



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
CENTRO MULTIDISCIPLINAR DE PAU DOS FERROS - CMPF
BACHARELADO EM CIÊNCIA E TECNOLOGIA

GISLANDYA SUELLY BANDEIRA SILVA

**DESENVOLVIMENTO DE UMA TOOLBOX NO MATLAB PARA O AUXÍLIO
DO PROCESSO DE ENSINO-APRENDIZAGEM DOS MÉTODOS NUMÉRICOS**

Pau dos Ferros - RN

2021

GISLANDYA SUELLY BANDEIRA SILVA

**DESENVOLVIMENTO DE UMA TOOLBOX NO MATLAB PARA O AUXÍLIO
DO PROCESSO DE ENSINO-APRENDIZAGEM DOS MÉTODOS NUMÉRICOS**

Monografia apresentada como requisito parcial para obtenção do título de Bacharela em Ciência e Tecnologia pela Universidade Federal Rural do Semi-Árido, UFERSA.

Orientador: Prof. Bel. Gabriel Caldas Barros e Sá.

Pau dos Ferros - RN

2021

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

S586d Silva, Gislandya Suelly Bandeira.

DESENVOLVIMENTO DE UMA TOOLBOX NO MATLAB PARA
O AUXÍLIO DO PROCESSO DE ENSINO-APRENDIZAGEM DOS
MÉTODOS NUMÉRICOS / Gislandya Suelly Bandeira
Silva. - 2021.
61 f.: il.

Orientador: Gabriel Caldas Barros e Sá.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Ciência e Tecnologia,
2021.

1. Métodos Numéricos. 2. Zeros das Funções. 3.
Sistemas Lineares. 4. MATLAB. 5. Informática na
Educação. I. Sá, Gabriel Caldas Barros e, orient.
II. Título.

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

GISLANDYA SUELLY BANDEIRA SILVA

**DESENVOLVIMENTO DE UMA TOOLBOX NO MATLAB PARA O AUXÍLIO DO
PROCESSO DE ENSINO-APRENDIZAGEM DOS MÉTODOS NUMÉRICOS**

Monografia apresentada para a
Universidade Federal Rural do Semi Árido
como requisito para a obtenção do título
de Bacharel em Ciência e Tecnologia.

Pau dos Ferros, 27 de maio de 2021.

BANCA EXAMINADORA

**GABRIEL CALDAS
BARROS E
SA:10953941426**

Assinado de forma digital por
GABRIEL CALDAS BARROS E
SA:10953941426
Dados: 2021.05.27 15:41:31 -03'00'

Prof. Bel. Gabriel Caldas Barros e Sá (UFERSA)
Presidente

**ADLLER DE OLIVEIRA
GUIMARAES:076142
16458**

Assinado de forma digital por
ADLLER DE OLIVEIRA
GUIMARAES:07614216458
Dados: 2021.05.27 13:28:53 -03'00'

Prof. Dr. Ádller de Oliveira Guimarães (UFERSA)
Membro Examinador

**REUDISMAM ROLIM DE
SOUSA:08392843495**

Assinado de forma digital por
REUDISMAM ROLIM DE
SOUSA:08392843495
Dados: 2021.05.27 15:25:58 -03'00'

Prof. Dr. Reudismam Rolim de Sousa (UFERSA)
Membro Examinador

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me guiado e me dado forças para conseguir realizar este projeto.

Agradeço aos meus pais Geralda e Leudimar por terem sido o meu porto seguro durante toda minha vida e aos meus irmãos que sempre estiveram ao meu lado diante de todas as adversidades. Sem vocês eu não teria chegado a lugar algum e não teria forças para ir atrás dos meus sonhos.

Agradeço ao meu querido orientador, prof. Bel. Gabriel Caldas Barros e Sá, por ter me aceitado e ajudado imensamente durante a realização deste trabalho. Obrigado pelo o seu apoio, paciência e ensinamentos durante o desenvolvimento deste trabalho.

Aos os amigos e colegas da faculdade, em especial, a Bárbara, Milla, Maria, Wisley, Mileno e João Paulo, que mesmo separados pela distância, estavam ao meu lado tornando esse momento mais divertido e leve.

Agradeço a Banca Examinadora, Prof. Dr. Reudismam Rolim de Sousa e Prof. Dr. Ádller de Oliveira Guimarães que se dispuseram a avaliar e colaborar para a aprimoração deste trabalho.

“Só se pode alcançar uma grande êxito
quando nos mantemos fiéis a nós
mesmos”

(Friedrich Nietzsche)

RESUMO

Nos cursos de exatas do ensino superior, é comum que alunos apresentem dificuldade em compreender os assuntos abordados em sala de aula devido ao nível de abstração que alguns componentes curriculares necessitam, como é o caso da disciplina de Cálculo Numérico. Dessa forma, a utilização de ferramentas computacionais e softwares educacionais tem se mostrado uma forte aliada no auxílio do processo do ensino-aprendizagem. Com o objetivo de contribuir na redução do problema em questão na disciplina de cálculo numérico, esse trabalho apresenta o desenvolvimento de uma *toolbox* de carácter educacional, denominada *Toolbox Métodos Numéricos*, no software MATLAB que busca auxiliar no ensino-aprendizagem ao permitir a solução de problemas e exibir o passo a passo da resolução dos mesmos. Para tal, foram implementados métodos para cálculo dos zeros das funções e para a resolução de sistemas lineares, bem como uma interface gráfica com o intuito de facilitar o uso da ferramenta. Os resultados obtidos neste trabalho foram satisfatórios, e atendeu todos os objetivos propostos. Por fim, a *toolbox* foi disponibilizada para o público em geral através da plataforma File Exchange da Mathworks.

Palavras-chave: Métodos Numéricos. Zeros das Funções. Sistemas Lineares. MATLAB. Informática na Educação.

ABSTRACT

In the higher education exact courses, it is common that students find difficulties to understand the subjects covered in their classes due to the level of abstraction that some subjects require, as is the case with numerical analysis courses. Thus, the use of computational tools and educational software has been proven a strong ally in assisting the teaching-learning process. Aiming to contribute in the reduction of the stated problem in numerical analysis courses, this work presents the development of an educational toolbox, called Toolbox Métodos Numéricos, in the software MATLAB. Therefore, methods for finding the roots of a function and for solving systems of linear equations were implemented, as well as a graphical interface in order to facilitate the use of the tool. The results obtained in this work were satisfactory, and met all the proposed objectives. Finally, the toolbox was made available to the general public through Mathworks' File Exchange platform.

Keywords: Numeric Methods. Zeros of Functions. Linear Systems. MATLAB. Informatics in Education.

LISTA DE FIGURAS

Figura 1 - Tela Inicial do MATLAB.....	22
Figura 2 - Representação gráfica da raiz de uma função	26
Figura 3 - Representação gráfica do método da Bisseção	29
Figura 4 - Representação gráfica do método de Newton-Raphson.....	30
Figura 5 - Representação gráfica do método da Secante	31
Figura 6 - Algoritmo para implementação do método da Bisseção	37
Figura 7 - Algoritmo para implementação do método da Newton-Raphson	37
Figura 8 – Algoritmo para implementação do método da Secante	38
Figura 9 - Algoritmo para implementação do método da Eliminação de Gauss	38
Figura 10 - Algoritmo para implementação do método da Decomposição LU.....	39
Figura 11 - Menu de seleção do método	42
Figura 12 - Tela inicial (Zeros da Função)	43
Figura 13 - Segunda guia da aplicação	44
Figura 14 - Tabela expandida para o método de Newton-Raphson	44
Figura 15 - Terceira guia da aplicação	45
Figura 16 - Guia do método da eliminação Gaussiana.....	46
Figura 17 - Guia do método da decomposição LU	46
Figura 18 - Primeira guia do método de Newton-Raphson.....	47
Figura 19 - Segunda guia do método de Newton-Raphson.....	48
Figura 20 - Primeira guia do método da Bisseção.....	48
Figura 21 - Segunda guia do método da Bisseção.....	49
Figura 22 - Primeira guia do método da Secante	50
Figura 23 - Segunda guia do método da Secante	50
Figura 24 - Terceira guia com o gráfico contínuo	51
Figura 25 - Terceira guia com o gráfico numérico	51
Figura 26 - Terceira guia com o gráfico geral	52
Figura 27 - Aplicativo para resolver sistemas por eliminação de Gauss	53
Figura 28 - Aplicativo para resolver sistemas por decomposição LU	54

LISTA DE TABELAS

Tabela 1 - Comparação das formas alternativas de ambientes para interface gráfica	23
Tabela 2 - Informações solicitadas em cada método	36
Tabela 3 - Componentes gráficos utilizados no aplicativo	40

LISTA DE GRÁFICOS

Gráfico 1 - Impactos positivos sobre o uso de tecnologias digitais para a educação.	20
---	----

LISTA DE ABREVIATURAS E SIGLAS

GUIDE	<i>Graphical User Interface Design Environment</i>
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
PISA	Programa Internacional de Avaliação de Alunos
POO	Programação Orientada à Objetos
TIC	Tecnologias da Informação e Comunicação
UFERSA	Universidade Federal Rural do Semi-Árido
UFJF	Universidade Federal de Juiz de Fora
UFMG	Universidade Federal de Minas Gerais
UNIVESP	Universidade Virtual do Estado de São Paulo
USP	Universidade de São Paulo

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Objetivos	17
1.1.1	Objetivo Geral	17
1.1.2	Objetivos Específicos	17
1.1.3	Organização do trabalho	17
2	REFERENCIAL TEÓRICO.....	19
2.1	Informática na educação	19
2.2	MATLAB	21
2.2.1	<i>App Designer</i>	22
2.3	Métodos numéricos.....	24
2.3.1	Zero da função	25
2.3.1.1	CrITÉrios de Parada.....	27
2.3.1.2	Método da Bissecção	28
2.3.1.3	Método de Newton-Raphson.....	29
2.3.1.4	Método da Secante	30
2.3.2	Sistemas lineares	32
2.3.2.1	Eliminação Gaussiana.....	32
2.3.2.2	Decomposição LU	34
3	TRABALHOS RELACIONADOS	35
4	METODOLOGIA	36
4.1	Métodos Numéricos no MATLAB.....	36
4.2	Escolha das Funcionalidades	39
4.3	Componentes	40
4.4	Codificação	41
5	RESULTADOS E DISCUSÕES	42
5.1	Aplicativos	42
5.2	Resultados da Implementação dos métodos.....	47
5.2.1	Newton-Raphson.....	47
5.2.2	Bissecção.....	48
5.2.3	Secante	49
5.2.4	Gráfico.....	50
5.2.5	Eliminação Gaussiana.....	52
5.2.6	Decomposição LU	53

6	CONSIDERAÇÕES FINAIS	55
	REFERÊNCIAS.....	56

1 INTRODUÇÃO

Com o desenvolvimento da tecnologia e a facilidade de acesso à internet nos últimos anos, a disseminação do conhecimento passou a ser cada vez mais rápida e presente no cotidiano das pessoas. Decorrente do processo de inclusão tecnológica nas mais diversas áreas da sociedade, a educação vivencia cada vez mais o desenvolvimento de um cenário voltado à utilização de ferramentas computacionais e pesquisas inovadoras e motivadoras. Essas ferramentas devem proporcionar ao aluno uma imersão mais atrativa e esclarecedora sobre os assuntos abordados dentro da sala de aula. Contudo, esta proposta de ensino em conjunto com a tecnologia ainda enfrenta certos obstáculos que precisam ser vencidos para acabar com o estigma de que a ponte para conhecimento adquirido pelo aluno é feita apenas pelo professor (VALENTE, 2002; VIEIRA et al., 2019).

No Brasil, os estudantes dos cursos de exatas do ensino superior apresentam dificuldades na compressão e absorção de conhecimento, isso se deve a vários fatores que podem estar entrelaçados tanto da realidade social do aluno como a formação precária do ensino básico (NEVES, 2016). Segundo Machado (1988), esta dificuldade apresentada no ensino das ciências exatas, especialmente, da matemática, também pode estar relacionada ao fato de ser uma ciência que tem praticamente todo conteúdo voltado para o campo da abstração e o enfoque em aspectos formais, se desvincilhando da realidade que aluno está acostumado.

Essas dificuldades se tornam ainda mais evidentes nas disciplinas de exatas que necessitam de conhecimento prévio, onde os discentes apresentam esse déficit de conhecimento proveniente ainda do ensino básico. A análise realizada por Corso e Dorneles (2010) com cerca de 79 alunos matriculados entre a 3ª e 6ª série do ensino fundamental, verificou que a disciplina que os alunos apresentaram maior dificuldade foi a de matemática. Já no ensino médio, segundo o relatório divulgado pelo Programa Internacional de Avaliação de Alunos (PISA), os alunos brasileiros de 15 e 16 anos estão abaixo do nível 2 em conhecimentos matemáticos, classificando o país na classificação 58 dos 65 países analisados (PISA, 2018).

As dificuldades citadas refletem diretamente na evasão no ensino superior, como na Universidade de São Paulo (USP), onde os cursos de exatas apresentaram em média 79,88% de evasão entre os anos de 1998 e 2008, enquanto entre 2009 e 2014, dados do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio

Teixeira (INEP) mostram que os cursos de Engenharia Metalúrgica e Engenharia Mecânica registraram uma taxa de evasão próxima de 90% e 60%, respectivamente (NEVES, 2016; SACCARO; FRANÇA; JACINTO 2019; HOED, 2016). Já um estudo realizado na Universidade Federal de Juiz de Fora (UFJF), mostrou que a taxa de reprovação geral entre os anos de 2009 a 2013 é 54,4% nas disciplinas do primeiro ciclo do curso de Bacharelado Interdisciplinar em Ciências Exatas (CIRIBELLI, 2013).

Os componentes curriculares dos cursos das ciências exatas geralmente compreendem as áreas de física, cálculo, química e programação, uma dessas disciplinas é o Cálculo Numérico (também chamado de Análises Numéricas ou Métodos Numéricos), que objetifica estudar as técnicas numéricas para a obtenção de soluções aproximadas para problemas reais (CUNHA; CASTRO, 2010; FERNANDES et al., 2019). No curso de Engenharia Elétrica oferecido pela Universidade Federal de Minas Gerais (UFMG), entre 2010 e 2017, 25,86% dos discentes matriculados foram reprovados ou realizaram o trancamento do componente, assim como possuíam uma probabilidade de evasão do curso de 72% após o discente ser reprovado na disciplina (ARAÚJO; ALVES, 2018). Na UFJF a taxa de reprovação desta disciplina é de 40%, onde este índice pode estar aliado ao conhecimento prévio que o discente deve ter através das disciplinas já cursadas, sendo elas algoritmos, cálculo I, geometria analítica e álgebra linear (CIRIBELLI, 2013). Na Universidade Federal Rural do Semi-Árido (UFERSA), a taxa de insucesso na disciplina de algoritmos, que é um dos principais requisitos para a disciplina de cálculo numérico seja paga com sucesso, é mais de 50%. Dessa forma, as ferramentas computacionais e o projeto “Pré-Algoritmos”, desenvolvido para alunos que iram pagar ou já estavam matriculados na disciplina, foram fundamentais para que esse índice diminuísse nos semestres atuais (SOUSA et al., 2020).

O uso de ferramentas computacionais é uma das formas de amenizar as dificuldades apresentadas pelos alunos, e objetivam diminuir a forma sistemática passiva do ensino tradicional, sendo uma mudança associada a popularização das Tecnologias da Informação e Comunicação (TIC's) e utilização da sala de aula invertida, permitindo que o professor acompanhe e oriente o aluno a realizar o que foi aprendido nas aulas teóricas em ambientes educacionais, como softwares, sites, aplicativos etc., aumentando a aprendizagem ativa durante a vida acadêmica (ZABALA; PARKER; VIEIRA, 2017; LUCKE; KEYSSNER; DUNN 2013).

Um exemplo de programa computacional que pode ser utilizado é o MATLAB, uma plataforma destinada à análise de dados, desenvolvimento de algoritmos e criação de modelos matemáticos (MATLAB, 2021).

Tendo em vista a problemática abordada, o objetivo desta pesquisa é desenvolver uma *toolbox* no software MATLAB, intitulada “Toolbox Métodos numéricos”, para o cálculo dos zeros das funções reais utilizando métodos iterativos da Bisseção, de Newton-Raphson e da Secante, bem como os métodos diretos da eliminação gaussiana e decomposição LU para resolução de sistemas lineares, sendo essa ferramenta totalmente em português, e capaz de auxiliar no processo do ensino-aprendizagem da disciplina de cálculo numérico, ao apresentar o passo a passo da resolução e os gráficos para determinados problemas.

Para desenvolver a ferramenta e decidir quais métodos devem ser implementados, foi realizada um estudo bibliográfico a respeito dos métodos numéricos, e, em seguida, através da ferramenta *App Designer* do MATLAB, a aplicação foi desenvolvida de forma a exibir o passo a passo das resoluções e os gráficos das funções informadas no cálculo dos zeros das funções e as matrizes obtidas no cálculo da decomposição LU e a eliminação Gaussiana.

1.1 Objetivos

Nesta seção são apresentados o objetivo geral e os objetivos específicos desta pesquisa.

1.1.1 Objetivo Geral

Este trabalho tem como propósito a criação de uma ferramenta no ambiente de desenvolvimento para interfaces gráficas do MATLAB, chamado de *App Designer*, que seja capaz de calcular os zeros das funções reais e resolver sistemas lineares através de métodos numéricos, bem como exibir alguns gráficos e o passo a passo da resolução, com o objetivo de auxiliar o processo do ensino-aprendizagem dos discentes na disciplina de cálculo numérico.

1.1.2 Objetivos Específicos

A fim de atingir o objetivo geral, os objetivos específicos foram definidos e estão listados a seguir:

- I. Definir as funcionalidades que o aplicativo vai oferecer ao usuário, como o passo a passo da resolução, gráficos e tabelas;
- II. Estudar e delimitar os métodos a serem implementados, sendo eles os métodos da Bisseção, de Newton, da Secante, Eliminação Gaussiana e Decomposição LU.;
- III. Realizar o estudo da linguagem utilizada no ambiente de desenvolvimento e as funções disponibilizadas pelo software;
- IV. Disponibilizar a *toolbox* para o público em geral.

1.1.3 Organização do trabalho

Este trabalho está organizado da seguinte forma: No Capítulo 2, o referencial teórico é disposto e dividido em três partes de maior relevância: a Seção 2.1 analisa a informática na educação; a Seção 2.2 explana sobre o software MATLAB e o ambiente *App Designer* e a Seção 2.3 faz uma breve explanação sobre os Métodos

numéricos, sendo este último ainda dividido em sete subseções: a Seção 2.3.1 aborda os zeros de uma função, composta pela Subseção 2.3.1.1, que apresenta os critérios de parada, e pelas subseções 2.3.1.2, 2.3.1.3 e 2.3.1.4, que tratam do método da Bisseção, de Newton-Raphson e da Secante, respectivamente. A Subseção 2.3.2 explana sobre os sistemas lineares e se divide em outras duas subseções: a Subseção 2.3.2.1 fala sobre a Eliminação Gaussiana e a 2.3.2.2 explica o método da Decomposição LU. No Capítulo 3 é tratado dos trabalhos relacionados a este, enquanto o Capítulo 4 apresenta a metodologia, estando dividido em quatro seções: a Seção 4.1 justifica a escolha do tema através de uma breve revisão bibliográfica, a 4.2 expõe o estudo dos métodos numéricos no MATLAB, a Seção 4.3 apresenta as principais funcionalidades selecionadas, a Seção 4.4 trata da utilização das componentes gráficas utilizadas no desenvolvimento da interface e a Seção 4.5 trata do processo de codificação do aplicativo. O Capítulo 5 expõe os resultados e discussões obtidos durante este trabalho onde, na Seção 5.1, os aplicativos desenvolvidos são exibidos, enquanto a Seção 5.2 apresenta os resultados da implementação dos métodos através de exemplos. Por fim, o Capítulo 6 dispõe das considerações finais e propostas para trabalhos futuros.

2 REFERENCIAL TEÓRICO

Este capítulo dispõe sobre a revisão e análise do que a literatura apresenta a respeito dos métodos numéricos e o software MATLAB.

2.1 Informática na educação

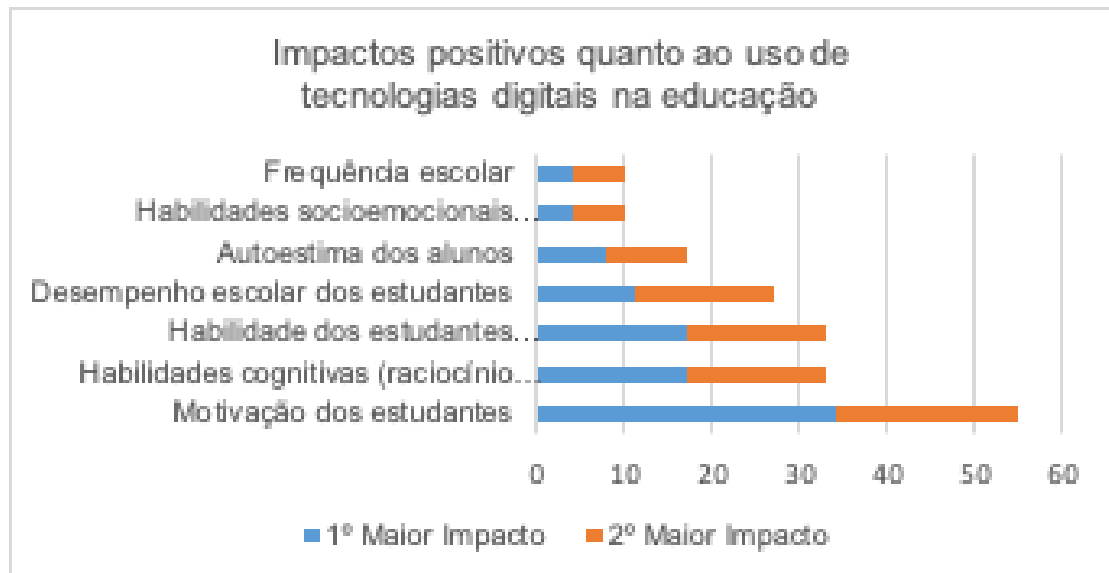
Com o advento e o desenvolvimento exponencial da tecnologia a partir da década de 1940 e sua inserção em todas as áreas sociais, corroborou para que a sociedade precisasse de pessoas que tivessem o conhecimento sobre ela, mesmo que de forma superficial, assim, às instituições de ensino passaram a ter o papel de formar os indivíduos para essa nova realidade. Como a escola sempre esteve a par com as novas formas de ensino e tecnologias, herdou-se a responsabilidade de formar cidadãos críticos e criativos em relação ao uso dessas tecnologias. A utilização da informática na educação traz aos alunos novas formas de se comunicar, pensar, aprender e também de ensinar. E seu uso deve ser visto como um recurso que ajude ao professor a repassar seus conhecimentos. (BADALOTTI, 2017, p. 53-54).

No Brasil, a inserção da tecnologia como parte do dia a dia de uma sala de aula ainda possui certas barreiras, pois o modelo educacional presente ainda é muito centrado na ideia de que professor está ali para passar instruções e de forma passiva estas informações são repassadas para os alunos (SOBRAL, 2019). No entanto, segundo Moran (2000), o processo de ensino-aprendizagem depende da curiosidade e iniciativa do aluno, pois um ambiente em que os alunos estão motivados é mais fácil a agregação de conhecimento durante o processo.

A utilização de multimídias e ferramentas computacionais está cada vez mais presente no ambiente escolar e a incorporação da tecnologia dentro de sala de aula traz inúmeras vantagens no processo de ensino-aprendizagem, pois o professor funciona como mediador, o aluno consegue ter uma participação mais ativa. Através dessa forma de ensino o aluno desperta mais interesse sobre os temas e assuntos, recebe as instruções repassadas e consegue transformar e incorporar na prática estas informações em conhecimento (VALENTE, 2014). No Gráfico 1, é mostrado as principais vantagens que o uso destas ferramentas traz para a aprendizagem e motivação do aluno, primeiro analisando o maior impacto positivo e o segundo maior

impacto positivo, sobre o uso e melhorias das tecnologias digitais (SOARES; ALVES; STOLFI, 2020).

Gráfico 1 - Impactos positivos sobre o uso de tecnologias digitais para a educação.



Fonte: Soares, Alves e Stolfi (2020).

Na situação atual devido a pandemia da COVID-19, diversas áreas da sociedade tiveram que se adaptar ao novo modo de trabalho, por conta da quarentena que perdura até este momento em diversos países. O ensino remoto passou a ser, predominantemente, a única forma de ensino, assim as ferramentas computacionais ganharam o papel de protagonistas neste cenário. O uso destas tecnologias tenta minimizar o impacto sobre o ensino-aprendizagem dos alunos e facilitar o compartilhamento de conhecimentos. Essas tecnologias tem como ônus o fato de oferecer uma variedade de recursos, se comparado a ensino presencial, onde se limita tradicionalmente ao quadro ou a apresentações (CANGANE, 2021).

De acordo com Sassano *et al.* (2021), no seu estudo sobre a satisfação de alunos com as ferramentas computacionais utilizadas nas disciplinas de exatas, que são vistas nos cursos de Engenharia da Produção e Engenharia da Computação foram ministradas remotamente na Universidade Virtual do Estado de São Paulo (UNIVESP), mostrou que a grande maioria dos estudantes tiveram a percepção de que melhoram e se sentiram mais motivados nas disciplinas que fizeram uso destes recursos.

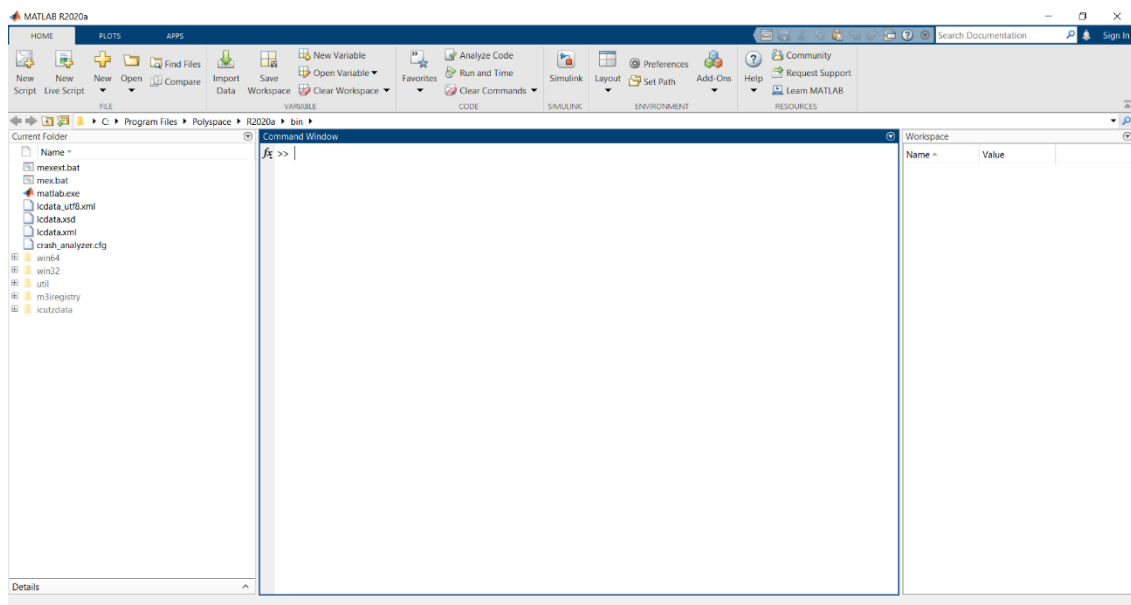
2.2 MATLAB

O software MATLAB, cujo nome é a abreviatura para “MATrix LABoratory”, é um programa matemático de alto desempenho, usado para desenvolver algoritmos e facilitar cálculos complexos através de sua linguagem voltada para a computação técnica. Devido a isso e ao fato de o software oferecer diversas funcionalidades, boa parte dos usuários são, principalmente, engenheiros, matemáticos e cientistas, sendo também muito usado para pesquisas na academia (SALVADOR, 2016). De acordo com a Mathworks, desenvolvedora do software, o MATLAB é capaz de realizar análises numéricas, cálculos matriciais, processamento de sinais, construção de gráficos e visualização e construção de computação numérica. Os cálculos são realizados de forma otimizada em comparação com outras linguagens de programação e são expressos de forma semelhante à sua escrita matemática. O MATLAB pode ser usado para as mais diversas finalidades, seja computacional, visualização ou ainda como ambiente de programação.

A concepção inicial para a criação do MATLAB foi do professor Clever Moler, que lecionava matérias de matemática e ciência da computação na Universidade do Novo México. O objetivo de Moler era poder facilitar para seus alunos a implementação de cálculos com matrizes e computação matemática, pois ele considerava a linguagem Fortran complexa para realizar estas finalidades. (SALVADOR, 2016).

Na tela inicial do software é apresentado o *Command Window*, que é o espaço reservado para o usuário realizar operações matemáticas, chamar funções, visualizar dados, etc. O usuário também pode criar *scripts* com a linguagem MATLAB, cuja sintaxe é similar a das linguagens de programação C e C++. Quando o algoritmo ou a operação é iniciada, todas as variáveis, matrizes, células e dados são guardados no *Workspace* e podem ser acessados através de comandos básicos. A Figura 1 apresenta a tela inicial em seu *layout* padrão. Outra funcionalidade que o MATLAB oferece é a criação de aplicativos e *toolboxes* dentro da própria plataforma. Inicialmente, o ambiente de desenvolvimento utilizado para esta finalidade era o GUIDE (*Graphical User Interface Design Environment*), porém, a partir da versão 2016a o *App Designer* foi introduzido, trazendo mais funcionalidades e aprimoramentos em comparação com seu antecessor que futuramente será descontinuado (MathWorks).

Figura 1 - Tela Inicial do MATLAB.



Fonte: Autor(es).

2.2.1 App Designer

A ferramenta *App Designer* permite desenvolver aplicativos com interface gráfica de maneira rápida e interativa. São disponibilizados para o desenvolvedor diversos componentes, como botões, menus e tabelas, que permitem a interação entre os usuários e o programa desenvolvido, o desenvolvimento da interface e a interação dos elementos se dá por meio do sistema “*drag and drop*” (arrastar e soltar), ou ainda por codificação dos mesmos. Essa comunicação entre as ações realizadas pelo usuário sobre os elementos gráficos e a resposta do programa é baseada no paradigma da Programação Orientada à Objetos (POO) (LOPES; RANGEL; MARTHA, 2019).

Pode-se criar interfaces gráficas através do software MATLAB de três maneiras distintas, sendo elas pelo *App Designer*, como já citado, programaticamente ou pelo ambiente GUIDE. De acordo com Lopes, Rangel e Martha (2019), cada técnica tem pontos positivos e pontos negativos, nessas comparações o *App Designer*, como mostra a Tabela 1, tem potencial melhor com relação aos demais em virtude de ser o ambiente mais atual para projetar interfaces gráficas e pelo fato de que a cada versão do MATLAB, ele ganha novas funcionalidades ou atualizações dos componentes.

Tabela 1 - Comparação das formas alternativas de ambientes para interface gráfica

Ambiente	Pontos Positivos	Pontos Negativos
<i>App Designer</i>	Ambiente gráfico interativo integrado com código; Formalização de código seguindo POO; Manuseio fácil e intuitivo; Atualizações e avanços no futuro.	Ainda em desenvolvimento; Não está otimizado, assim aplicações complexas durante a execução podem ficar lentas.
GUIDE	Ambiente gráfico interativo para manuseio de componentes gráficos; Relativamente eficiente; Maior extensão de referências e fóruns online para buscar ajuda.	Código menos formal para o contexto de programação estruturada; Menos opções de elementos gráficos; Sem perspectiva de avanço devido a descontinuação.
Código	Maior controle de posição e comportamento dos componentes gráficos em cada diálogo;	Código deve ser fornecido para desenvolver todos os aspectos da interface gráfica;

	Maior controle da interação entre diferentes GUI's; Maior liberdade ao desenvolvedor.	Leva-se mais tempo para desenvolver a interface.
--	---	--

Fonte: Adaptado de Lopes, Rangel e Martha (2019).

Devido às vantagens oferecidas pelo *App Designer*, se optou pela utilização do mesmo, para criar a ferramenta que é o objetivo deste trabalho.

2.3 Métodos numéricos

Ao longo da história surgiram problemas matemáticos em que não se podia obter uma solução exata ou a sua aplicação não era conveniente devido a sua complexidade. Desse modo, estudiosos desenvolveram métodos capazes de resolver estes problemas através de aproximações sucessivas. Os babilônios desenvolveram há mais de 3700 anos técnicas para encontrar a solução de equações quadráticas, para calcular aproximações de raízes quadradas de um número inteiro e também usavam a interpolação para calcular os juros compostos. Na dinastia Han, há 2000 anos, e a dinastia Sung (960-1279), há manuscritos que mostram a utilização de métodos semelhantes a eliminação de Gauss e a obtenção de soluções de equações de grau superior, respectivamente. Neste milênio, matemáticos como Horner, Runge Kutta, Napier, Newton, Kepler, Taylor, Fibonacci, Leibniz, entre outros contribuíram significativamente com os seus estudos a respeito de resoluções para problemas de soluções aproximadas (MOURA et al, 2005).

Atualmente, os métodos numéricos são descritos como a transformação de um problema real em uma formulação matemática, utilizando uma técnica que proporciona que sua resolução seja possível via operações aritméticas e lógicas, podendo ser referida também como matemática computacional, pois é muito aplicada no processo de resolução de problemas nos computadores digitais (ZULYADAINI, 2020; CHAPRA, 2013).

Os métodos numéricos podem ser iterativos ou diretos e consistem na aplicação de algoritmos para resolver problemas matemáticos através de técnicas que facilitam a sua resolução. Os métodos iterativos produzem uma sequência de passos que podem ou não convergir para a solução aproximada dos problemas; já nos métodos diretos, a solução é definida com um número fixo de passos e se direciona para a solução exata (ARAÚJO, 2017; MOURA et al, 2005).

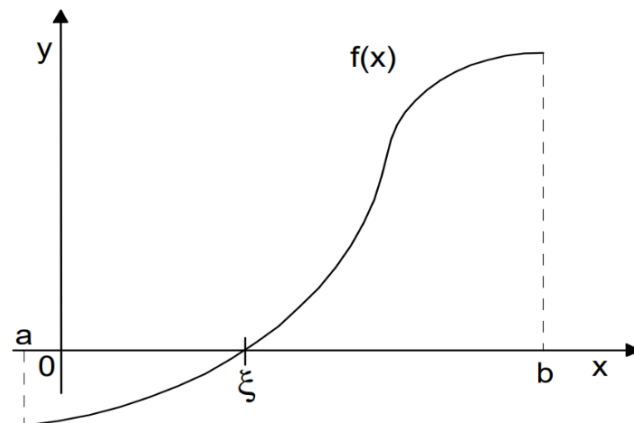
Dentre as várias áreas de estudo dos métodos numéricos, podemos citar o cálculo do zero das funções, a resolução de sistemas lineares, a derivação e integração numérica e a resolução de equações diferenciais ordinárias de forma numérica (RUGGIERO; LOPES, 1997). Alguns exemplos reais da aplicação desses métodos podem ser vistos em detonações celulares bidimensionais em curvas de tubos lisos (LI; REN; NING, 2013) e na maximização da receita líquida pela frequência de limpezas em módulos de energia fotovoltaica (KHATCHATOURIAN; TRETER, 2021).

Os métodos numéricos abordados neste trabalho são referentes às técnicas utilizadas para calcular os zeros das funções reais, Subseção 2.3.1, e resolver sistemas de equação lineares, como descrito na Subseção 2.3.2.

2.3.1 Zero da função

Um assunto importante na área de estudo da matemática é a raiz ou zero de uma função. Segundo Lobão, Sanches e Furlan (2017), para uma função real $f(x)$ definida em um intervalo $[a, b]$, a raiz ou zero da função em $[a, b]$ é todo $\xi \in (a, b)$ de tal maneira que ao aplicar o valor de ξ , temos $f(\xi) = 0$. Graficamente, o zero de uma função real é definido como sendo o valor da abscissa dos pontos da curva que intercepta o eixo $\overrightarrow{Ox}$, como mostra a Figura 2.

Figura 2 - Representação gráfica da raiz de uma função



Fonte: Lobão, Sanches e Furlan (2017).

De acordo com Cherri *et al.* (2017), para se obter raízes de uma equação qualquer de forma numérica, podem ser utilizados métodos iterativos. Esses métodos necessitam pelo menos de um valor inicial que ao ser aplicado permite encontrar uma sequência de soluções aproximadas do valor que converge para a solução desejada. Ainda segundo Cherri *et al.* (2017), o procedimento para encontrar as raízes das funções reais é dividido em duas fases:

- Fase 1: isolamento da raiz, que consiste em localizar um intervalo $[a, b]$ que contém a raiz, utilizando a análise teórica e gráfica da função $f(x)$, este isolamento é aplicado em métodos que necessitam de intervalo, caso contrário, apenas um valor é “chutado” de tal maneira que seja suficiente próximo a raiz a ser calculada.
- Fase 2: refinamento do intervalo, que consiste na aplicação dos algoritmos dos métodos numéricos e tem como objetivo convergir para solução aproximada de maneira sucessiva.

Para restringir a quantidade de iterações e definir um ponto de convergência utilizam-se os critérios de parada de acordo com o grau de exatidão desejado (CAMPOS, 2007; RUGGIERO; LOPES, 1997; PIRES, 2015).

Dentre os métodos numéricos iterativos para o cálculo dos zeros das funções reais, este trabalho aborda o da Bisseção, Newton-Raphson e Secante, que estão

descritos mais detalhadamente nas subseções 2.3.1.2, 2.3.1.3 e 2.3.1.4, respectivamente.

2.3.1.1 Critérios de Parada

Por meio dos métodos numéricos é possível encontrar aproximações dos valores reais dos zeros de uma função. Ao utilizar métodos iterativos, faz-se necessário determinar quando há convergência do valor calculado, para isso são utilizados os critérios de parada. O critério de parada deve avaliar o quanto o valor calculado está próximo da raiz exata com base na tolerância desejada. Caso o critério não seja atendido, pode-se dizer que não existe aproximação que satisfaça as condições impostas e, então, a execução do método deve ser interrompida (RUGGIERO; LOPES, 1997).

Os critérios de parada podem ser a análise do valor da função, o erro absoluto, o erro relativo e os limites do intervalo, e sua representação matemática é dada pelas Equações (1), (2), (3) e (4), respectivamente. Porém, como os métodos são implementados computacionalmente, faz-se necessário que além dos critérios matemáticos, o número máximo de iterações desejadas também seja definido como um critério de parada a fim de reduzir o esforço computacional (RUGGIERO; LOPES, 1997).

Os critérios de parada são relacionados com a tolerância, que é um valor determinado de acordo com o problema. Os métodos abordados nas seções 2.3.1.2, 2.3.1.3 e 2.3.1.4 são iterativos e necessitam do intervalo a e b , que corresponde aos valores inicial e final, respectivamente, ou apenas de um valor inicial. Assim, a cada iteração esses valores são atualizados, correspondendo às variáveis x_i e x_{i-1} em (1), (2) e (3) (RUGGIERO; LOPES, 1997; FRANCO, 2006).

$$|f(x_i)| < \epsilon \quad (1)$$

$$|x_i - x_{i-1}| < \epsilon \quad (2)$$

$$\frac{x_i - x_{i-1}}{x_i} < \epsilon \quad (3)$$

$$\frac{b-a}{2} < \epsilon \quad (4)$$

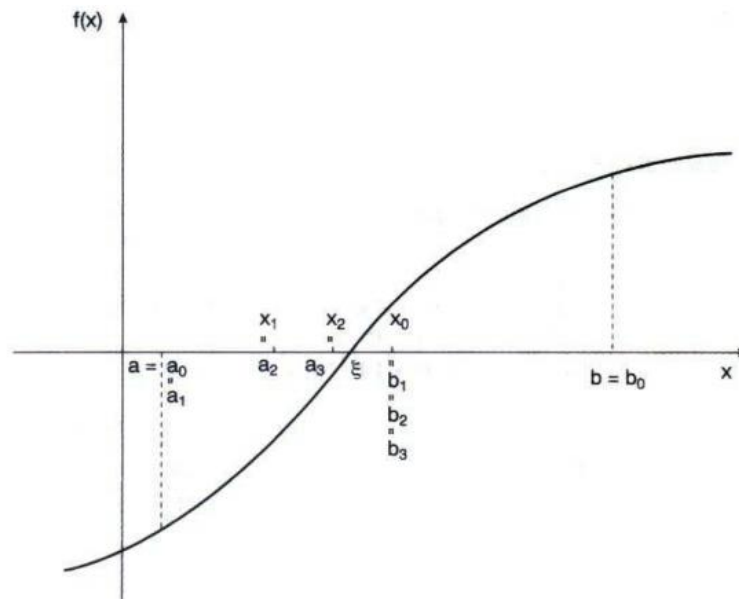
2.3.1.2 Método da Bissecção

O método da Bissecção é baseado no Teorema de Valor Imediato, garantindo a existência de uma solução para $f(x) = 0$ no intervalo escolhido, desde que a função seja contínua nesse intervalo e satisfaça a expressão $f(a) \cdot f(b) < 0$, que é conhecido como Teorema de Cauchy. Para tal, faz-se uso da redução do intervalo que contém a raiz, levando em consideração a divisão dos limites superior e inferior ao meio sucessivas vezes, enquanto não atingir a quantidade máxima de iterações requeridas, encontrar o valor da raiz ou satisfazer o critério de parada especificado (NASCIMENTO, 2015).

Matematicamente, Ruggiero e Lopes (1997) descreve que é necessário definir os valores iniciais de x_k e x_{k+1} , que no primeiro momento, representam o ponto inicial e final, respectivamente, bem como o critério de parada desejado. A partir desse momento, o novo x_k é calculado sucessivamente de acordo da Equação (5), sendo esse processo realizado até que $f(x_k) \cong 0$, onde x_k é a aproximação da raiz de $f(x)$, até que o critério de parada seja satisfeito ou o número máximo de iterações seja atingido, como demonstra visualmente a Figura 3.

$$x_k = \frac{x_k + x_{k+1}}{2} \quad (5)$$

Figura 3 - Representação gráfica do método da Bissecção



Fonte: Ruggiero e Lopes (1997).

Uma das desvantagens do método da bissecção é o maior esforço computacional requerido quando comparado com outros métodos numéricos (RUGGIERO; LOPES, 1997).

2.3.1.3 Método de Newton-Raphson

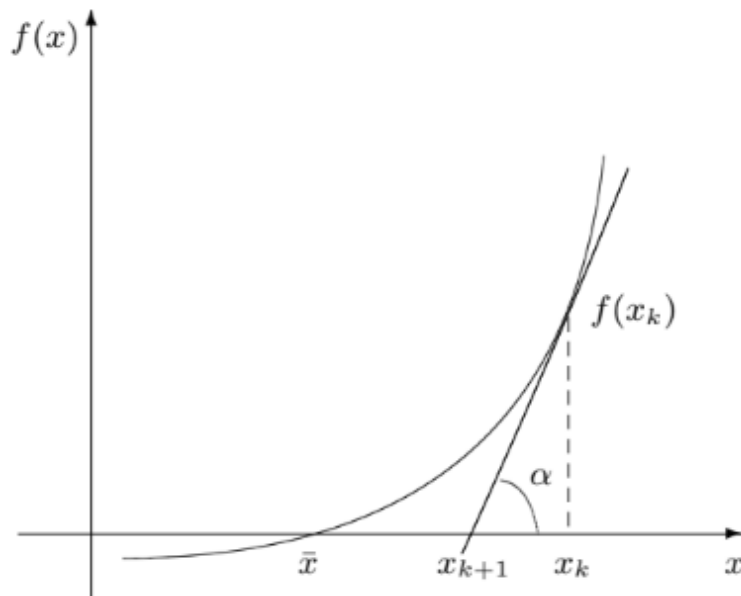
Também conhecido como método das tangentes, esse modelo matemático foi proposto no século XVII em uma junção das pesquisas de Isaac Newton, publicada em seu livro "Método de Fluxões", e Joseph Raphson. Entretanto, o método de Joseph é considerado mais simples e, com as contribuições de Newton, o mesmo se tornou apto a ser utilizado em qualquer função real conhecida, se tornando um trabalho mútuo de ambas personalidades (SANTOS *et al.*, 2018a).

Esse método é entendido como um aprimoramento particular para o método do Ponto Fixo, permitindo a aproximação para a solução de uma função não linear a partir de um valor inicial, utilizando gradientes para conseguir gerar uma sequência de valores até que a aproximação almejada seja alcançada (SOUZA *et al.*, 2018; FRANCO, 2006). Desse modo, cada valor de x é uma aproximação para a raiz da função escolhida, obtido através da Equação (6), assim como mostra a Figura 4.

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (6)$$

Onde $f(x)$ é a função, $f'(x)$ representa sua respectiva derivada, x_k a aproximação da anterior da raiz e x_{k+1} é a aproximação atual da raiz. No entanto, as desvantagens desse método são a necessidade de se conhecer a derivada da função e o fato de a mesma ter que possuir um valor não-nulo. Apesar disso, esse método é um dos mais indicados para calcular os zeros das funções reais (RUGGIERO; LOPES, 1997).

Figura 4 - Representação gráfica do método de Newton-Raphson



Fonte: Franco (2006).

2.3.1.4 Método da Secante

Visando contornar a limitação da necessidade de derivação do método de Newton-Raphson, pode-se usar o método da Secante, que substitui a derivada pela utilização das retas secantes da função, também conhecido como quociente da diferença (SOUZA et al., 2020; LOPES et al., 2018), conforme a Equação (7).

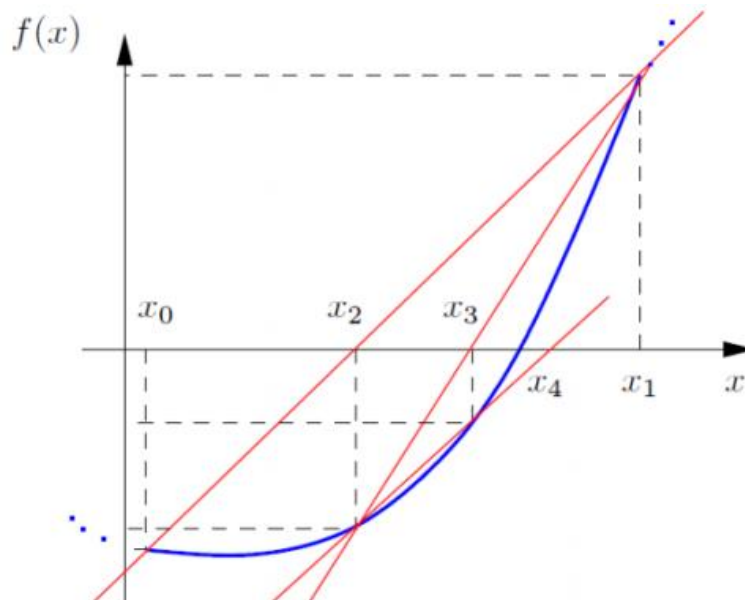
$$f'(x_k) \cong \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \quad (7)$$

Onde x_k e x_{k-1} são duas aproximações para a raiz x e $f'(x_k)$ representa a aproximação da derivada para os valores aplicados na função (FRANCO, 2006). Com isso, aplicando a Equação (7) em (5), obtém-se a formulação matemática para o método da Secante, como mostrado na Equação (8), sendo necessário definir o intervalo cujos extremos são os dois valores iniciais para x_k e x_{k-1} para, então, iniciar o cálculo da raiz a partir desse método (RUGGIERO; LOPES, 1997).

$$x_{k+1} = \frac{x_{k-1} \cdot f(x_k) - f(x_{k-1}) \cdot x_k}{f(x_k) - f(x_{k-1})} \quad (8)$$

A execução desse método é bastante similar ao método da Bissecção, mas obtendo sua convergência superlinear ≈ 1.168 , relacionando-se com as retas secantes da função. Para a resolução, é necessário definir o critério de parada e o intervalo x_0, x_1 (TEIXEIRA; SILVA, 2018; RUGGIERO; LOPES, 1997). A Figura 5 apresenta uma representação gráfica do funcionamento do método da Secante.

Figura 5 - Representação gráfica do método da Secante



Fonte: Ruggiero e Lopes (1997).

consequentemente, menos tempo de execução. Sua aplicação é baseada no teorema I, descrito abaixo (RUGGIERO; LOPES, 1997; GILAT; SUBRAMANIAM, 2008):

Teorema I:

- (i) Trocar a posição de duas equações;
- (ii) Multiplicar uma equação por uma constante diferente de zero;
- (iii) Adicionar um múltiplo de uma equação por uma outra equação.

Para exemplificar este teorema, podemos utilizar uma matriz genérica A de tamanho 3x3.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \end{cases} \quad (10)$$

Etapa 1: Transformar o sistema de equações (10) em uma matriz aumentada do A | b.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \vdots & b_1 \\ a_{21} & a_{22} & a_{23} & \vdots & b_2 \\ a_{31} & a_{32} & a_{33} & \vdots & b_3 \end{bmatrix} \quad (11)$$

Etapa 2: Aplicar o teorema (I) na Equação (11), até se obter a matriz triangular superior que tenha solução equivalente à matriz original, como mostra a Equação (12).

$$\begin{bmatrix} a'_{11} & a'_{12} & a'_{13} & \vdots & b'_1 \\ 0 & a'_{22} & a'_{23} & \vdots & b'_2 \\ 0 & 0 & a'_{33} & \vdots & b'_3 \end{bmatrix} \quad (12)$$

Etapa 3: Através de substituições regressivas e manipulações matemáticas encontrar o valor que satisfaça $A \cdot x = b$ a partir da matriz triangular superior $A' \cdot x = b'$ (RUGGIERO; LOPES, 1997).

2.3.2.2 Decomposição LU

A decomposição LU é um método muito usado para resolver sistemas de equações lineares. Este método é muito útil para calcular a matriz inversa de um sistema. Em termos de tempo computacional e da quantidade de operações matemáticas, o método LU é bem eficiente, outra vantagem é que para realização dos cálculos precisamos apenas da matriz A, pelo menos, até o passo de substituir os valores, na qual utilizamos o vetor solução b . É importante frisar que o vetor b não é alterado no decorrer do cálculo (GILAT; SUBRAMANIAM, 2008).

Neste método, o primeiro passo é fatorar a matriz A em um produto de duas outras matrizes.

$$A = L \cdot U \quad (13)$$

Onde a matriz L (do inglês, *Lower*) é uma matriz do tipo triangular inferior e a matriz U (do inglês, *Upper*) é do tipo triangular superior. Assim, temos que o sistema de equações tem a forma matricial:

$$L \cdot U \cdot x = b \quad (14)$$

Nesse sistema, temos que resolver a Equação (14):

$$U \cdot x = y \quad (15)$$

Primeiramente, deve-se resolver a Equação (15) em termos de y , depois que encontramos o seu valor, substituímos em (16) para encontrar o valor de x .

$$L \cdot x = b \quad (16)$$

Há duas maneiras que podem ser aplicadas para resolver a decomposição LU, a primeira é a própria eliminação de Gauss. A matriz U é o resultado final da matriz de coeficientes, a Matriz L, entretanto, não tem a forma explícita, mas seus valores são dados no processo. A outra forma de se resolver é utilizando o método de Crout,

que decompõe a matriz A em duas matrizes L e U respectivamente, onde a matriz U tem os elementos de sua diagonal iguais a 1 (GILAT; SUBRAMANIAM, 2008).

3 TRABALHOS RELACIONADOS

Este capítulo dispõe sobre resumos de alguns trabalhos científicos que possuem relação com esta pesquisa.

O trabalho apresentado por Góes, Silva, Guedes e Silva (2015), tem como objetivo a implementação de uma interface desenvolvida no MATLAB com o foco para resoluções de métodos numéricos. Os autores destacam que os métodos numéricos são a alternativa mais prática e econômica para resolver problemas que por métodos analíticos sua resolução é complexa ou não há solução. O autor cita ainda que a utilização de ambientes computacionais, como é o caso da MATLAB, pode otimizar os cálculos, trazendo precisão e velocidade. Para a criação da interface foi utilizada a ferramenta GUIDE, que assim como *App Designer* é um editor interativo em que se pode criar e editar interfaces. Os métodos numéricos abordados pelo trabalho são, raízes, derivada (Método da Diferença Central, Progressiva e Regressiva), integração (Método do Trapézio, o Método de Simpson e o Método de Simpson Repetido), aproximações de funções (Interpolação) e sistemas lineares (com a limitação de aceitar no máximo uma matriz 3×3).

Em sua monografia, Moura (2017), desenvolveu, também através do ambiente GUIDE uma interface para resolução de métodos numéricos iterativos. Os métodos implementados foram o da Bissecção, de Newton-Raphson, da Secante e iteração linear. O foco deste projeto é tornar as aulas mais atrativas para o discente e que ele possa assimilar as simulações com as resoluções de exemplos feitas pelo professor na sala de aula, tornando o processo de aprendizagem mais fácil. O aplicativo disponibilizado permite ao usuário visualizar, a função informada e as suas respectivas raízes aproximadas no gráfico e verificar os dados na componente tabela, da raiz e da função calculada aplicada no ponto.

No trabalho realizado por Paulista *et al.* (2018), foi desenvolvido o software RaizCalc, que utiliza a linguagem C para calcular os zeros das funções polinomiais via método da bissecção. O software mostra os valores obtidos a cada iteração e mostra os intervalos que contém a raiz, para que o usuário possa obter o resultado

final da raiz aproximada, é necessário que ele insira o intervalo que contém a raiz encontrado na primeira etapa.

Os trabalhos citados nessa seção não demonstram o passo a passo da resolução (MOURA, 2017), não implementaram os métodos de Newton-Raphson e Secante (PAULISTA *et al.*, 2018), não permite a inserção de uma matriz maior do que uma 3x3 ou foi feita em um ambiente que futuramente será descontinuado (GÓES; SILVA; GUEDES; SILVA, 2015). Baseado nisso, esta produção acadêmica visa desenvolver uma ferramenta no *App Designer* que apresente funcionalidades que outros trabalhos desenvolvidos referentes ao tema não tinham ou eram trabalhados de maneira não tão aprofundada, como é o caso do passo a passo, a tabela com todos os dados calculados e poder inserir uma matriz de qualquer tamanho.

4 METODOLOGIA

Este capítulo apresenta as etapas para o desenvolvimento deste trabalho de forma detalhada.

4.1 Métodos Numéricos no MATLAB

Antes de iniciar o desenvolvimento da ferramenta no *App Designer*, foi necessário a implementação computacional dos métodos a serem utilizados para verificar os resultados gerados, para isso, utilizando o software MATLAB na versão R2020a, foram programados 5 *scripts*. Os três primeiros calculam o valor aproximado do zero de uma função real via método da bissecção, da secante e Newton-Raphson. Os outros dois referem-se à eliminação Gaussiana e decomposição LU, respectivamente. As figuras 6, 7, 8, 9 e 10 apresentam os pseudocódigos referentes aos métodos da Bissecção, Newton-Raphson, da Secante, Eliminação Gaussiana e decomposição LU, respectivamente. A Tabela 2 apresenta as variáveis descritas no pseudocódigo.

Tabela 2 - Informações solicitadas em cada método

Método	Variáveis
Newton-Raphson	N, t, x

Bisseção	N, t, x_0, x_1
Secante	N, t, x_0, x_1
Eliminação de Gauss	A, b
Decomposição LU	A, b

Fonte: Autor(es).

Figura 6 - Algoritmo para implementação do método da Bisseção

```

Algoritmo "metodo_Bissecacao"

entradas
  N: inteiro    // numero maximo de iteracoes
  x0, x1: real  // intervalo
  t: real       // tolerancia
saidas
  x1: real      // raiz
inicio
  i=0
  enquanto i <= N faca
    xk = (x0 + x1)/2
    se (f(xk) == 0 OU (x1-x0)/2 < t) entao
      // raiz encontrada
      interrompa
    fimse
    se f(x0)*f(x1) < 0 entao
      x1 = xk
    senao
      x0 = xk
    fimse
    i = i + 1
  fimenquanto
  se i > N entao
    // nao foi possivel encontrar a raiz
  fimse
fimalgoritmo

```

Fonte: Adaptado de Ruggiero e Lopes (1997).

Figura 7 - Algoritmo para implementação do método da Newton-Raphson

```

Algoritmo "metodo_Newton_Raphson"

entradas
  N: inteiro // numero maximo de iteracoes
  x: real    // estimativa inicial
  t: real    // tolerancia
saidas
  xk: real   // raiz
inicio
  i=0
  enquanto i <= N faca
    xk = x - f(x)/f'(x)
    se (abs((x-xk)/xk)<t) entao
      // raiz encontrada
      interrompa
    fimse
    x = xk
    i = i + 1
  fimenquanto
  se i > N entao
    // nao foi possivel encontrar a raiz
  fimse
fimalgoritmo

```

Fonte: Adaptado de Ruggiero e Lopes (1997).

Figura 8 – Algoritmo para implementação do método da Secante

```

Algoritmo "metodo_Secante"

entradas
    N: inteiro      // numero maximo de iteracoes
    x0, x1: real    // intervalo
    t: real         // tolerancia
saidas
    x1: real        // raiz
inicio
    i=0
    enquanto i <= N faca
        xk = (x0*f(x1) - x1*f(x0)) / (f(x1)-f(x0))
        se (abs((xk-x1)<t) entao
            // raiz encontrada
            interrompa
        fimse
        x0 = x1
        x1 = xk
        i = i + 1
    fimenquanto
    se i > N entao
        // nao foi possivel encontrar a raiz
    fimse
fimalgoritmo

```

Fonte: Adaptado de Ruggiero e Lopes (1997).

Figura 9 - Algoritmo para implementação do método da Eliminação de Gauss

```

Algoritmo "Eliminação_Gaussiana"

entradas
    A: matriz[ i...linhas, j...colunas ] de reais; //matriz de coeficientes
    b: matriz[ i...linhas, j...colunas ] de reais; //matriz resposta
saidas
    B: matriz[ i...linhas, j...colunas ] de reais; //matriz equivalente
inicio
    a: matriz[A;b]
    i = 0
    k = 0
    j = 0
    para k = 1, ..., (n-1) faça
        para i = (k+1), ..., n faça
            m = a(ik)/a(kk)
            a(ik) = 0
            para j = (k+1), ..., n faça
                a(ij) = a(ij) - m*a(kj)
            fimpara
            b(i) = b(i) - m * b(k)
        fimpara
    fimpara

// resolução do sistema triangular superior
x(n) = b(n)/a(nn)
para i = (n-1), ..., 1 faça
    soma = 0
    para j = (i+1), ..., n faça
        soma = soma + a(ij)*x(j)
    fimpara
    x(i) = (b(i) - soma)/a(ij)
fimpara
fimalgoritmo

```

Fonte: Adaptado de Ruggiero e Lopes (1997).

Figura 10 - Algoritmo para implementação do método da Decomposição LU

Algoritmo “Decomposição_LU”

```

entradas
    A: matriz[ i...linhas, j...colunas ] de reais;    //matriz de coeficientes
    b: matriz[ i...linhas, j...colunas ] de reais;    //matriz resposta
saídas
    L: matriz[ i...linhas, j...colunas ] de reais;    //matriz Inferior L
    U: matriz[ i...linhas, j...colunas ] de reais;    //matriz Superior U
Inicio
    a = A
    j = 0
    k = 0
    para j = 1, ..., (n-1) faça
        pivo = a(jj)
        para i = (j+1), ..., n faça
            fator = a(ij)/pivo
            L(ij) = fator
            a(i:,)= a(i:,-)fator a(j,:)
        fimpara
    fimpara
    U = a
fimalgoritmo

```

Fonte: Adaptado de Ruggiero e Lopes (1997).

4.2 Escolha das Funcionalidades

Uma vez que os *scripts* foram criados, as funcionalidades da ferramenta foram definidas com base nos dados de entrada que cada método necessita e dos objetivos que o aplicativo deve atender. Tais funcionalidades estão descritas a seguir:

- Entrada de dados (a depender do método deve-se receber dados escalares ou vetoriais);
- Exibir os resultados em uma tabela;
- Mostrar o passo a passo da resolução (de forma analítica ou por instruções);
- Mostrar o gráfico, caso o método implementado permita, para se obter uma melhor visualização dos resultados;
- Comunicação entre as telas da ferramenta.

Devido ao tempo disponível para realização deste trabalho, se optou por apenas mostrar o gráfico com todos os valores calculados até a raiz final aproximada.

4.3 Componentes

De acordo com as funcionalidades definidas e com as interações que devem ocorrer entre o usuário e a ferramenta, alguns componentes gráficos disponibilizados pelo próprio *App Designer* que permitem visualização, entrada ou saída de dados que foram utilizados. A Tabela 2 apresenta a lista de componentes para os métodos iterativos (usados no cálculo dos zeros das funções) e os diretos (usados na solução de sistemas lineares).

Tabela 3 - Componentes gráficos utilizados no aplicativo

Componentes	Métodos Diretos	Métodos Indiretos
<i>Edit Field (Numeric)</i>		✓
<i>Edit Field (Text)</i>	✓	✓
<i>Table</i>	✓	✓
<i>Panel</i>	✓	✓
<i>Text Area</i>	✓	✓
<i>Push Button</i>	✓	✓
<i>Drop Down</i>	✓	✓
<i>Label</i>	✓	✓
<i>Tab Group</i>		✓
<i>Axes</i>	✓	✓
<i>Radio Button</i>		✓
<i>Image</i>		✓
<i>Menu Bar</i>	✓	✓
<i>Context Menu</i>	✓	

Fonte: Autor(es).

4.4 Codificação

Depois de definir a ferramenta, o ambiente e as funcionalidades, foi iniciada a codificação. No *App Designer*, os códigos podem ser escritos no *Code View* em três tipos de blocos: o de propriedades que é responsável por armazenar os dados dentro de variáveis globais; o bloco de métodos que contém as funções que são relacionadas as classes de objetos, como é o caso das componentes; e por fim, o bloco de eventos que é responsável por permitir a comunicação entre os objetos quando alguma informação é alterada (Mathworks). Assim, para se obter uma comunicação entre as componentes do aplicativo todos os blocos foram utilizados em seu desenvolvimento.

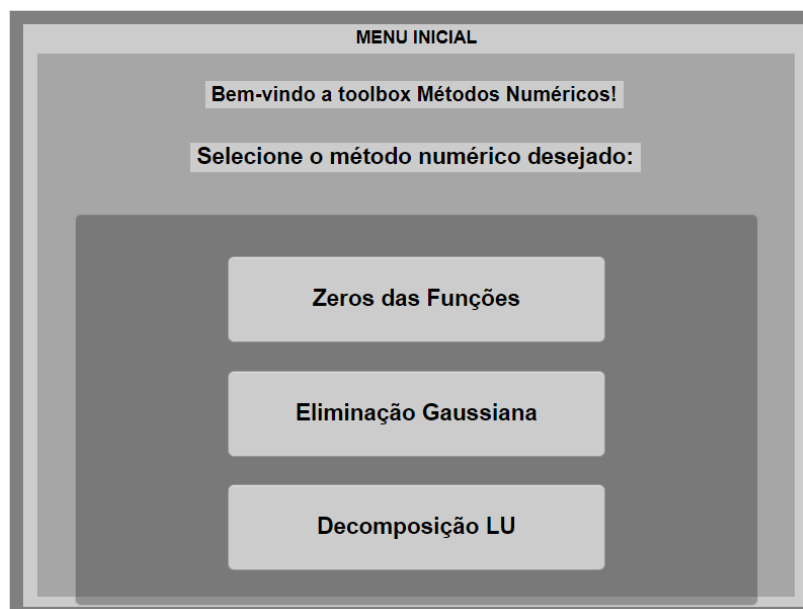
5 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados e discussões a respeito do desenvolvimento da *toolbox* Métodos Numéricos a partir dos métodos aplicados e que foi disponibilizada gratuitamente para acesso da comunidade pela plataforma *MATLAB Central File Exchange*¹. A ferramenta é composta por quatro aplicativos que se complementam durante a execução, sendo eles a tela para seleção do método, o aplicativo para cálculo dos zeros das funções, para o método da eliminação Gaussiana e, por fim, para decomposição LU. As equações que devem ser inseridas nos aplicativos desenvolvidos seguem a mesma sintaxe do programa MATLAB.

5.1 Aplicativos

Na tela de menu inicial são apresentados para o usuário três opções, onde ele pode escolher qual método deseja utilizar, conforme a Figura 11. Após selecionar o método desejado o usuário é redirecionado para o aplicativo correspondente.

Figura 11 - Menu de seleção do método

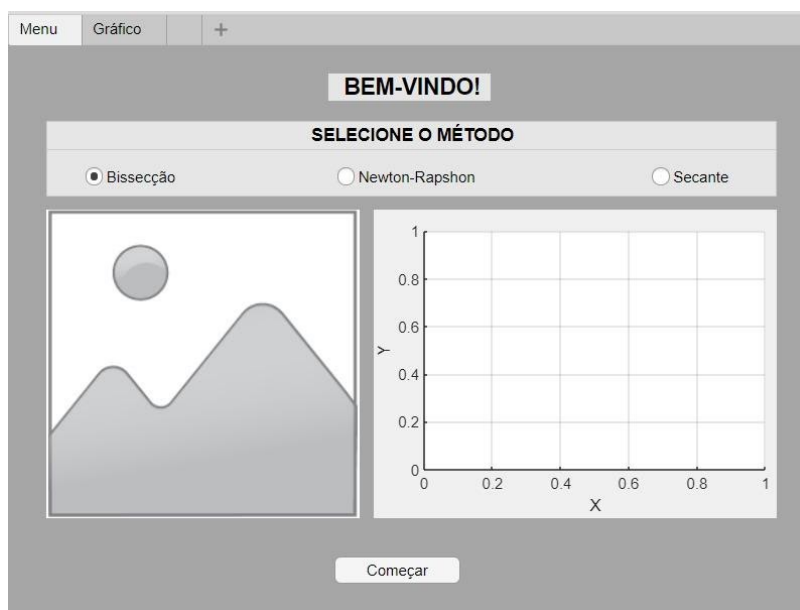


Fonte: Autor(es).

¹ <https://www.mathworks.com/matlabcentral/fileexchange/92673-toolbox-metodos-numericos>

Caso o usuário opte pela opção zeros das funções, a ferramenta abre um novo aplicativo e redireciona o usuário para a primeira guia do mesmo, que contém um menu onde o usuário pode escolher entre os métodos implementados, sendo eles o da Bisseção, de Newton-Raphson e da Secante. Ao selecionar um dos métodos citados, um texto e um gráfico informativos são exibidos, conforme a Figura 12. Ao clicar em **COMEÇAR**, a segunda guia do aplicativo é aberta.

Figura 12 - Tela inicial (Zeros da Função)

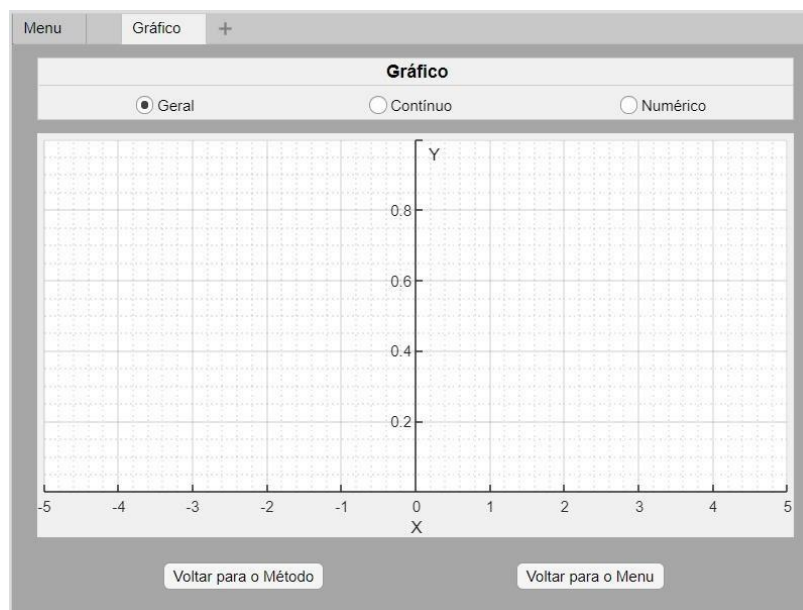


Fonte: Autor(es).

Na segunda guia, o usuário é solicitado a informar os dados de entradas de acordo com o método selecionado na guia 1, bem como o critério de parada que se deseja utilizar, conforme a Figura 13. Ao clicar no botão **CALCULAR**, o método selecionado é executado e realiza as iterações a fim de encontrar a raiz da função informada de acordo como os dados definidos, exibindo o passo a passo da resolução de forma textual e os resultados em forma de tabela. O botão **GRÁFICO** redireciona o usuário para a terceira guia, onde é possível visualizar o gráfico da função informada, e o botão **LIMPAR** deve ser selecionado cada vez que o usuário desejar informar novos dados de entrada. Na segunda guia há ainda os botões **EXPANDIR TABELA**, que expande a tabela dos resultados, mostrando informações mais completas sobre os cálculos a cada iteração, cujos dados variam de acordo com o

analítico da função e raiz calculada; numérico, exibindo o gráfico dentro do intervalo inicial informado, bem como a raiz; ou geral, exibindo tanto o gráfico numérico quanto o contínuo. O botão **VOLTAR PARA O MÉTODO** permite que o usuário retorne à segunda guia, enquanto **VOLTAR PARA O MENU** o redireciona para a tela inicial do aplicativo dos zeros das funções.

Figura 15 - Terceira guia da aplicação



Fonte: Autor(es).

Caso o usuário selecione a eliminação gaussiana na tela principal da toolbox, o respectivo aplicativo é chamado e apresenta a interface conforme a Figura 16. Os elementos presentes no aplicativo são: o botão de **CALCULAR**, que executa os cálculos para encontrar o sistemas de equações equivalentes a original; o botão **LIMPAR** que, assim como no aplicativo dos zeros das funções, deve ser selecionado sempre que o usuário deseje informar novos valores de entrada; um menu *Drop Down*, que permite que o usuário escolha qual matriz da etapa de resolução ele deseja visualizar as componentes de campo de texto A e b, que permitem que o usuário informe as matrizes dos coeficientes e da solução, respectivamente. Nos campos **MATRIZ EXPANDIDA** e **SOLUÇÃO**, o aplicativo exibe a matriz dos coeficientes e solução de acordo com a seleção no menu *Drop Down*, enquanto no campo **PASSO A PASSO**, todos os procedimentos para realização dos cálculos são apresentados.

Figura 16 - Guia do método da eliminação Gaussiana

Fonte: Autor(es).

De modo análogo, se a opção escolhida na tela principal for a decomposição LU, o aplicativo correspondente é chamado. A Figura 17 demonstra a tela visualizada para o este método. A sua interface e elementos utilizados são semelhantes aos do aplicativo da eliminação gaussiana, com a ressalva de que agora o *Drop Down* a depender da escolha do usuário mostra as matrizes L, U ou A na tabela.

Figura 17 - Guia do método da decomposição LU

Fonte: Autor(es).

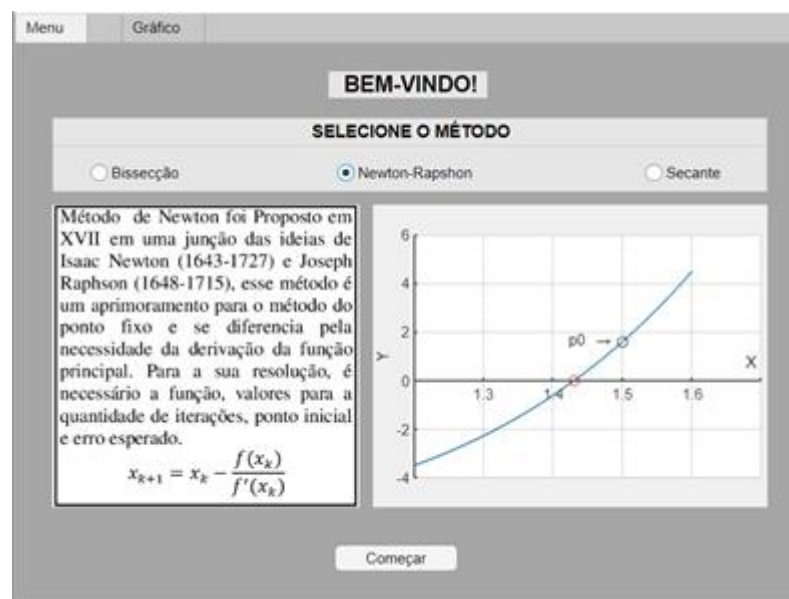
5.2 Resultados da Implementação dos métodos

Esta seção apresenta os resultados da implementação dos métodos numéricos iterativos da Bisseção, Newton-Raphson e da Secante.

5.2.1 Newton-Raphson

Caso o usuário escolha realizar o cálculo para encontrar o zero de uma função com do método de Newton-Raphson, a tela mostrada será a da Figura 18. Na segunda guia os valores necessários são solicitados. Neste caso, não é necessário que usuário informe a derivada da função já que o próprio aplicativo realiza o cálculo da derivação, conforme a Figura 19, que apresenta um exemplo com a função $f(x) = x^5 - 6$, uma tolerância de 0.01 e o critério de parada "análise do valor da função", com um máximo de 5 iterações e 1.5 como o valor inicial.

Figura 18 - Primeira guia do método de Newton-Raphson



Fonte: Autor(es).

Figura 19 - Segunda guia do método de Newton-Raphson

The interface is titled "MÉTODO DE NEWTON-RAPHSON". It has three tabs: "Menu", "Newton-Raphson", and "Gráfico". The "Newton-Raphson" tab is active.

ENTRADA DE DADOS

Função: x0: [Calcular]

Iterações: [Gráfico]

Tolerância: Critério: [Limpar]

Iteração

Iteração	Xn	f(Xn)	erro
1.0000	1.4370	0.1283	0.1283
2.0000	1.4310	0.0011	0.0011

[Expandir Tabela]

PASSO A PASSO

DADOS DE ENTRADA:

f(x) = x⁵-6

x0 = 1.5000 i = 05 tol = 0.0100

d(x0) = 25.3125 f(x0) = 1.5938

1ª Iteração

f(x0) = 1.5938 d(x0) = 25.3125

x1 = 1.5000 - (1.5938/25.3125)

[Voltar para o Menu]

Fonte: Autor(es).

5.2.2 Bissecção

Se o usuário optar por utilizar o método da bissecção, como na Figura 19, a segunda guia solicita o intervalo que compreende a possível raiz. Logo, considerando os mesmos dados da Figura 19, mas atribuindo x_0 como 1.2 e x_1 como 1.5, obtém-se a resolução demonstrada na Figura 21.

Figura 20 - Primeira guia do método da Bissecção

The interface is titled "BEM-VINDO!". It has two tabs: "Menu" and "Gráfico". The "Menu" tab is active.

SELECIONE O MÉTODO

☒ Bissecção ☐ Newton-Raphson ☐ Secante

O método da Bissecção é baseado no Teorema de Valor Imediato, que garante uma raiz para a função em um intervalo desde que a função seja contínua. Para sua resolução, é necessário solicitar os valores iniciais e finais (x_0 e x_1), assim como o valor máximo de iteração, erro esperado, função escolhida, sendo necessário a divisão de maneira sucessiva.

$$x_k = \frac{x_k + x_{k+1}}{2}$$

Gráfico

The graph shows a function $f(x) = x^5 - 6$ on a coordinate system. The x-axis ranges from -2.4 to -2.0, and the y-axis ranges from -6 to 4. The function is a straight line passing through the points $(-2.4, -6)$ and $(-2.0, 4)$. The root is marked at $x = -2.0$. The points p_0 and p_1 are marked on the x-axis at -2.4 and -2.2 respectively.

[Começar]

Fonte: Autor(es).

Figura 21 - Segunda guia do método da Bisseção

Menu Bisseção Gráfico

MÉTODO DA BISSECÇÃO

ENTRADA DE DADOS

Função: x0:

Iterações: x1:

Tolerância: Critério:

Iteração	Xn	f(Xn)	erro
1.0000	1.3500	-1.5160	1.5160
2.0000	1.4250	-0.1241	0.1241
3.0000	1.4625	0.6908	0.6908
4.0000	1.4438	0.2728	0.2728
5.0000	1.4344	0.0717	0.0717

PASSO A PASSO

DADOS DE ENTRADA:

$f(x) = x^5 - 6$

$x_0 = 1.2000$ || $i = 05$ || $f(x_0) = -3.5117$

$x_1 = 1.5000$ || $tol = 0.0100$ || $f(x_1) = 1.5938$

TEOREMA DE CAUCHY

$f(a) \times f(b) < 0$

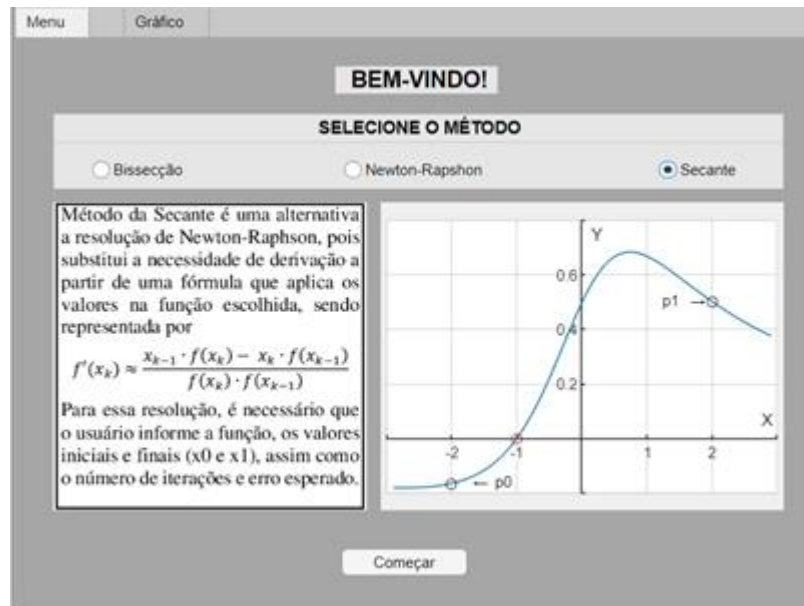
$-3.5117 \times 1.5938 < 0$

Fonte: Autor(es).

5.2.3 Secante

Caso seja escolhido o método da Secante na primeira guia, como demonstra a Figura 22, a segunda guia é atualizada e a ferramenta solicita as mesmas variáveis do método da bisseção. Contudo, a implementação do método é completamente distinta. Portanto, para efeito de demonstração, foram considerados os mesmos valores atribuídos na Figura 21, como demonstra a Figura 23.

Figura 22 - Primeira guia do método da Secante



Fonte: Autor(es).

Figura 23 - Segunda guia do método da Secante



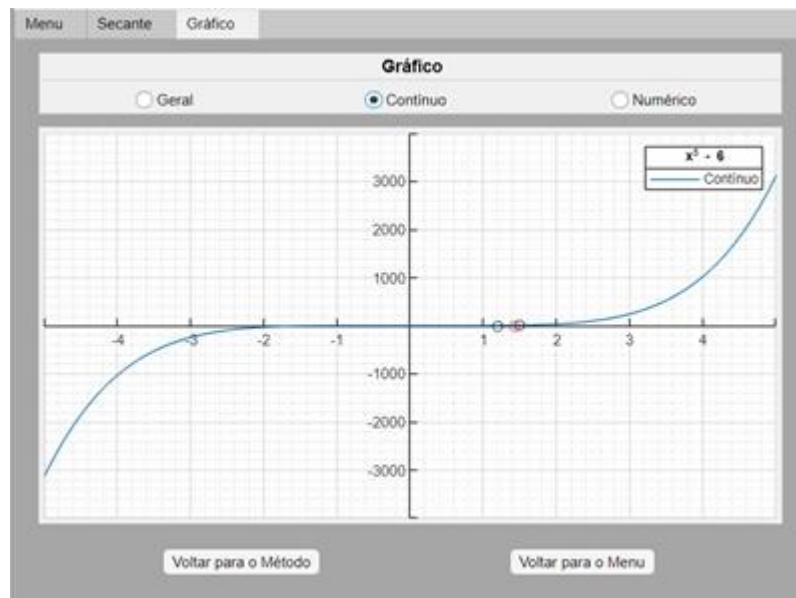
Fonte: Autor(es).

5.2.4 Gráfico

Ao clicar no botão **GRÁFICO** na primeira guia, a terceira guia da ferramenta é exibida, onde são apresentados os gráficos da função informada. Para a função de

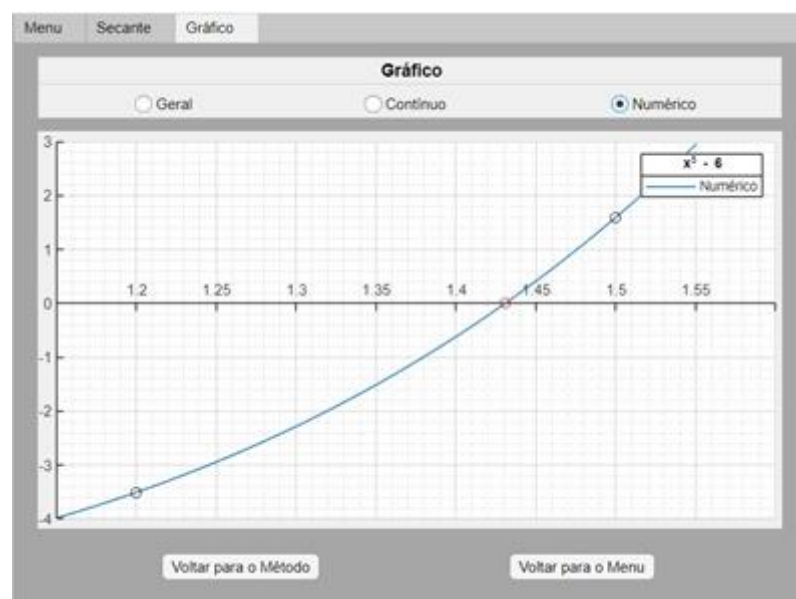
$x^5 - 6$, a Figura 24 exibe o gráfico contínuo, contendo as raízes e intervalo selecionado; a Figura 25 apresenta o gráfico numérico, com foco no intervalo e na raiz calculada; e a Figura 26 apresenta o gráfico geral, que exibe ambos os gráficos contínuo e numérico.

Figura 24 - Terceira guia com o gráfico contínuo



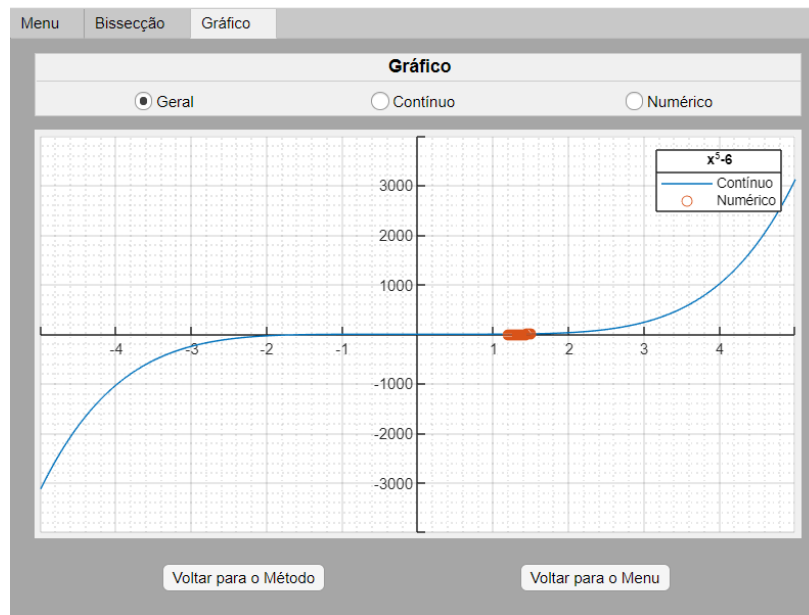
Fonte: Autor(es).

Figura 25 - Terceira guia com o gráfico numérico



Fonte: Autor(es).

Figura 26 - Terceira guia com o gráfico geral



Fonte: Autor(es).

5.2.5 Eliminação Gaussiana

Se o aplicativo da eliminação Gaussiana for chamado pelo menu na tela inicial, a interface disposta na Figura 27 é exibida. O usuário deve informar a matriz dos coeficientes e a matriz resposta nos campos de texto **A** e **b**, usando a sintaxe padrão do MATLAB, ou seja, os elementos da linha devem ser separados por vírgula ou por um espaço, enquanto as colunas devem ser separadas por ponto e vírgula, e todos os valores devem estar entre colchetes, as matrizes informadas podem ser de qualquer tamanho. Para exemplificar o funcionamento do aplicativo, foi utilizada a matriz exibida na Figura 27:

Figura 27 - Aplicativo para resolver sistemas por eliminação de Gauss

Menu

ELIMINAÇÃO DE GAUSS

ENTRADA DE DADOS:

A: [6,2,-1;2,4,1; 3,2,8] **Calcular**

b: [7,7,13] **Limpar**

Clique com o botão direito do mouse em cima do nome das componentes para obter informações **Matriz:** 1 ▼

MATRIZ EXPANDIDA				SOLUÇÃO
6	2	-1	7	1.0000
2	4	1	7	1.0000
3	2	8	13	1.0000

PASSO A PASSO

DADOS DE ENTRADA:

Matriz A = [6,2,-1;2,4,1; 3,2,8]
Matriz b = [7,7,13]

Matriz expandida: Matriz 1

1ª Iteração

$L2 \leftarrow L2 - L1 \times (a_{21}/a_{11})$

Matriz 2

$L3 \leftarrow L3 - L1 \times (a_{31}/a_{11})$

Matriz 3

2ª Iteração

$L3 \leftarrow L3 - L2 \times (a_{32}/a_{22})$

Matriz 4

Fonte: Autor(es).

5.2.6 Decomposição LU

Caso o usuário escolha o método da Decomposição LU no menu da tela inicial, ele será redirecionado para a interface disposta na Figura 28. O usuário, assim como no aplicativo da eliminação gaussiana, deve informar as matrizes dos coeficientes e resposta nos campos de texto **A** e **b**. O teste foi realizado com a mesma matriz da Figura 27. Um ponto a ser destacado, é que nem o método da eliminação Gaussiana e nem o método LU possui a funcionalidade de mostrar o gráfico das equações do sistema, desse modo, o aplicativo mostra apenas a resolução do problema.

Figura 28 - Aplicativo para resolver sistemas por decomposição LU

Menu

MÉTODO LU

ENTRADA DE DADOS:

A: [6,2,-1;2,4,1; 3,2,8]

Calcular

b: [7;7;13]

Limpar

Clique com o botão direito do mouse em cima do nome das componetes para obter informações

Matriz: Matriz L

MATRIZ			SOLUÇÃO
1.0000	0	0	1.0000
0.3333	1.0000	0	
0.5000	0.3000	1.0000	

PASSO A PASSO

DADOS DE ENTRADA:

Matriz A = [6,2,-1;2,4,1; 3,2,8]

Matriz b = [7;7;13]

1ª Iteração

$$U_{11} = A_{11} - (0) \times (L_1 \times U_1)$$

$$U_{11} = 6$$

$$L_{21} = A_{21} - (0) \times (L_2 \times U_1)$$

$$L_{21} = 0.3333$$

$$L_{31} = A_{31} - (0) \times (L_3 \times U_1)$$

$$L_{31} = 0.5$$

2ª Iteração

Fonte: Autor(es).

6 CONSIDERAÇÕES FINAIS

Este trabalho apresentou o desenvolvimento de uma *toolbox* no MATLAB, denominada “*Toolbox Métodos Numéricos*”, cujo objetivo é auxiliar no processo do ensino-aprendizagem da disciplina de cálculo numérico a partir da resolução dos métodos numéricos iterativos para cálculo dos zeros das funções reais (Bisseção, Newton-Raphson e Secante) e métodos diretos para resolução de sistemas lineares (Eliminação Gaussiana e Decomposição LU), sendo uma ferramenta totalmente em português e que exibe o passo de passo de resolução e os seus gráficos.

Os resultados foram satisfatórios e cumpriram os objetivos propostos, tendo em vista que a ferramenta foi capaz de implementar os métodos previstos e realizar as funcionalidades estabelecidos neste trabalho. A ferramenta foi disponibilizada na plataforma de compartilhamento de arquivos da MathWorks, a MATLAB *Central File Exchange*.

Para trabalhos futuros, recomenda-se o implementar mais funcionalidades gráficas para que permitam a visualização dos pontos a cada etapa do cálculo, bem como a implementação de mais métodos numéricos, tais como métodos de interpolação, integração numérica e solução de equações diferenciais ordinárias. Testes da ferramenta em sala de aula também podem ser realizados com a finalidade de validar a mesma no que diz respeito a melhoria do processo de ensino aprendizagem na disciplina de métodos numéricos. Desenvolver em outras plataformas para melhorar a portabilidade, como para Android e web.

É importante ressaltar que a *toolbox* desenvolvida neste trabalho tem como foco o auxílio do processo do ensino-aprendizagem dos métodos numéricos, portanto seu uso deve ser apenas para fins de estudo ou ensino.

REFERÊNCIAS

ARAÚJO, Eduardo. **Métodos numéricos para simulação na engenharia**. 2017. Disponível em: <https://www.esss.co/blog/metodos-numericos-para-simulacao-na-engenharia/>. Acesso em: 02 maio 2021.

ARAÚJO, Isabella Carolina; ALVES, Sandy Pinheiro. **Avaliação do desempenho acadêmico dos estudantes de graduação: Engenharia elétrica**. 2018. Disponível em: https://www.ufmg.br/prograd/arquivos/docs/RelatGrad/2018/avalDesempenhoEstudantesGrad_Presencial/EngenhariaEletrica. Acesso em: 09 abr. 2021.

BADALOTTI; Greisse Moser. **Educação e tecnologias**. UNIASSELVI, 2017. 179 p.: il.

BRASIL. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Brasil no Pisa 2018 [recurso eletrônico]. – Brasília: Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, 2020. 185 p.: il.

CANGAGE, Letícia. **Antes e depois da pandemia**: Como as ferramentas do ensino a distancia podem beneficiar o ensino universitário. USP - São Paulo. Disponível em: <https://www.poli.usp.br/noticias/36180-antes-e-depois-da-pandemia-como-as-ferramentas-do-ensino-a-distancia-podem-beneficiar-o-ensino-universitario.html>. Acesso em: 02 de junho de 2021.

CAMPOS, F.F., **Algoritmos Numéricos**, Rio de Janeiro: LTC, 2.ed., 2007.

CHAPRA, Steven C. **Métodos Numéricos Aplicados com MATLAB® para Engenheiros e Cientistas-3**. AMGH Editora, 2013.

CHERRI, Adriana et al. **CÁLCULO DE ZEROS DE FUNÇÕES REAIS**. 2017. Disponível: <http://www.fc.unesp.br/~adriana/Numerico/Funcoes.pdf>. Acesso em: 03 de maio de 2021.

CIRIBELLI, Bruno César de Nazareth. **RETENÇÃO E EVASÃO ESCOLARES NO BACHARELADO INTERDISCIPLINAR EM CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA**. 2013. 123 f. Dissertação (Mestrado) - Curso de Programa de Pós-Graduação Profissional em Gestão e Avaliação da Educação Pública, Centro de Políticas Públicas e Avaliação da Educação, Universidade Federal de Juiz de Fora, Juiz de Fora, 2015.

CORSO, Luciana Vellinho; DORNELES, Beatriz Vargas. Senso numérico e dificuldades de aprendizagem na matemática. **Rev. psicopedag.**, São Paulo, v. 27, n. 83, p. 298-309, 2010. Disponível em <http://pepsic.bvsalud.org/scielo.php?script=sci_arttext&pid=S0103-84862010000200015&lng=pt&nrm=iso>. acesso em 10 abr. 2021.

CUNHA, Francisco Gêvane Muniz; CASTRO, Jânio Kléo de Sousa. **Cálculo Numérico**. Fortaleza: UAB/IFCE, 2010. 163 p.

DERIVE. University of Hawai'i, 2012. Disponível em: <<http://math.hawaii.edu/wordpress/derive/>>. Acesso em: 10 abr. 2021.

FERNANDES, Amunike Jorge Moreira *et al.* APLICAÇÃO DE MÉTODOS NUMÉRICOS PARA SOLUÇÃO DE PROBLEMAS DE ENGENHARIA ECONÔMICA. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão**, v. 10, n. 1, 2019.

FRANCO, Neide Maria Bertoldi. **Cálculo Numérico**. São Paulo: Pearson Prentice Hall, 2006. 520 p.

GILAT, Amos; SUBRAMANIAM, Vish. **Métodos numéricos para engenheiros e cientistas**: uma introdução com aplicações usando o matlab. Porto Alegre: Bookman, 2008. 474 p.

GÓES, Duaymy B. R.; SILVA, Giovane M. da; GUEDES, Danielle L.; SILVA, Orlando Fonseca. Desenvolvimento de Interface em MATLAB para Aprendizado e Comparação de Métodos Numéricos. **Rev..Proceeding Series Of The Brazilian Society Of Applied And Computational Mathematics**, Natal, v. 3, n. 1, p. 1-2, 2005.

HOED, Raphael Magalhães. **Análise da evasão em cursos superiores**: o caso da evasão em cursos superiores da área de computação. 2016. 188 f. Dissertação (Mestrado) - Curso de Mestrado Profissional em Computação Aplicada, Departamento de Ciência da Computação, Universidade de Brasília, Brasília, 2016.

JI, Zhanglong et al. Differentially private distributed logistic regression using private and public data. *BMC medical genomics*, v. 7, n. 1, p. 1-10, 2014.

KHATCHATOURIAN, Oleg; TRETER, Jaciara. Aplicação da lógica fuzzy para avaliação econômico-financeira de cooperativas de produção. **JISTEM J.Inf.Syst. Technol. Manag.** (Online), São Paulo, v. 7, n. 1, p. 141-162, 2010. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1807-17752010000100007&lng=en&nrm=iso. Acesso em 11abr. 2021. <https://doi.org/10.4301/S1807-17752010000100006>.

LOBÃO, Diomar Cesar; SANCHES, Ionildo José; FURLAN, Diógenes Lago. **INTRODUÇÃO AOS MÉTODOS NUMÉRICOS**. Volta Redonda, RJ: Universidade Federal Fluminense, 2017. 160 p.

LOPES, Pedro Cortez; RANGEL, Rafael Lopez; MARTHA, Luiz Fernando. **O Ambiente App Designer do MATLAB**. PUC - Rio de Janeiro, 2019. 40 slides, color. Disponível em: http://webserver2.tecgraf.puc-rio.br/ftp_pub/lfm/CIV2801-AmbienteAppDesigner.pdf. Acesso em: 05 maio 2021.

LUCKE, Terry; KEYSSNER, Ulrike; DUNN, Peter. The use of a Classroom Response System to more effectively flip the classroom. In: **2013 IEEE Frontiers in Education Conference (FIE)**. IEEE, 2013. P. 491-495.

MACHADO, N.J. **Matemática e Realidade**. São Paulo: Cortez, 1988.

MARIANI, V. C.; PETERS, Sérgio. Utilização do Software Derive no Ensino de Cálculo Numérico. **Revista Exatas**, Chapecó, v. 02, n.02, p. 37-49, 1998.

MATLAB. **Object-Oriented Programming in MATLAB** 2021. Disponível em: <https://www.mathworks.com/products/matlab/object-oriented-programming.html>. Acesso em: 14 maio. 2021.

MATLAB. **What is Matlab?** 2021. Disponível em: <https://www.mathworks.com/discovery/what-is-matlab.html>. Acesso em: 08 abr. 2021.

MORAN, José Manuel et al. **Novas Tecnologias e mediação pedagógica**. 6. Ed. Campinas; Papirus, 2000.

MOURA, Allison de Sousa. **Uma ferramenta computacional para ensino-aprendizagem de métodos numéricos**. 2017. 61 f. TCC (Graduação) – Curso de Bacharelado em Matemática Computacional, Centro de Informática, Universidade Federal da Paraíba, João Pessoa, 2017.

MOURA, Carlos Mendonça e et al. **Alguns apontamentos da história da Análise Numérica**, 2005. 15 slides. Disponível: <Alguns apontamentos da história da Análise Numérica - PDF Download grátis (docplayer.com.br)>. n. 1, 2014, p. 141-166.

NASCIMENTO, Demilson Antônio. **Métodos para Encontrar Raízes Exatas e Aproximadas de Funções Polinomiais até o 4º Grau**. 2015. 81 f. Dissertação (Mestrado) – Curso de Mestrado Profissional em Matemática, Centro de Ciências Exatas e da Natureza, Universidade Federal da Paraíba, João Pessoa, 2015.

NEVES, Thiago José Galvão das. **Causas da evasão de alunos nos cursos de graduação presencial da UFPE**. 2016. Disponível em: https://www.ufpe.br/documents/38954/371376/r_evaso_16.pdf/53642e52-41fb-4b43-b098-98db6a470176. Acesso em: 09 abr. 2021.

PAULISTA, Cássio Rangel et al. DESENVOLVIMENTO DE SOFTWARE PARA SOLUÇÃO NUMÉRICA DE EQUAÇÕES POLINOMIAIS: UMA ABORDAGEM INTERDISCIPLINAR. **Revista Mundi Engenharia, Tecnologia e Gestão** (ISSN: 2525-4782), v. 3, n. 2, 2018.

PIRES, Augusto de Abreu. **Cálculo Numérico: prática com algoritmos e planilhas**. São Paulo: Atlas, 2015. 224 p.

RUGGIERO, M.A.G.; LOPES, V.L.R. **Cálculo Numérico: Aspectos Teóricos e Computacionais**, Makron Books, 2a Edição, 1997.

SACCARO, Alice; FRANÇA, Marco Túlio Aniceto; JACINTO, Paulo de Andrade. Fatores Associados à Evasão no Ensino Superior Brasileiro: um estudo de análise de sobrevivência para os cursos das áreas de Ciência, Matemática e Computação e

de Engenharia, Produção e Construção em instituições públicas e privadas. **Estudos Econômicos (São Paulo)**, v. 49, n. 2, p. 337-373, 2019.

SALVADOR, Valter. **O que é o Matlab?** 2016. Disponível em: <https://www.portalgsti.com.br/2016/08/o-que-e-o-matlab.html>. Acesso em: 12 maio 2021.

SASSANO, Fernando et al. O USO DE FERRAMENTAS COMPUTACIONAIS NO PROCESSO DE ENSINO E APRENDIZAGEM DE MATEMÁTICA NA EAD. **Revista Brasileira de Aprendizagem Aberta e A Distância**, [S.L.], v. 20, n. 1, 18 maio 2021. ABED - Associação Brasileira de Educação a Distância. <http://dx.doi.org/10.17143/rbaad.v20i1.551>.

SOARES, Welington; ALVES, Thaise Dias; STOLFI, Ana Maria. **Tópicos especiais em educação**. Livro Digital: Uniasselvi, 2020.

SOBRAL, Andre. **Informática na educação**. 2019. Disponível em: <https://medium.com/reflex%C3%A3o-computacional/inform%C3%A1tica-na-educa%C3%A7%C3%A3o-82d4dc7fdd28>. Acesso em: 02 maio 2021.

SOUSA, Reudismam Rolim *et al.* Pré-Algoritmos – Ações de Apoio à Melhoria do Ensino de Graduação. **Brazilian Journal Of Development**, [S.L.], v. 6, n. 3, p. 12625-12635, 2020. Brazilian Journal of Development. <http://dx.doi.org/10.34117/bjdv6n3-213>.

SOUZA, Quezia Emanuely de Oliveira *et al.* Comparativo entre os Métodos da Secante, de Newton e de Newton Utilizando Derivada Numérica. **Proceeding Series of the Brazilian Society of Computational and Applied Mathematics**, v. 7, n. 1, 2020.

SOUZA, Quézia Emanuely de Oliveira. **Comparativo entre os métodos de Newton-Raphson e das raízes múltiplas no cálculo de raízes de funções**. 2018. 10 f. TCC (Graduação) - Curso de Ciência e Tecnologia, Centro de Ciências Exatas e Naturais, Universidade Federal Rural do Semi-Árido, Mossoró, 2018. Tecnologias Digitais de Informação e Comunicação. **Revista UNIFESO**. v. 1,

TEIXEIRA, Thiago; SILVA, Iara. Método do Formigueiro para Encontrar os Zeros de Funções Reais. **Revista Eletrônica de Iniciação Científica em Computação**, v. 16, n. 4, 2018.

VALENTE, J. Armando. A Comunicação e a Educação baseada no uso das Tecnologias Digitais de Informação e Comunicação. **Revista UNIFESO**. v. 1, n. 1, 2014, p. 141-166.

VALENTE, José Armando. Uso da internet em sala de aula. **Educar em revista**, n. 19, p. 131-146, 2002.

VIEIRA, Hélida Vasques Peixoto *et al.* O Uso de Aplicativos de Celular como Ferramenta Pedagógica para o Ensino de Química. **Revista Debates em Ensino de Química**, v. 5, n. 1 ESP, p. 125-138, 2019.

ZABALA, Francisco José Correa; PARKER, Heidi E.; VIEIRA, Camilo. Implementing an active learning platform to support student learning in a numerical analysis course. In: **2017 IEEE Frontiers in Education Conference (FIE)**. IEEE, 2017. p. 1-6.

ZULYADAINI, Z. **Development of teaching materials in numerical methods**. Ukrainian Journal of Educational Studies and Information Technology, v. 8, n. 1, p. 28-38, 27 mar. 2020.