; "Sparse matrices for transient simulations with computing memory reduction", Electric Power Systems Research, vol.183, 7 pp, June, 2020.
- [22] GRILO, C. V. C.; MOREIRA, A. C.; MENZINGER, J. S.; LUCA, C. C. S.; RONCHINI, C. M. B.; PRADO, A. J.; "Análise modificada de transitórios eletromagnéticos em linhas de transmissão", Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, vol.7, no. 1, 2 pp., 2020.
- [23] GRILO, C. V. C.; MOREIRA, A. C.; Pereira, T. G.; MENZINGER, J. S.; PRADO, A. J.; "Rotina numérica para simulação de transitórios eletromagnéticos utilizando ambiente Python", Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, vol. 6, no. 2, 2 pp., 2018.
- [24] PRADO, A. J.; BESPALHULK, K. J.; SILVA, B. F.; CONCEIÇÃO, K. O.; CARGNIN STIELER, M.; GENNARO, E. M.; PISSOLATO FILHO, J.; "Influences of damping resistances on transient simulations in transmission lines", PIER B - Progress In Electromagnetics Research B, ISSN: 1937-6472, DOI 10.2528/PIERB1609270, vol. 75, pp. 27-39, 2017.
- [25] RONCHINI, C. M. B.; MENZINGER, J. S.; LUCA, C. C. S.; GRILO, C. V. C.; MOREIRA, A. C.; PEREIRA, T. G.; LESSA, L. S.; FERREIRA, A. A.; PRADO, A. J., "Electromagnetic

phenomena simulation with application of damping resistances", XIII Latin-american Congress on Electricity, Generation and Transmission - CLAGTEE 2019, 7 pp. 20 a 23 de outubro de 2019, Santiago, Chile.

- [26] MOREIRA, A. C.; PEREIRA, T. G.; LUCA, C. C. S.; RONCHINI, C. M. B.; GRILO, C. V. C.; LESSA, L. S.; Gennaro, E. M.; AGOPIAN, P. G. D.; PRADO, A. J., "Physical model of transmission lines: adaptation for PLC systems", XII Latin-american Congress on Electricity, Generation and Transmission - CLAGTEE 2019, 4 pp., 20 a 23 de outubro de 2019, Santiago, Chile.
- [27] GRILO, C. V. C.; MOREIRA, A. C.; LUCA, C. C. S.; RONCHINI, C. M. B.; MENZINGER, J. S.; PEREIRA, T. G.; LESSA, L. S.; PRADO, A. J. "Simulation of transient phenomena in Python platform using 2-order matrices", XIII Latin-american Congress on Electricity, Generation and Transmission - CLAGTEE 2019, 6 pp., 20 a 23 de outubro de 2019, Santiago, Chile.
- [28] SANTOS, M. O.; SOUZA, W. S.; BRAGAGNOLLE, T. A.; BOBADILLA, L. D. B.; ALDAYA, I.; PRADO, A. J.; FERREIRA, A. A.; ABBADE, M. L. F.; NASCIMENTO, L. H. B.; "All-optical spectral shuffling applied to 16-QAM signals", International Optics and Photonics Conference, IOPC-SBFoton 2019, 5 pp., 7 a 9 de outubro de 2019, São Paulo, SP.
- [29] BRAGAGNOLLE, T. A.; NOGUEIRA, M. P.; SANTOS, M. O.; PRADO, A. J.; FERREIRA, A. A.; FAGOTTO, E. A. M. F.; ALDAYA, I.; ABBADE, M. L. F.; "All-Optical Spectral Shuffling of Signals Traveling through Different Optical Routes", The 21st International Conference on Transparent Optical Networks, ICTON-2019, 4 pp., 9 a 13 de julho de 2019, Angers, France.
- [30] PEREIRA, T. G.; MOREIRA, A. C.; GRILO, C. V. C.; MENZINGER, J. S.; FERREIRA, A. A.; PRADO, A. J.; SCHEUER, R.; "Laboratory tests with model based on circuits considering the introduction of branches composing by RL parallel elements", International Telecommunications Energy Conference - INTELEC 2018, 6 pp., 7 a 11 de outubro de 2018, Turim, Itália.
- [31] ABBADE, M. L. F.; LESSA, L. S.; SANTOS, M. O.; PRADO, A. J.; ALDAYA, I.; "A new DSP-based physical layer encryption technique applied to passive optical networks", 20th International Conference on Transparent Optical Networks - ICTON 2018, 4 pp., 1 a 5 de julho de 2018, Bucarest, Romênia.
- [32] MOREIRA, A. C.; PRADO, A. J.; GRILO, C. V. C.; LUCA, C. C. S.; RONCHINI, C. M. B.; MENZINGER, J. S.; PEREIRA, T. G.; "Modelo físico de linhas de transmissão com ramos adicionais para introdução da influência da frequência", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [33] MOREIRA, A. C.; PRADO, A. J.; PEREIRA, T. G.; "Modelo físico de linhas de transmissão com ramos adicionais para introdução da influência da frequência", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.

- [34] LUCA, C. C. S.; PRADO, A. J.; RONCHINI, C. M. B.; "Análises de transitórios eletromagnéticos em linhas de transmissão utilizando métodos numéricos", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [35] GRILO, C. V. C.; PRADO, A. J.; "Rotinas numéricas em ambiente Python para simulação de transitórios eletromagnéticos", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [36] RONCHINI, C. M. B.; PRADO, A. J.; LUCA, C. C. S.; "Simulações de fenômenos transitórios eletromagnéticos em redes elétricas utilizando resistências de amortecimento", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [37] MENZINGER, J. S.; PRADO, A. J.; FERREIRA, A. A.; "Modelo físico de linhas de transmissão com componentes de alto fator de qualidade", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [38] PEREIRA, T. G.; PRADO, A. J.; "Testes de laboratorio utilizando o modelo de circuito modificado", XXX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2018), 22 de agosto de 2018, São Joao da Boa Vista, SP.
- [39] JUS, L. H.; SANTOS, M. O.; PEREIRA, T. G.; MOREIRA, A. C.; PRADO, A. J.; "Electromagnetic transients effect on blocks of circuits with damping resistance", XII Latin-american Congress on Electricity Generation and Transmission - CLAGTEE 2017, CD-ROM, 5 pp., 12 a 15 de novembro de 2017, Mar del Plata, Argentina.
- [40] SANTOS, M. O.; JUS, L. H.; PEREIRA, T. G.; MOREIRA, A. C.; PRADO, A. J.; "Introduction of damping resistances in simulations of electromagnetic transients using alternate structures", XII Latin-american Congress on Electricity Generation and Transmission - CLAGTEE 2017, CD-ROM, 5 pp., 12 a 15 de novembro de 2017, Mar del Plata, Argentina.
- [41] PEREIRA, T. G.; MOREIRA, A. C.; SANTOS, M. O.; JUS, L. H.; FERREIRA, A. A.; PRADO, A. J.; "Laboratory tests with model based on p circuits considering the introduction of branches composing by RL parallel elements", XII Latin-american Congress on Electricity Generation and Transmission - CLAGTEE 2017, CD-ROM, 5 pp., 12 a 15 de novembro de 2017, Mar del Plata, Argentina.
- [42] MOREIRA, A. C.; PEREIRA, T. G.; JUS, L. H.; SANTOS, M. O.; FERREIRA, A. A.; PRADO, A. J.; "Physical model of transmission lines with additional branches to introduce the influence of the frequency", XII Latin-american Congress on Electricity Generation and Transmission - CLAGTEE 2017, CD-ROM, 5 pp., 12 a 15 de novembro de 2017, Mar del Plata, Argentina.
- [43] PEREIRA, T. G.; MOREIRA, A. C.; PRADO, A. J.; FERREIRA, A. A.; PISSOLATO FILHO, J.; "Laboratorial tests with transmission line model based on odified circuits", The 38th Progress

in Electromagnetics Research Symposium - PIERS 2017, 2017, CD-ROM, vol.1, 7 pp., 22-25 May, 2017, St. Petersburg, Russia.

- [44] SANTOS, M. O.; JUS, L. H.; PRADO, A. J.; GENNARO, E. M.; PISSOLATO FILHO, J.; "Influence of damping resistance in electromagnetic transients using alternate structures of circuits", The 38th Progress in Electromagnetics Research Symposium - PIERS 2017, 2017, CD-ROM, vol.1, 8 pp., 22-25 May, 2017, St. Petersburg, Russia.
- [45] MOREIRA, A. C.; PRADO, A.J.; PEREIRA, T. G.; SCHEUER, R.; TRPCHEVA, M.; "Modelo físico de linhas de transmissão com ramos adicionais para introdução da influência da frequência", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.
- [46] GRILO, C. V. C.; PRADO, A. J.; "Rotinas numéricas em ambiente Python para simulação de fenômenos transitórios eletromagnéticos", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.
- [47] MENZINGER, J. S.; SCHEUER, R.; TRPCHEVA, M.; PRADO, A. J.; FERREIRA, A.A.; "Modelo físico de linhas de transmissão com componentes de alto fator de qualidade", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.
- [48] SANTOS, M. O.; PRADO, A. J.; JÚS, L.H.; "Influência da resistência de amortecimento em transitórios eletromagnéticos utilizando estruturas alternadas", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.
- [49] JÚS, L.H.; PRADO, A. J.; SANTOS, M. O.; "Transitórios eletromagnéticos em blocos de circuitos com resistência de amortecimento", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.
- [50] PEREIRA, T. G.; PRADO, A. J.; MOREIRA, A. C.; "Testes de laboratório com modelo de circuitos com introdução de ramos RL", XXIX Congresso de Iniciação Científica da UNESP (CIC-UNESP 2017), 25 de setembro de 2017, São Joao da Boa Vista, SP.