



CÁLCULO DIFERENCIAL E INTEGRAL EVOLUINDO A INTELIGÊNCIA COMPUTACIONAL MEDIANTE A LINGUAGEM PYTHON DE PROGRAMAÇÃO

**Rodrigo Antônio Pereira Junior¹, Denis Carlos Lima Costa², Erick Freitas da Costa³,
Adriane Cristina Fernandes Reis⁴, Bruno Henrique Parente de Carvalho⁵,
Pedro Dimas da Cunha Lima⁶, Mara Lúbia Viana de Lima⁷, Edson Costa Cruz⁸**

¹Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (rpereira.junior@ifpa.edu.br)

²Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (denis.costa@ifpa.edu.br)

³Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (efreitas256@gmail.com)

⁴Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (adriane.fr22@gmail.com)

⁵Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (brunoparente22@gmail.com)

⁶Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (cunhapedro665@gmail.com)

⁷Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (mara.lima@ifpa.edu.br)

⁸Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Cametá, Brasil (edson.cruz@ifpa.edu.br)

Resumo: A proeminente relevância do Cálculo Diferencial e Integral, na Ciência e na Tecnologia, foram delineadas neste artigo. Explanados computacionalmente, Limites, Derivadas e Integrais, foram desenvolvidos em PYTHON, criando uma interface que favorece o processo de ensino-aprendizagem, com característica multidisciplinar. Três estudos estão dispostos no artigo, comprovando a aplicabilidade da estratégia em resultados, analíticos e gráficos, com altíssimo grau de precisão e acurácia.

Palavras-chave: Matemática; Ciência; Tecnologia; Cálculo; Computação.

INTRODUÇÃO

Gottfried Wilhelm Leibniz (1646 - 1716) e Isaac Newton (1643 - 1727), desenvolveram, paralelamente, um dos mais belos e importantes métodos destinados à interpretação da Natureza: O Cálculo Diferencial e Integral.

Essa metodologia permite uma ampliação da percepção do comportamento de um dado fenômeno, descrito por funções matemáticas (Fulini, 2016).

Para Eves (2012), O Cálculo Diferencial e Integral adicionado à Geometria Analítica, formam uma das principais descobertas do Século XVII.

Os conceitos e aplicações de Limite, Derivada e Integral transcenderam os séculos. Suas aplicações estão cada vez mais presentes em estudos de inúmeras

áreas da Ciência e suas Tecnologias. Assim sendo, a resolução algébrica dessa, magnífica, análise infinitesimal, necessita de maior dinâmica. Logo, um mecanismo capaz de resolver cálculos, cada vez mais complexos e com alta precisão e acurácia, como a Computação, é imprescindível.

Batias et al (2021), expõem resultados de um estudo histórico-epistemológico realizado para identificar critérios, a fim de projetar tarefas que promovam a compreensão da noção de Limite em uma função de variável Real. Os critérios pelos autores permitem destacar não só a complexidade matemática subjacente ao estudo do Limite de uma função variável Real, mas também a riqueza de significados que podem ser desenvolvidos para ajudar a compreender esta noção.



Burgos et al (2021), destacam que apreender os conceitos e procedimentos do Cálculo Diferencial, em particular o conceito de Integral, é um desafio para professores e alunos em suas trajetórias acadêmicas. Desenvolver um plano para melhorar os processos instrucionais, requer peculiar atenção à complexidade das características matemáticas da Integral e sua aplicação na interpretação de fenômenos da Natureza.

Verón e Giacomone (2021), discutem diversos significados do conceito de Cálculo Diferencial e aplicam ferramentas e teorias Ontosemióticas, enfatizando os princípios pragmáticos apresentados por Leibniz, Cauchy, Fréchet e Robinson.

Carvalho et al (2021), indagam sobre as aplicações do Cálculo Diferencial e Integral nas Ciências e Engenharias.

Neste artigo, incrementa-se ao Cálculo Diferencial e Integral, a Modelagem Computacional. Para isso, emprega-se a Linguagem PYTHON de Programação, pois, segundo Costa et al (2019), essa linguagem disponibiliza todos os requisitos preconizados pelo Cálculo.

O trabalho compôs uma interface computacional proficiente às operações com Limites, Derivadas e Integrais, assim como à resolução e representação gráfica de funções com uma variável independente. São apresentados três estudos de casos que envolvem a multidisciplinaridade.

Essa pesquisa ascendeu o desenvolvimento de uma série de componentes curriculares no Bacharelado em Ciência e Tecnologia do Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA Campus Ananindeua, denominadas Cálculo Diferencial e Computacional I, II, III e IV. Nessas componentes, utiliza-se, além da Linguagem PYTHON, as Linguagem R e MATLAB.

MATERIAL E MÉTODOS

A metodologia aplicada neste artigo está fundamentada nos seguintes trabalhos:

Costa et al (2019), apresentam Métodos Matemáticos desenvolvidos em ambiente computacional, utilizando para isso a linguagem PYTHON de programação.

Costa et al (2020) aplicam a Modelagem Matemática-Computacional para interpretar fenômenos relacionados à gestão ambiental.

Costa et al (2021), divulgam a Matemática Computacional aplicada à Ciência e Tecnologia, desenvolvida em linguagem MATLAB de programação (Mathworks, 2021).

Em todos esses trabalhos é sempre destacado a interdisciplinaridade como instrumento responsável pela evolução das Ciências e suas Tecnologias. E ainda, Costa et al (2019 a 2021) reconhecem que a

Modelagem Matemática pode ser, amplamente, implementada por meio da Modelagem Computacional.

Diante disso, esta pesquisa parte da premissa que o ensino de Cálculo Diferencial e Integral pode ser executado com múltiplos objetivos, envolvendo sistemas com várias linguagens, a partir da aplicação das modelagens Matemática e Computacional. Essa combinação de linguagens irá proporcionar uma análise muito mais ampla e uma interpretação bem mais precisa dos fenômenos da Natureza.

A primeira fase da metodologia está vinculada com a ideia de Limites: Se os valores da função $f(x)$ se aproximarem cada vez mais do número L , enquanto x se aproximar cada vez mais do número a , diz-se que L é o Limite de $f(x)$, quando x tende a a , e escreve-se, conforme a Equação (1) (Leithold, 1994).

$$\lim_{x \rightarrow a} f(x) = L \quad (1)$$

Para modelos que retratam Limites comuns e Limites especiais, como por exemplo, $\frac{0}{0}$, $\frac{k}{0}$ ($k \neq 0$), $\frac{\infty}{\infty}$, $\frac{k}{\infty}$ ($k \neq 0$), entre outros, é possível aplicar o Script 1, apresentado nas Figuras 1 e 2, modelado para resolver as seguintes expressões:

$$\lim_{x \rightarrow 4} \frac{x^2 - 2x - 8}{x - 4} = \frac{0}{0} \quad (2)$$

$$\lim_{x \rightarrow \infty} \frac{x^2 - 2x - 8}{x - 4} = \frac{\infty}{\infty} \quad (3)$$

```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 16:53:50 2022
4
5  @author:
6  Instituto de Modelagem Matemática e Modelagem Computacional
7  Cálculo de Limites
8
9  """
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (x**2 - 2*x - 8)/(x - 4)
16 # Visualizando f
17 print('f =', f)
18 # Valor da Tendência: a
19 # Valor do limite de f para x -> a: L
20 a = 4
21 L = sy.limit(f, x, a)
22 # Visualizando L
23 print('L =', L)
24 print('')
25

```

Figura 1. Script 1 - Expressão (2).

A solução é dada por

$$f = (x^2 - 2x - 8)/(x - 4)$$

$$L = 6$$



```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul 2 16:53:50 2022
4
5 @author:
6 UFPA - Universidade do Pará - Instituto de Física
7 Cálculo de Limites
8 """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 import math as mt
13 # Declarando a variável simbólica: x
14 x = sy.Symbol('x')
15 # Declarando a Função: f(x)
16 f = (x**2 - 2*x - 8)/(x-4)
17 # Visualizando f
18 print('f = ', f)
19 # Valor da Tendência: a
20 # Valor do Limite de f para x --> a: L
21 a = mt.inf
22 L = sy.limit(f, x, a)
23 # Visualizando L
24 print('L = ', L)
25 print('-----')
```

Figura 2. Script 1 - Expressão (3).

A solução é dada por

$$f = (x^2 - 2x - 8)/(x - 4)$$

$$L = \infty$$

Percebe-se que, em ambos os casos, é utilizada a biblioteca *sympy*. Essa biblioteca é utilizada para cálculos com matemática simbólica (Lutz e Ascher, 2007). Ou seja, operações com sistema de álgebra computacional completo, como os executados no Cálculo Diferencial e Integral com uma ou mais variáveis.

Na resolução da Expressão (3) foi necessário a importação de mais uma biblioteca, a *math*. Essa biblioteca disponibiliza elementos e funções matemáticas especiais, como números complexos e as operações com o infinito, ∞ (Ketkar, 2017).

O script em PYTHON, destinado a representação gráfica de funções, pode ser visualizado na Figura 3. O código foi desenvolvido baseado na Função (4).

$$f(x) = \frac{x^2 - 2x - 8}{x - 4} \quad (4)$$

$$-2 \leq x \leq 10.$$

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Jul 3 10:41:45 2022
4
5 @author:
6 UFPA - Universidade do Pará - Instituto de Física
7 Representação Gráfica de funções
8 """
9
10 # Importando Bibliotecas
11 import numpy as np
12 import matplotlib.pyplot as plt
13 # Declarando o intervalo de análise
14 # x = np.linspace(início, fim, número de pontos)
15 x = np.linspace(-2, 10, 20)
16 # Declarando a Função: f(x)
17 f = (x**2 - 2*x - 8)/(x-4)
18 # Representação Gráfica
19 plt.plot(x, f, 'r', linewidth = 3)
20 plt.xlabel('Valores de x')
21 plt.ylabel('Valores de f(x)')
22 plt.title('f = (x**2 - 2*x - 8)/(x-4)')
23 plt.grid(True)
```

Figura 3. Script 2 para Função (4).

O gráfico de (4), gerado a partir execução do Script 2, está exposto na Figura 4.

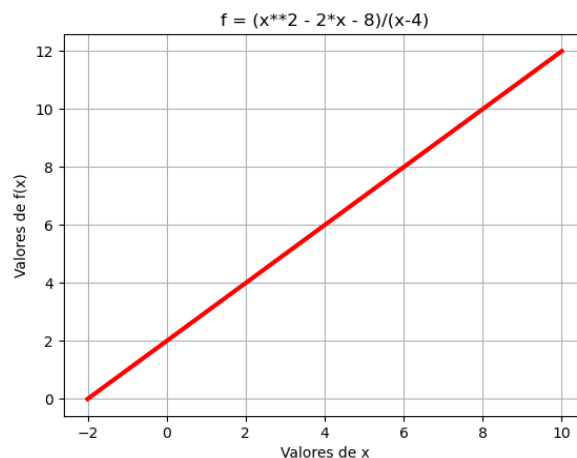


Figura 4. Representação gráfica da Função (4).

A segunda fase da metodologia está vinculada com a ideia das Derivadas. Para Villegas (2014), a Derivada de uma função, definida em relação à variável Real x , da qual depende, se e somente se, o Limite existir no domínio de $\mathbb{R}$, é dada pela Expressão (5).

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (5)$$

O Script 3, evidenciado na Figura 5, apresenta a Modelagem Computacional, em linguagem PYTHON, designado às Derivadas de ordem 1 e 2, para a Função (6).

$$f(x) = x^2 + e^{4x} + \sin(3x) \quad (6)$$

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul 2 17:07:34 2022
4
5 @author:
6 UFPA - Universidade do Pará - Instituto de Física
7 Cálculo de Derivadas
8 """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = x**2 + sy.exp(4*x) + sy.sin(3*x)
16 # Visualizando f
17 print('f = ', f)
18 # Derivada 1ª de f em relação a x: df
19 df = sy.diff(f, x)
20 # Visualizando df
21 print('df = ', df)
22 # Derivada 2ª de f em relação a x: d2f
23 d2f = sy.diff(df, x)
24 # Visualizando d2f
25 print('d2f = ', d2f)
```

Figura 5. Script 3 para Função (4).

O conjunto de soluções é dado por

$$f = x^2 + \exp(4x) + \sin(3x)$$

$$d1f = 2x + 4 \exp(4x) + 3 \cos(3x)$$

$$d2f = 2 + 16 \exp(4x) - 9 \sin(3x)$$



No Script 3, linha 14, percebe-se que as Funções Exponencial e Seno precisam ser importadas da biblioteca *sympy*. Por esse motivo, elas aparecem precedidas do termo *sy*. Processo semelhante ocorre com a operações de Limites, Derivadas e Integrais.

O Script 2 foi adaptado para representar, graficamente, as resoluções do Script 3, conforme mostra a Figura 6.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Jul 3 10:41:45 2022
4
5 @author:
6 GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7 Representação Gráfica de funções
8 """
9
10 # Importando Bibliotecas
11 import numpy as np
12 import matplotlib.pyplot as plt
13 # Declarando o intervalo de análise
14 # x = np.linspace(início, fim, número de pontos)
15 x = np.linspace(9,10,50)
16 # Declarando a Função: f(x)
17 f = x**2 + np.exp(4*x)*np.sin(3*x)
18 df = 2*x + 4*np.exp(4*x) + 3*np.cos(3*x)
19 d2f = 2 + 16*np.exp(4*x) - 9*np.sin(3*x)
20 # Representação Gráfica
21 plt.plot(x,f,'r',Label = 'f(x)',linewidth = 3)
22 plt.plot(x,df,'b',Label = 'df',linewidth = 3)
23 plt.plot(x,d2f,'g',Label = 'd2f',linewidth = 3)
24 plt.xlabel('x')
25 plt.ylabel('f,df,d2f')
26 plt.title('Derivadas de f(x)')
27 plt.legend()
28 plt.grid(True)
```

Figura 6. Script 2 ajustado à Função (6) e suas derivadas.

Percebe-se que o Script 2 sofreu ajustes que possibilitam a plotagem, simultânea, de várias funções. Essa produção de múltiplas curvas facilita a comparação entre a função primitiva $f(x)$ e suas derivadas $\frac{df}{dx}$ e $\frac{d^2f}{dx^2}$, no mesmo diagrama.

A representação gráfica da resolução do Script 3, referente a Função (6), aparece na Figura 7.

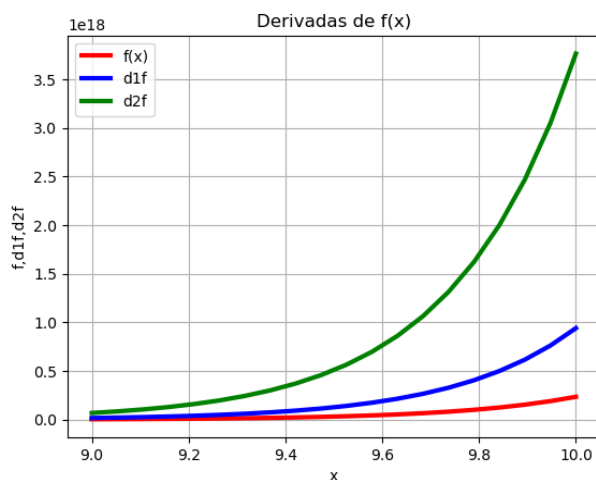


Figura 7. Comportamentos de $f(x)$, $\frac{df}{dx}$ e $\frac{d^2f}{dx^2}$.

A terceira fase da metodologia está vinculada com a ideia das Integrais. De acordo com Villegas (2014), a Integral é uma antiderivada e, portanto, se uma função da forma

$$f(x) = \frac{dF(x)}{dx},$$

decorre que,

$$Ifx = \int f(x)dx = \int \frac{dF(x)}{dx}dx = F(x) + C \quad (7)$$

Sendo $F(x)$ a função primitiva de $f(x)$.

Generalizando, tendo como premissa o teorema fundamental do Cálculo, pode-se afirmar que, existindo uma função $f(x)$, integrável em um intervalo fechado $[a, b]$, então

$$Ifx = \int_a^b f(x)dx = F(b) - F(a) \quad (8)$$

A Figura 8 mostra o Script 4, referente à Modelagem Computacional da Integral Indefinida de $f(x)$.

A função aplicada, na simulação, está representada na Expressão (9).

$$f(x) = x + \sin(x) \quad (9)$$

Em seguida, na Figura 9, o Script 4 experimentará uma adequação que permitirá efetuar a Integral Definida de $f(x)$, no intervalo $[0, 2\pi]$. Dessa forma, estará realizada a Modelagem Matemática-Computacional para as Expressões (7) e (8).

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul 2 17:12:08 2022
4
5 @author:
6 GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7 Cálculo de Integrais
8 """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = x + sy.sin(x)
16 # Visualizando f
17 print(f)
18 # Integral de f(x) em relação a x: Ifx
19 Ifx = sy.integrate(f,x)
20 # Visualizando Ifx
21 print('Ifx =', Ifx)
```

Figura 8. Script 4a – Integral Indefinida de $f(x)$.

A solução é dada por

$$f = x + \sin(x)$$

$$Ifx = x**2/2 - \cos(x)$$

Sendo Ifx a Integral da função f .



```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 17:12:08 2022
4
5  @author:
6  UFPA - Universidade do Pará - Faculdade de Engenharia e Tecnologia
7  Cálculo de Integrais
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = x + sy.sin(x)
16 # Visualizando f
17 print('f =', f)
18 # Integral de f(x) em relação a x: Ifx
19 # Limites de Integração: [a,b]
20 a = 0
21 b = 2*sy.pi
22 Ifx = sy.integrate(f, (x, a, b))
23 # Visualizando Ifx
24 print('Ifx =', Ifx)

```

Figura 9. Script 4b – Cálculo da Integral Definida de uma função $f(x)$.

A solução é dada por

$$f = x + \sin(x)$$

$$Ifx = 2 * \pi ** 2$$

A representação gráfica da Função (9) e sua Integral (Ifx), foi limitada pelo intervalo

$$-2\pi \leq x \leq 2\pi,$$

conforme mostra a Figura 10.

```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sun Jul 3 10:41:45 2022
4
5  @author:
6  UFPA - Universidade do Pará - Faculdade de Engenharia e Tecnologia
7  Representação Gráfica de funções
8  """
9
10 # Importando Bibliotecas
11 import numpy as np
12 import matplotlib.pyplot as plt
13 # Declarando o intervalo de análise
14 x = np.linspace(início, fim, número de pontos)
15 x = np.linspace(-2*np.pi, 2*np.pi, 40)
16 # Declarando as Funções: f(x) e Ifx
17 f = x + np.sin(x)
18 Ifx = x**2/2 - np.cos(x)
19 # Representação Gráfica
20 # Gráfico da função primitiva: f
21 plt.subplot(2, 1, 1)
22 plt.plot(x, f, 'r', label = 'f(x)', linewidth = 3)
23 plt.title('Funções Primitiva e Integral')
24 plt.ylabel('f(x)')
25 plt.legend()
26 # Gráfico da Integral de f: Ifx
27 plt.subplot(2, 1, 2)
28 plt.plot(x, Ifx, 'b', label = 'Ifx', linewidth = 3)
29 plt.xlabel('x')
30 plt.ylabel('Ifx')
31 plt.legend()
32 plt.grid(True)
33

```

Figura 10. Script 2 – ajustado para o cálculo da Integral de $f(x)$.

O produto da execução do Script 2, ajustado para a representação gráfica de $f(x)$ e sua Integral Definida, pode ser avaliado na Figura 11.

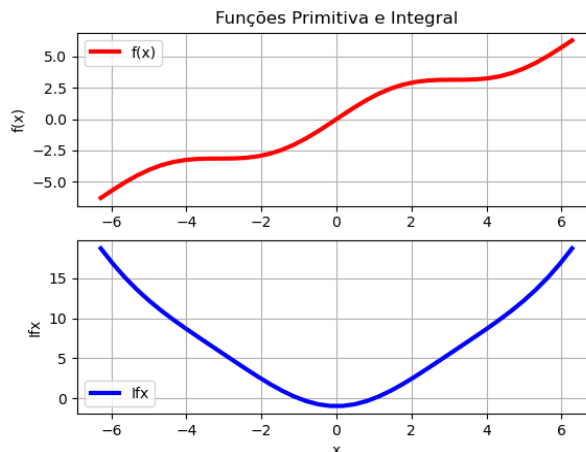


Figura 11. Comportamentos das Funções $f(x)$ e Ifx .

RESULTADOS E DISCUSSÃO

Esta pesquisa realizou aplicações multidisciplinares, de Cálculo Diferencial e Computacional, segmentadas em três estudos de caso.

Caso I – Este estudo relaciona a taxa de crescimento de uma planta em função no nível de precipitação. Para Song et al (2016), as alterações no clima têm ampla interferência nos ciclos de carbono e água do ecossistema, modificando as permutas gasosas das plantas. Em seus experimentos de manipulação de mudanças globais foi revelado respostas ecofisiológicas, indicando que o tamanho de uma planta, em aquecimento climático, varia com a precipitação x (mm), em concordância com a Equação (10).

$$B(x) = \frac{40x}{\sqrt{1+x^2}} + 20 \quad (10)$$

Sendo B , o tamanho da planta em metro (m).

A taxa de variação do crescimento, em relação à precipitação, pode ser estimada pela Derivada 1ª da Equação (10), de acordo com a Equação (11).

$$\frac{dB}{dx} = \frac{40}{\sqrt{1+x^2}} - \frac{40x^2}{\sqrt{(1+x^2)^3}} \quad (11)$$

Para uma precipitação tendendo a 62.75 mm, o tamanho da planta tenderá a 59.9949 m, conforme estimado por Song et al (2016). Este resultado pode ser comprovado fazendo

$$\lim_{x \rightarrow 62.75} \frac{40x}{\sqrt{1+x^2}} + 20 = 59.9949 \quad (12)$$

Analisando a Figura 12, percebe-se que o Script 3 foi regulado para calcular a Derivada 1ª da Equação (10).



```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 17:07:34 2022
4
5  @author:
6  GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Derivadas
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (40*x/sy.sqrt(1+x**2))+20
16 # Visualizando f
17 print('f = ',f)
18 # Derivada 1ª de f em relação a x: dif
19 dif = sy.diff(f,x)
20 # Visualizando dif
21 print('dif = ',dif)

```

Figura 12. Script 3 – ajustado à Equação (10).

O resultado encontrado está de acordo com a Equação (11).

$$d1f = -40 * x ** 2 / (x ** 2 + 1) ** (3/2) + 40 / \text{sqrt}(x ** 2 + 1)$$

A Figura 13 exibe o Script 1, regulado para calcular o Limite da Equação (10), avalizando a sua tendência, em conformidade com a Expressão (12).

```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 16:53:50 2022
4
5  @author:
6  GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Limites
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (40*x/sy.sqrt(1+x**2))+20
16 # Visualizando f
17 print('f = ',f)
18 # Valor da Tendência: a
19 a = 62.75
20 # Valor do Limite de f para x --> a: L
21 L = sy.Limit(f,x,a)
22 # Visualizando L
23 print('L = ',L)
24 print('L = ',L)
25 print('L = ',L)

```

Figura 13. Script 1 – ajustado à Equação (10).

As dinâmicas da relação $B = f(x)$ e da sua taxa de variação, estão ilustradas na Figura 14.

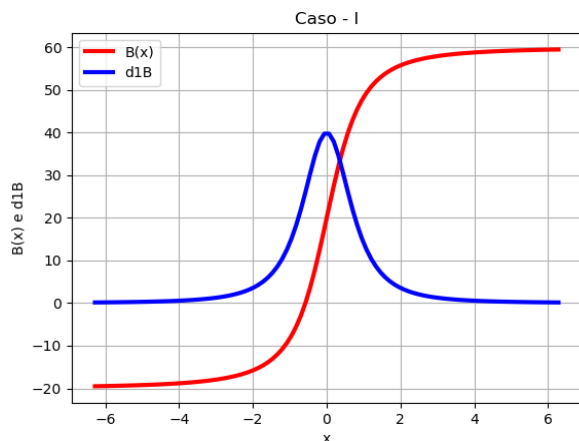


Figura 14. Diagrama das Equações (10) e (11).

Caso II – Este estudo relaciona o Valor Médio Quadrático da Corrente Elétrica. Fundamentado em Costa et al (2021), entende-se que a transmissão eficiente em um Sistema Elétrico está associada à Corrente Elétrica (i) do circuito CA (Corrente Alternada). Essa Corrente, frequentemente, é configurada em um modelo de onda senoidal, cujo Valor Médio Quadrático (i_{vmq}) pode ser calculado pela Equação (13).

$$i_{vmq} = \sqrt{\int_0^t [10e^{-t} \sin(2\pi t)]^2 dt} \quad (13)$$

Sendo, t o instante final de avaliação do Sistema de Energia Elétrica.

A Figura 15 apresenta o Script 4, ajustado para resolver a Equação (13), no intervalo $t = [0, 1]$ s.

```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 17:12:08 2022
4
5  @author:
6  GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Integrais
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 t = sy.Symbol('t')
14 # Declarando a Função: f(x)
15 f = 10*sy.exp(-1*t)*sy.sin(2*sy.pi*t)
16 # Visualizando f
17 print('f = ',f)
18 # Integral de f(x) em relação a x: Ifx
19 # Limites de Integração: [a,b]
20 a = 0
21 b = 1
22 Ifx = sy.integrate(f,(t, a, b))
23 i_vmq = sy.sqrt(Ifx)
24 # Visualizando Ifx
25 print('Ifx = ',Ifx)
26 # Visualizando i_vmq
27 print('i_vmq = ',i_vmq)

```

Figura 15. Script 4 – ajustado à Equação (13).

Os resultados obtidos são

$$f = 10 * \exp(-t) * \sin(2 * \pi * t)$$

$$Ifx = -20 * \pi / (e + 4 * e * \pi ** 2) + 20 * \pi / (1 + 4 * \pi ** 2)$$

Para a Integral Indefinida tem-se

$$i_{vmq} = \sqrt{-10 * \sin(2 * \pi * t) / (\exp(t) + 4 * \pi ** 2 * \exp(t)) - 20 * \pi * \cos(2 * \pi * t) / (\exp(t) + 4 * \pi ** 2 * \exp(t))} \quad (14)$$

Para a Integral Definida tem-se

$$i_{vmq} = \sqrt{-20 * \pi / (e + 4 * e * \pi ** 2) + 20 * \pi / (1 + 4 * \pi ** 2)}$$

Ou seja,

$$Ifx = 0.981197$$

$$i_{vmq} = 0.990554 \text{ A.}$$



A representação gráfica da Equação (14) pode ser visualizada na Figura 16.

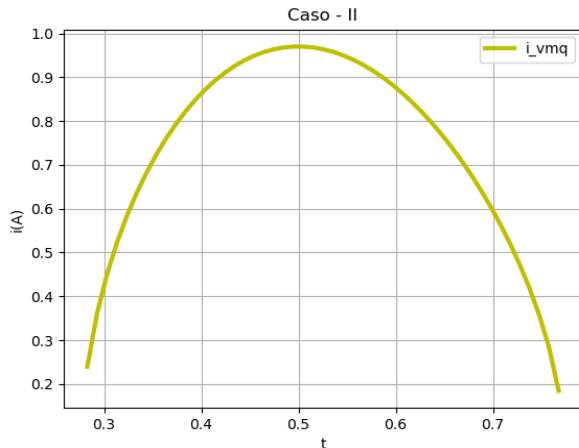


Figura 16. Diagrama da Equação (14).

Para a construção do gráfico da Equação (14) foi utilizado o Script 2, adaptado, para esta expressão, como mostra a Figura 17.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Jul 3 10:41:45 2022
4
5 @author:
6 GHS - Gradiente de Modelagem Matemática e Simulação Computacional
7 Representação Gráfica de funções
8 """
9 # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # t = np.linspace(início, fim, número de pontos)
14 t = np.linspace(0.3, 0.7, 100)
15 # Declarando as Funções: f(x) e Ifx
16 f = np.sqrt(-10*np.sin(2*np.pi*t)/(np.exp(t)+4*np.pi**2*np.exp(t))-
17          20*np.pi*np.cos(2*np.pi*t)/(np.exp(t)+4*np.pi**2*np.exp(t)))
18 # Representação Gráfica
19 # Gráfico da função primitiva: f
20 plt.plot(t, f, 'y', label = 'i_vmq', linewidth = 3)
21 plt.xlabel('t')
22 plt.ylabel('i(A)')
23 plt.title('Caso - II')
24 plt.legend()
25 plt.grid(True)
26
```

Figura 17. Script 2 – adaptado à Equação (14).

Caso III – Este estudo refere-se à análise de modelos que interpretam a eficiência de Custo, Receita e Lucro das unidades, para a tomada de decisão (Ashrafi e Kaleibar, 2017). Para os autores a eficiência de financeira pode ser definida como a relação entre os Custos a Receita.

Suponha que a taxa de variação da Receita de uma empresa possa ser estimada pela Equação (15).

$$\frac{dR}{dt} = -t^2 + 11t \quad (15)$$

Sendo, R o valor (em \$ 10^6) da Receita Bruta em venda daqui a t anos.

Da mesma forma, a taxa de variação dos Custos (em \$ 10^6) pode ser calculada pela Equação (16).

$$\frac{dC}{dt} = -t^3 + 7t^2 - 4t \quad (16)$$

Em conformidade com Ashrafi e Kaleibar (2017), a eficiência do Lucro Total (em \$ 10^6) pode ser evidenciada pela Equação (18).

$$L(t) = \int_{t_0}^{t_1} [-t^2 + 11t] dt - \int_{t_0}^{t_1} [-t^3 + 7t^2 - 4t] dt \quad (18)$$

Os comportamentos das Equações (17) e (18), podem ser examinados na Figura 18.

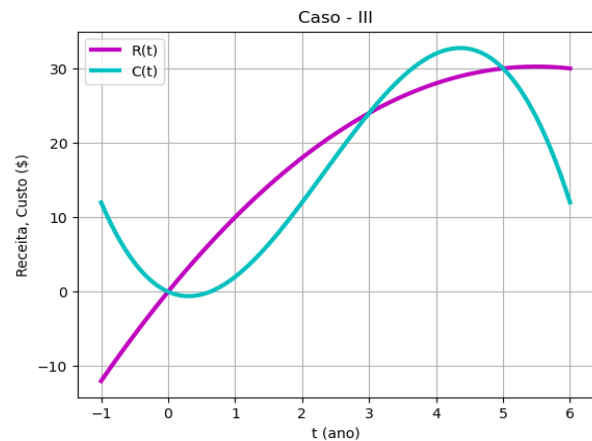


Figura 18. Diagrama da Equação (18).

Para a construção do gráfico da Equação (18) foi utilizado o Script 2, adaptado, para esta expressão, como mostra a Figura 19.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Jul 3 10:41:45 2022
4
5 @author:
6 GHS - Gradiente de Modelagem Matemática e Simulação Computacional
7 Representação Gráfica de funções
8 """
9 # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # t = np.linspace(início, fim, número de pontos)
14 t = np.linspace(-1, 6, 100)
15 # Declarando as Funções Receita, Custo e Lucro
16 R = -t**2 + 11*t
17 C = -t**3 + 7*t**2 - 4*t
18 L = R - C
19 # Representação Gráfica
20 # Gráfico das funções R e C
21 plt.figure(1)
22 plt.plot(t, R, 'm', label = 'R(t)', linewidth = 3)
23 plt.plot(t, C, 'c', label = 'C(t)', linewidth = 3)
24 plt.xlabel('t (ano)')
25 plt.ylabel('Receita, Custo ($)')
26 plt.title('Caso - III')
27 plt.legend()
28 plt.grid(True)
29 # Gráfico da função L
30 plt.figure(2)
31 plt.plot(t, L, 'r', label = 'L(t) = R(t) - C(t)', linewidth = 3)
32 plt.xlabel('t(ano)')
33 plt.ylabel('Lucro ($)')
34 plt.title('Caso - III')
35 plt.legend()
36 plt.grid(True)
37
```

Figura 19. Script 2 – adaptado à Equação (18).

Para determinação do Lucro, em função de tempo, deve-se estabelecer os valores para o intervalo de análise, ou seja, $[t_0, t]$. Esses valores serão utilizados



como limites para as Integrais, em conformidade com a Equação (18). Percebe-se, visualizando a Figura 18, que há três interseções entre as funções $R(t)$ e $C(t)$. Estes pontos de interseção definem os intervalos de análise e foram determinados da seguinte maneira:

$$R(t) = C(t)$$

Ou ainda,

$$R(t) - C(t) = 0.$$

O Script 5 foi implementado para resolver equações, ou seja, encontrar o conjunto solução (S) da variável independente responsável por zerar uma determinada função.

A Figura 20 apresenta o Script 5 estruturado resolver a Equação (20).

$$L(t) \rightarrow R(t) - C(t) = 0 \quad (20)$$

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Jul 5 15:10:00 2022
4
5 @author:
6 UFPA - Universidade do Pará - Faculdade de Engenharia
7 Resolução de Equações - Encontrar Raízes
8 """
9 # Importando Biblioteca
10 import sympy as sy
11 # Declarando a variável simbólica: t
12 t = sy.Symbol('t')
13 print('Equação a ser resolvida:')
14 # Declarando a Equação a ser resolvida: L = R - C
15 R = -t**2 + 11*t
16 C = -t**3 + 7*t**2 - 4*t
17 L = R - C
18 print('L =', L)
19 # Conjunto Solução: S
20 S = sy.solve(L)
21 print('Solução da Equação:')
22 print('S =', S)
23
```

Figura 20. Script 2 – adaptado à Equação (20).

Os resultados obtidos são

$$L = t^3 - 8t^2 + 15t$$

$$S = [0, 3, 5]$$

Ou seja,

$$L(t) = t^3 - 8t^2 + 15t \quad (21)$$

O comportamento da Equação (21), pode ser observado na Figura 21.

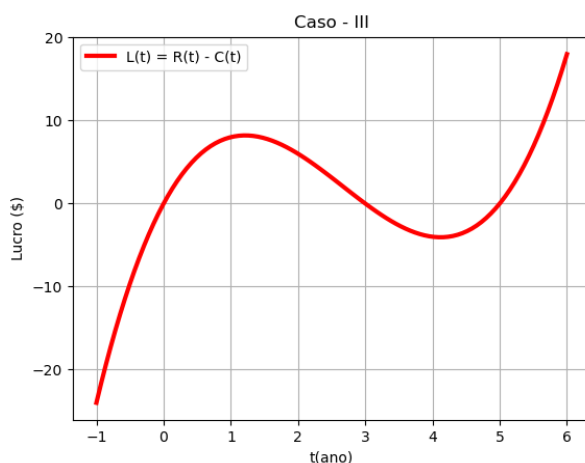


Figura 21. Gráfico da Equação (21).

De acordo com o gráfico da Figura 21, percebe-se que no intervalo, $t_p = [0, 3]$ o Lucro é positivo e, no intervalo $t_n = [3, 5]$, o Lucro é negativo (prejuízo).

Essa informação é corroborada pela análise da Figura 18, em que no intervalo $t_p = [0, 3]$ a curva da Função Receita está acima da curva da Função Custo e, no intervalo $t_n = [3, 5]$ a curva da Função Receita está abaixo da curva da Função Custo.

Diante disso, é presumível que, o Lucro Total pode ser contabilizado como o somatório das áreas delimitadas pelas curvas $R(t)$ e $C(t)$, no intervalo $t = [0, 5]$, limitantes mínimo e máximo da interseção entre elas, conforme a Equação (18) e ilustrado na Figura 22.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul 2 17:12:00 2022
4
5 @author:
6 UFPA - Universidade do Pará - Faculdade de Engenharia
7 Cálculo de Integrais
8 """
9 # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 t = sy.Symbol('t')
13 # Declarando as Funções: R(t) e C(t)
14 R = -t**2 + 11*t
15 C = -t**3 + 7*t**2 - 4*t
16 # Visualizando R e C
17 print('R =', R)
18 print('C =', C)
19 # Limites de Integração: [a,b]
20 a = 0.0
21 b = 5.0
22 # Integral de R em relação a t: IRt
23 IRt = sy.integrate(R, (t, a, b))
24 # Integral de C em relação a t: ICt
25 ICt = sy.integrate(C, (t, a, b))
26 # Arredondando: round(Variável, nº de casas)
27 # Visualizando IRt
28 print('IRt =', round(IRt, 2))
29 # Visualizando ICt
30 print('ICt =', round(ICt, 2))
31 # Valor do Lucro: L = IRt - ICt
32 L = IRt - ICt
33 # Visualizando o Valor do Lucro
34 print('L =', round(L, 2))
35
```

Figura 22. Script 4 – ajustado à Equação (18).

Os resultados obtidos são

$$R = -t^2 + 11t$$

$$C = -t^3 + 7t^2 - 4t$$

$$IRt \cong 95.83$$

$$ICt \cong 85.41$$

$$L \cong 10.42$$

A Inteligência Computacional permite resolver n – vezes, Cálculos de Limites, Derivadas e de Integrais, com altíssima precisão e acurácia. Assim sendo, a aferição das áreas, delimitadas pelas curvas $R(t)$ e $C(t)$, será executada em duas formas.

As Figuras 23 e 24, exibem essas resoluções considerando os seguintes limites de Integração:

Intervalo I $\rightarrow t_p = [0, 3]$;

Intervalo II $\rightarrow t_n = [3, 5]$.

Considerando essas duas circunstâncias, as observações sobre o comportamento do Lucro estarão mais evidenciadas.



```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 17:12:08 2022
4
5  @author:
6  """
7  # Importando Bibliotecas
8  import sympy as sy
9  # Declarando a variável simbólica: x
10 t = sy.Symbol('t')
11 # Declarando as Funções: R(t) e C(t)
12 R = -t**2 + 11*t
13 C = -t**3 + 7*t**2 - 4*t
14 # Visualizando R e C
15 print('R =', R)
16 print('C =', C)
17 # Limites de Integração: [a,b]
18 a = 0.0
19 b = 3.0
20 # Integral de R em relação a t: IRt
21 IRt = sy.integrate(R, (t, a, b))
22 # Integral de C em relação a t: ICt
23 ICt = sy.integrate(C, (t, a, b))
24 # Arredondando: round(Variável, n° de casas)
25 # Visualizando IRt
26 print('IRt =', round(IRt, 4))
27 # Visualizando ICt
28 print('ICt =', round(ICt, 4))
29 # Valor do Lucro: L = IRt - ICt
30 L = IRt - ICt
31 # Visualizando o Valor do Lucro
32 print('L =', round(L, 2))
33

```

Figura 23. Script 4 – Equação (18), intervalo I.

Os resultados obtidos são

$$R = -t^2 + 11t$$

$$C = -t^3 + 7t^2 - 4t$$

$$IRt = 40.5000$$

$$ICt = 24.7500$$

$$L = 15.75$$

```

1  #-*- coding: utf-8 -*-
2  """
3  Created on Sat Jul 2 17:12:08 2022
4
5  @author:
6  """
7  # Importando Bibliotecas
8  import sympy as sy
9  # Declarando a variável simbólica: x
10 t = sy.Symbol('t')
11 # Declarando as Funções: R(t) e C(t)
12 R = -t**2 + 11*t
13 C = -t**3 + 7*t**2 - 4*t
14 # Visualizando R e C
15 print('R =', R)
16 print('C =', C)
17 # Limites de Integração: [a,b]
18 a = 3.0
19 b = 4.0
20 # Integral de R em relação a t: IRt
21 IRt = sy.integrate(R, (t, a, b))
22 # Integral de C em relação a t: ICt
23 ICt = sy.integrate(C, (t, a, b))
24 # Arredondando: round(Variável, n° de casas)
25 # Visualizando IRt
26 print('IRt =', round(IRt, 4))
27 # Visualizando ICt
28 print('ICt =', round(ICt, 4))
29 # Valor do Lucro: L = IRt - ICt
30 L = IRt - ICt
31 # Visualizando o Valor do Lucro
32 print('L =', round(L, 2))
33

```

Figura 24. Script 4 – Equação (18), intervalo II.

Os resultados obtidos são

$$R = -t^2 + 11t$$

$$C = -t^3 + 7t^2 - 4t$$

$$IRt = 26.1667$$

$$ICt = 28.5833$$

$$L = -2.42$$

CONCLUSÃO

A crescente evolução tecnológica exige um constante aperfeiçoamento científico. A consciência matemática contribui para que essa evolução acrescente uma concepção crítica e reflexiva sobre a mecânica da Natureza, em todos os seus aspectos, sociais, ambientais, físicos, químicos, biológicos, entre outros.

Esse artigo transfere uma estratégia que permite otimizar a execução do Cálculo Diferencial e Integral, sem reduzir a relevância algébrica dessa esplêndida criação da humanidade.

Elaborada, inicialmente por Gottfried Wilhelm Leibniz e Isaac Newton, o Cálculo Diferencial e Integral foi incrementado e obteve uma interface computacional que estimula suas aplicações em modelos interdisciplinares.

Fundamentado pela Linguagem PYTHON de programação, foram apresentados *scripts* que operam Limites, Derivadas e Integrais. Essa Modelagem Matemática-Computacional diminui a possibilidade de erros algébricos e maximiza a compreensão do fenômeno estudado.

Três estudos foram retratados. Todos apresentaram elevados níveis de precisão e acurácia em seus resultados analíticos e gráfico.

Para trabalhos futuros, estima-se a implementação de modelos baseados em funções com múltiplas variáveis independentes, para análises de multicritérios.

AGRADECIMENTOS

Ao Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA *Campus* Ananindeua, pela administração do Bacharelado em Ciência e Tecnologia desse *campus*. E ao grupo de pesquisa Gradiente de Modelagem Matemática e Simulação Computacional – GM²SC, pela gestão e motivação científica de Discentes e Docentes.

REFERÊNCIAS

- ASHRAFI, A.; KALEIBAR, M. Mansouri. Cost, Revenue and Profit Efficiency Models in Generalized Fuzzy Data Envelopment Analysis. Fuzzy Information and Engineering - Elsevier. <https://doi.org/10.1016/j.fiae.2017.06.007>. 2017.
- BASTIAS, Daniela A.; PINO-FAN, Luis R.; MEDRANO, Iván G.; CASTRO, Walter F. Epistemic Criteria for Designing Limit Tasks on a Real Variable Function. Scientific Electronic Library Online. Bolema, Rio Claro (SP), v. 35, n. 69, p. 179-205.



ISSN 1980-4415 DOI: 4842-2766-4.
<http://dx.doi.org/10.1590/1980-4415v35n69a0>. 2021.

BURGOS, María; BUENO, Seydel; PÉREZ, Olga; Godino, Juan D. Onto-semiotic complexity of the Definite Integral. Implications for teaching and learning Calculus REDIMAT – Journal of Research in Mathematics Education, 10(1), 4-40. doi:10.17583/redimat.2021.6778. 2021.

CARVALHO, João P. A.; MACÊDO, Josué A. de; LOPES, Lailson dos R. P. Some Applications of Differential and Integral Calculus. Research, Society and Development, [S.l.], v. 10, n. 8, p. e22410817220. DOI: 10.33448/rsd-v10i8.17220. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/17220>. 2021.

COSTA, Denis C. L., COSTA, Heictor A. de O., & NEVES, Lucas P. Métodos Matemáticos Aplicados nas Engenharias via Sistemas Computacionais. SINEPEM-IFPA. 2019.

COSTA, Denis C. L.; COSTA, Heictor A. de O.; CASTRO, Ana P. S.; CRUZ, Edson C.; AZANCORT NETO, Julio L.; CRUZ, Brennus C. C. da. The dimensions of Mathematical and Computational Modeling prescribed to Environmental Management. Research, Society and Development, [S. l.], v. 9, n. 10, p. e6939109013. DOI: 10.33448/rsd-v9i10.9013. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/9013> 2020.

COSTA, Denis C. L., COSTA, Heictor A. de O., Silva, Hugo C. M. da, & Silva, Silvio T. T. da. Matemática Computacional Aplicada à Ciência e Tecnologia. Belém, PA, SINEPEM-IFPA. 2021.

COSTA, Heictor A. de O.; COSTA, Denis C. L.; MENESES, Lair A. de. Interdisciplinarity Applied to the Optimized Dispatch of Integrated Electricity and Natural Gas Networks using the Genetic Algorithm. Research, Society and Development, [S. l.], v. 10, n. 2, p. e42110212641. DOI: 10.33448/rsd-v10i2.12641. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/12641> 1. 2021.

FULINI, Márcio A. História do Cálculo Diferencial e Integral. Universidade Federal de São João Del-Rei – UFSJ Núcleo de Educação à Distância Departamento de Matemática e Estatística – DEMAT. <http://hdl.handle.net/123456789/86>. 2016.

EVES, Howard. Introdução à História da Matemática. Tradução HYGINO H. Domingues. 5ª ed. Campinas: Editora da Unicamp, 2012.

KETKAR, Nikhil. Deep Learning with PYTHON – A Hands-on Introduction. Apress. ISBN-13 (pbk): 978-1-4842-2765-7. ISBN-13 (electronic): 978-1-

4842-2766-4. DOI 10.1007/978-1-4842-2766-4. Springer Science+Business Media New York. 2017.

LEITHOLD, Louis. Cálculo com Geometria Analítica. Tradução: Cyro de Carvalho Patarra. São Paulo: Harbra, 1994.

LUTZ, Mark e ASCHER, David. Aprendendo PYTHON. Tradução - João Tortello. Porto Alegre: Bookman. ISBN: 978-85-7780-013-1. 2ª ed. 2007.

SONG, Bing; NIU, Shuli; WAN, Shiqiang. Precipitation regulates plant gas exchange and its long-term response to climate change in a temperate grassland. Journal of Plant Ecology. Volume 9, Issue 5, Pages 531–541, <https://doi.org/10.1093/jpe/rtw010>. 2016.

MATHWORKS. MathWorks leading developer of Mathematical Computing software for Engineers and Scientists. <https://www.mathworks.com/>. 2021.

VERÓN, Manuel A.; GIACOMONE, Belén. Análise dos significados do conceito de Diferencial de uma perspectiva Ontosemiótica. Revemop - Revista científica da Universidade Federal de Ouro Preto – UFOP/PPGEDMAT. v. 3, p. e202109. DOI: <https://doi.org/10.33532/revepope.e202109>. 2021.

VILLEGAS, Mauricio V. Física Integrada con Cálculo. Ibagué: Universidad de Ibagué. 492 p. ISBN Digital: 978-958-754-112-0. 2014.



SISTEMA DE ENERGIA MODELADO PELA DENSIDADE DE POTÊNCIA ELÉTRICA

**Denis Carlos Lima Costa¹, Ana Beatriz das Neves da Silva²,
Flávio Oliveira Dias³, Diego Alves Carvalho⁴,
Hector Matheus Machado Robatto⁵, Márcio José Oliveira de Araujo⁶**

¹Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (denis.costa@ifpa.edu.br)

²Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (ananeves537@gmail.com)

³Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (diasflavio95@gmail.com)

⁴Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (diegoalvescarvalho87@gmail.com)

⁵Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (hectormachadorobatto@gmail.com)

⁶Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (marcioj-araujo@hotmail.com)

Resumo: Os avanços tecnológicos da geração de energia estão relacionados com as pesquisas sobre as Potências Elétricas. Este artigo disponibiliza um estudo que apresenta a Densidade de Potência Elétrica como uma grandeza capaz de estimar as variações de energia elétrica, e sua região de ação, em um sistema de 14 barras. O trabalho aplica a interdisciplinaridade como estratégia de coexistência entre a Matemática, a Física e a Computação, interpretando assim, o comportamento do Sistema Elétrico.

Palavras-chave: Energia Elétrica; Modelagem Matemática; Interdisciplinaridade; Simulação Computacional.

INTRODUÇÃO

Os estudos sobre o comportamento da Energia Elétrica, em um sistema de barramentos, apresentam excepcional relevância à área dos despachos elétricos seguros (Costa et al, 2016).

Frequentemente as análises são realizadas tendo como, principais, premissas as variáveis Tensão Elétrica, Ângulo de Fase, Potência Ativa e Potência Reativa, em cada barra do sistema (Costa, Denis et al 2022a).

Esse artigo propõe uma investigação interdisciplinar da distribuição das Potências Elétricas Ativa, Reativa e Aparente, simultaneamente, em um sistema de 14 barras, utilizando, para isso, a Modelagem Matemática-Computacional incrementada por Costa, Heictor et al (2022).

Essa análise permitirá visualizar como a tríade das Potências varia, entendendo, dessa maneira, como a Densidade de Potência Elétrica está distribuída.

Vignale (1995) usou a covariância da equação de Schrödinger para transformação de um quadro de referência acelerado, mostrando que a função de Densidade Linear, de um sistema eletrônico acionado por um potencial, dependente do tempo em uma frequência na presença de um potencial estático é a densidade de equilíbrio.

Puga (2017) apresentou atividades, com uso computacional, sobre Área e Volume de Sólido de Revolução, utilizando metodologia acadêmica aplicada na componente curricular de Cálculo Diferencial e Integral, comprovando a importância da utilização de uma ferramenta de simulação computacional no processo interdisciplinaridade.



Dreßle et al (2019) apresentam um caso específico da função de Densidade Linear Estática e uma base específica para o cálculo da representação do autosistema, bem como da expansão do momento, ilustrando o procedimento real para aplicar a expansão do momento da função de resposta da Densidade Linear a uma molécula de água que está sujeito a um potencial de perturbação externo variável.

Belardinelli (2019) questiona a deformação estática da mola quando esta é suspensa verticalmente, com ou sem massa pendurada na extremidade livre, e aplica a Densidade de Massa Linear de molas para avaliar uma função de Densidade não-constante.

Dreßler e Sebastiani (2021) propõem a função de Densidade com uma forma eficiente de calcular forças moleculares para potenciais externos arbitrários, sem a necessidade de realizar um cálculo químico quântico autoconsistente explícito para cada configuração do ambiente químico.

Bukenberger e Lensch (2021) indicam que a Densidade dentro de um tetraedro pode variar linearmente, ou seja, Campos de Densidade arbitrários podem ser aproximados especificando a Densidade em todos os vértices de uma malha tetraédrica.

Costa, Denis et al (2022b), em seu trabalho, apresentaram uma sistematização interdisciplinar que relaciona Matemática, Física, Algoritmos Computacionais e Lógica de Programação para explicar os Sistemas Lineares que explicam as Leis de Kirchhoff.

Este trabalho apresenta a Densidade de Potência Elétrica (DPE) variável, expondo para isso soluções exatas relacionadas às Potências Elétricas Ativa (P), Reativa (Q) e a Aparente (P_{ap}), mediante representação de modelos não-lineares.

MATERIAL E MÉTODOS

Nos cursos vinculados à Ciência e Tecnologia, há um capítulo destinado ao cálculo do Centroide e das Densidades Linear, Superficial e Volumétrica. Esses modelos indicam, com magnífica precisão, a distribuição da massa de uma estrutura, em função da sua densidade variável.

Tais processos foram empregados para determinar o comportamento da Energia Elétrica em relação à dispersão da Potência Elétrica em sistema de barramentos.

As simulações computacionais foram realizadas em Linguagem MATLAB de Programação. Os dados aplicados são do Sistema 14 barras do IEEE – *Institute of Electrical and Electronics Engineers*, conforme indica a Figura 1 (PSTCA 1993).

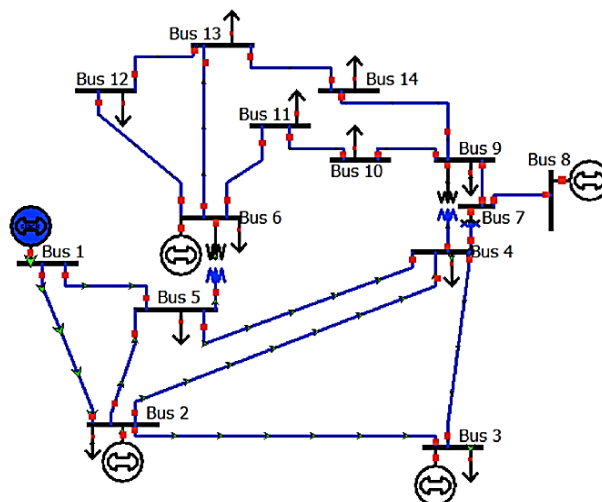


Figura 1. Rede Elétrica – Sistema de 14 barras.

Os valores de P , Q e P_{ap} , destacados na Tabela 1, foram submetidos ao Método de Regressão Múltipla Não-Linear (MRMNL) apresentado por Costa et al (2020), com a finalidade de modelar o comportamento da DPE.

Tabela 1. Valores das Potências Elétricas.

Barra	P (MW)	Q (MVA)	P_{ap} (MVA)
01	19,68	56,67	59,99
02	30,10	22,52	37,59
03	20,09	0,27	20,09
04	39,41	6,13	39,88
05	32,24	6,23	32,84
06	17,76	4,72	18,38
07	-30,88	1,49	30,92
08	24,77	7,86	25,99
09	14,32	6,66	15,79
10	22,36	31,76	38,84
11	10,66	-0,26	10,66
12	8,12	1,98	8,36
13	19,42	5,38	20,15
14	-0,00	-2,07	2,07

Fonte: PSTCA, 1993.



A partir desse processo foi possível encontrar a Equação (1), considerando a barra 1 como referência.

$$P_{ap}(P, Q) = 16.76 + 12.97P + 8.676Q + 2.564P^2 - 0.2088PQ - 1.174P^3 - 2.325P^2Q \quad (1)$$

Essa equação representa a variação da P_{ap} em função de P e Q .

Aplicando o MRMNL, alegando que $Q = f(P)$, tem-se a Equação (2).

$$Q(P) = 0.3641P^2 + 2.0950P + 4.5020 \quad (2)$$

Os valores assumidos por P que anulam a Equação (2) são, respectivamente,

$$P_1 = -2.8770 - 2.0218i$$

$$P_2 = -2.8770 + 2.0218i.$$

Essa informação ratifica que a conjectura de que $Q = f(P)$, sendo Q a Potência Reativa.

A partir dos modelos matemáticos, apresentados nas Equações (1) e (2), foi possível simular computacionalmente a DPE. Mediante a representação gráfica evidenciou-se como as Potências Elétricas atuam no sistema de 14 barras, exposto na Figura 1.

Uma vez conhecida a função $Q = f(P)$, dada pela Equação (2), é presumível que ela represente a Densidade de Potência Elétrica (DPE) variável. A localização da maior Concentração da Potência Elétrica (C), pode ser prevista pela Equação (3)

$$C = \frac{\int_a^b P \cdot Q(P) dP}{\int_a^b Q(P) dP} \quad (3)$$

Resolvendo a integral do numerador de (3), tem-se

$$\frac{P^2(1.0923P^2 + 8.38P + 27.012)}{12}$$

A solução da integral do denominador de (3) é

$$\frac{P(0.3641P^2 + 3.1425P + 13.506)}{3}$$

Dessa forma, a resolução de (3) pode ser representada da seguinte forma

$$C = \frac{P(1.0923P^2 + 8.38P + 27.012)}{4(0.3641P^2 + 3.1425P + 13.506)}$$

A DPE apresenta uma área de ação dada pela Equação (4)

$$A = 2\pi \int_a^b Q(P) \sqrt{1 + \left(\frac{dQ}{dP}\right)^2} dP \quad (4)$$

A solução da Equação (4) pode ser, facilmente, encontrada numericamente, mediante o seguinte *script* desenvolvido em Linguagem MATLAB de Computação:

```
% Cálculo da Equação (4)
% Variável simbólica: P
syms P
% Coeficientes da Equação (2)
p1 = 0.3641; p2 = 2.0950; p3 = 4.5020;
% Equação (2)
Q = p1*P.^2 + p2*P + p3;
% Resolvendo a Derivada dQ/dP
derivada = diff(Q);
a = Q*sqrt(1+derivada^2);
% Encontrando a Área de ação da DPE
area1 = 2*pi*int(a)
```

O volume de distribuição da DPE pode ser determinado pela Equação (5)

$$V = \int_a^b \pi [Q(P)]^2 dP \quad (5)$$

Sendo, a e b os valores máximo e mínimo da Potência Ativa.

É possível encontrar a solução numérica da Equação (5), por meio do seguinte *script*, implementado em Linguagem MATLAB de Computação:

```
% Cálculo da Equação (5)
% Variável simbólica: P
syms P
% Coeficientes da Equação (2)
p1 = 0.3641; p2 = 2.0950; p3 = 4.5020;
% Equação (2)
Q = p1*P.^2 + p2*P + p3;
% Função Volume da DPE
Fv = pi*Q^2;
% Resolução da Integral
V = int(Fv)
```



RESULTADOS E DISCUSSÃO

Devido a Potência Aparente, exposta na Equação (1), ser uma função de múltiplas variáveis, $P_{ap}(P, Q)$, o seu ajuste foi estimado por uma superfície, cujo acerto evidencia uma precisão de 99,07% e acurácia 98,65%.

A Figura 2 exibe o comportamento, simultâneo, das Potências Elétricas do sistema de 14 barras. Nela estão disponíveis as variações da P_{ap} em relação às variações de P e Q .

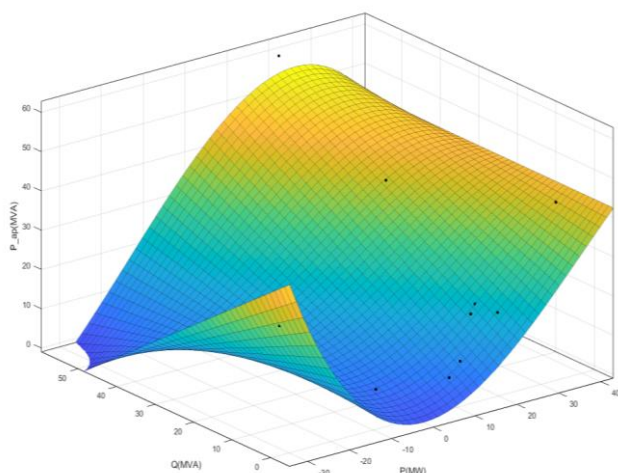


Figura 2. Superfície de ajuste dos dados das Potências.

Com o propósito de maximizar a análise promovida pelo Método de Regressão Múltipla Não-Linear (MRMNL), a superfície engendrada será visualizada por diferentes perspectivas. Essas ilustrações serão retratadas nas Figuras 3, 4, 5, 6, 7 e 8.

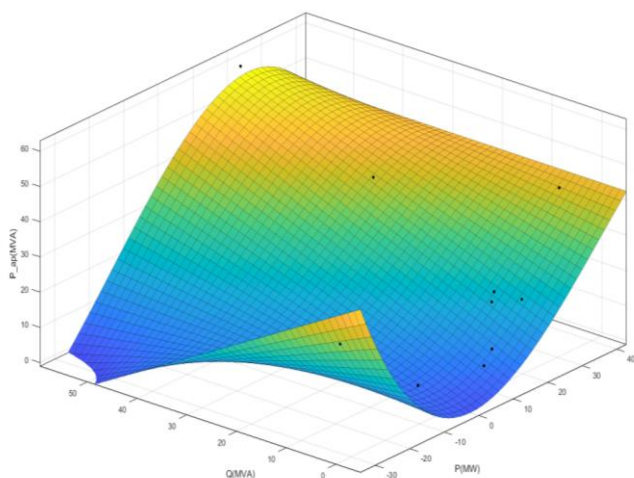


Figura 3. Superfície de ajuste – Perspectiva 2.

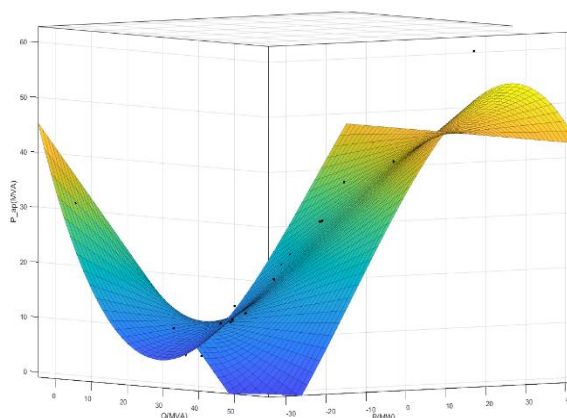


Figura 4. Superfície de ajuste – Perspectiva 3.

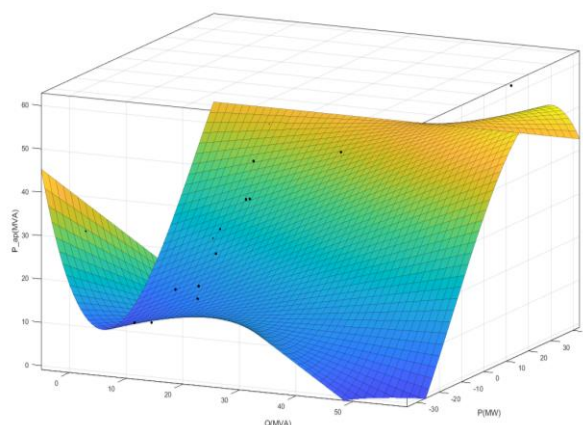


Figura 5. Superfície de ajuste – Perspectiva 4.

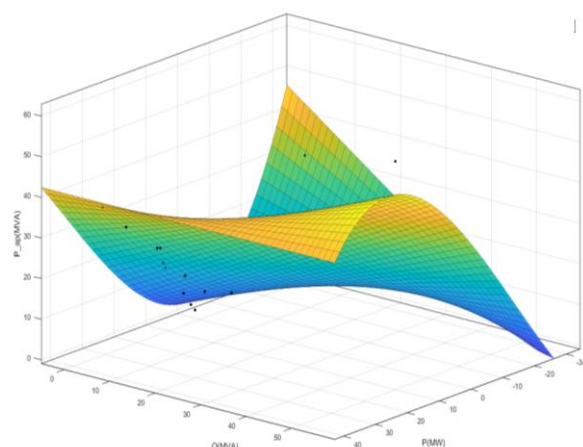


Figura 6. Superfície de ajuste – Perspectiva 5.

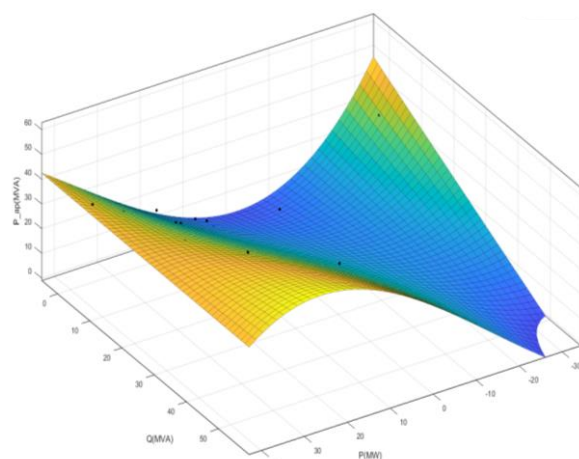


Figura 7. Superfície de ajuste – Perspectiva 6.

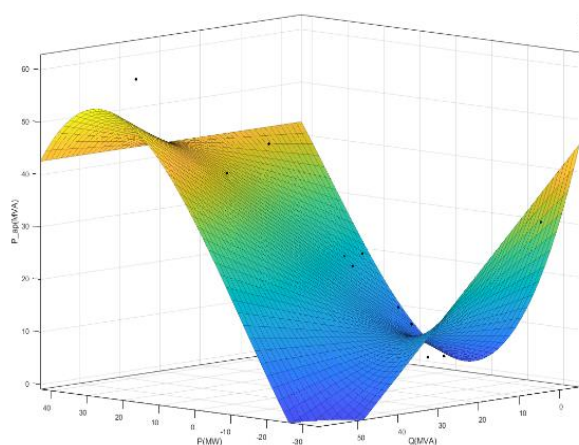


Figura 8. Superfície de ajuste – Perspectiva 7.

A Figura 9 expõe a relação $Q = f(P)$, conforme a Equação (2).

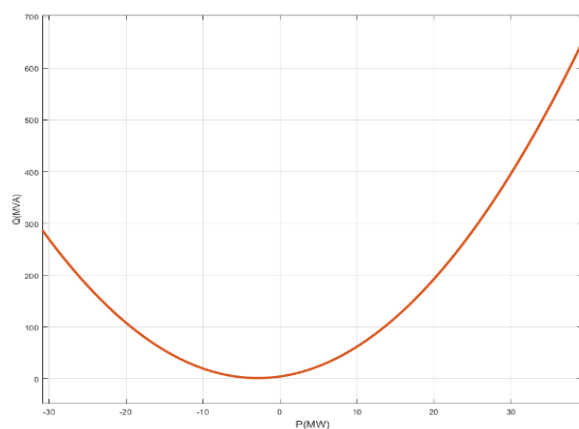


Figura 9. Função de DPE.

A função de DPE estabelece o campo de ação da Potência Elétrica. Essa densidade é explicada pela estrutura gerada pela revolução da Equação (2).

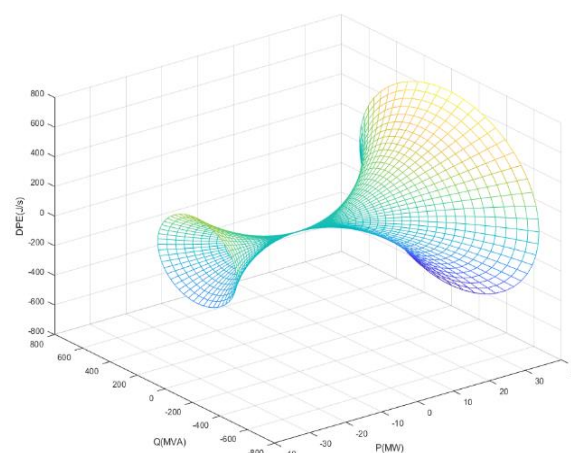


Figura 10. Campos de ação da DPE – Perspectiva 1.

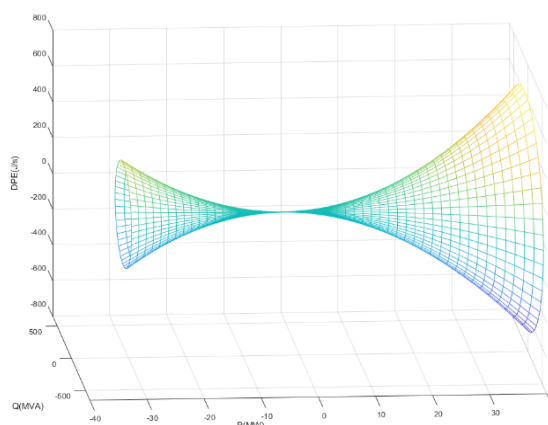


Figura 11. Campos de ação da DPE – Perspectiva 2.

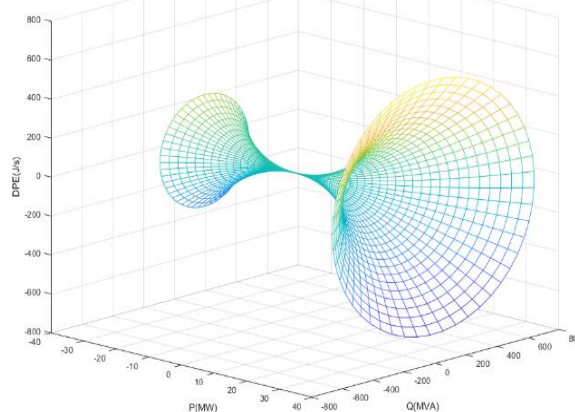


Figura 12. Campos de ação da DPE – Perspectiva 3.

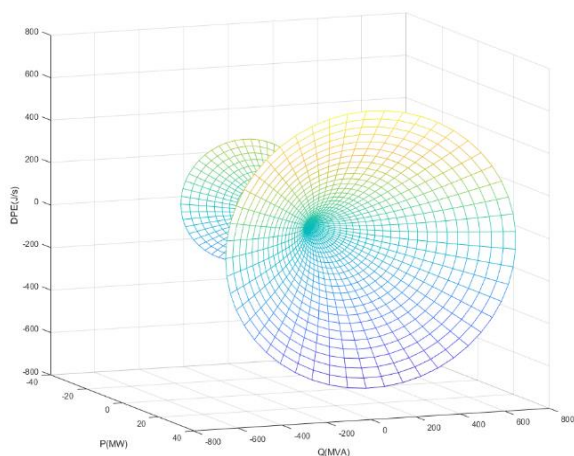


Figura 13. Campos de ação da DPE – Perspectiva 4.

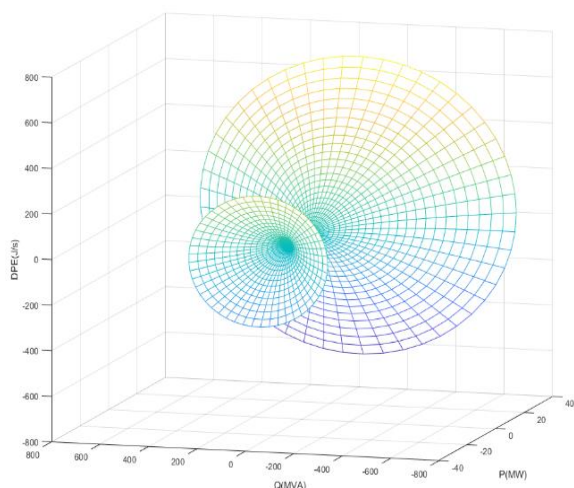


Figura 14. Campos de ação da DPE – Perspectiva 5.

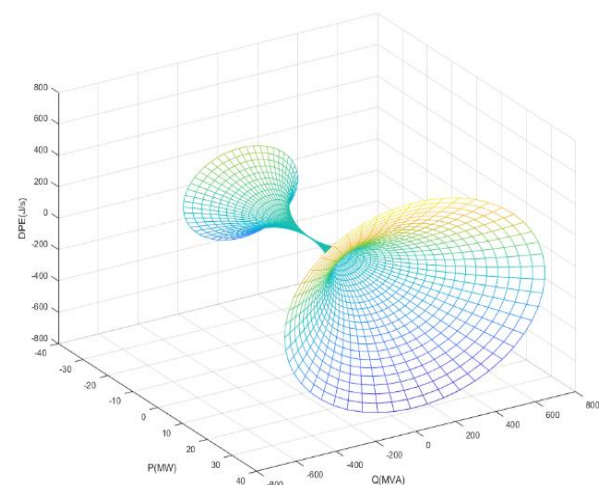


Figura 15. Campos de ação da DPE – Perspectiva 6.

Aplicando a Equação (2) na Equação (4), determina-se o valor da área de ação da DPE. Para os valores estudados nessa pesquisa, tem-se

$$A = 1.5987 \times 10^6 \text{ unidade}^2$$

Empregando a Equação (2) na Equação (5), determina-se o valor do volume de ação da DPE. Para os valores estudados nessa pesquisa, tem-se

$$V = 1.2809 \times 10^7 \text{ unidade}^3$$

Para estimar o valor de maior concentração da DPE, utiliza-se as Equações (2) e (3).

$$DPE = 16.8631 \text{ J/s}$$

Esse valor indica que a DPE apresenta maior ação nas adjacências da barra 6.

As representações geométricas apresentadas nas Figuras 2, 3, 4, 5, 6, 7 e 8, proporcionam o comportamento, sincrônico, das Potências Elétricas Ativa, Reativa e Aparente. Essas imagens comprovam os valores dessas potências em cada ponto do Sistema Elétrico de 14 barras, conforme citam Costa et al (2022).

As Figuras 8, 9, 10, 11, 12, 13, 14 e 15 reiteram como a Densidade de Potência Elétrica está distribuída. Elas evidenciam que a maior concentração da DPE está correlata com a sua estrutura geométrica.

CONCLUSÃO

As análises sobre a segurança das Redes Elétricas, considerando cenários de operação dos Sistemas de Energia, demonstram estudos com crescente interdisciplinaridade: Matemática, Física e Computação são exemplos de conteúdos acadêmicos aplicados nas modelagens desses sistemas.

A formulação matemática do problema apresentado nesse artigo consiste na aplicação das equações, frequentemente, usadas para o cálculo do centroide. Esses modelos matemáticos, implementados em linguagem computacional, foram adaptados com a intenção de determinar a Densidade de Potência Elétrica, seu ponto de maior concentração e a sua região de ação.

Os resultados demonstraram magnífica precisão, 99,07% e excelente acurácia, 98,65%. A partir dessa estratégia e dos excepcionais resultados, estima-se o aprimoramento dos Sistemas de Energia, propondo como futuros trabalhos, a otimização das Potências Elétricas.

AGRADECIMENTOS

Ao Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA Campus Ananindeua, pela



administração do Bacharelado em Ciência e Tecnologia desse *campus*. E ao grupo de pesquisa Gradiente de Modelagem Matemática e Simulação Computacional – GM²SC, pela gestão e motivação científica de Discentes e Docentes.

REFERÊNCIAS

BELARDINELLI, Cyril. Linear mass density of vertically suspended heavy springs. Cornell University, Physics: Classical Physics. <https://doi.org/10.48550/arXiv.1909.08689>. 2019.

BUKENBERGER, Dennis R. e LENSCH, Hendrik P. A. Tetrahedra of varying density and their applications. *The Visual Computer*. 37:2447–2460 <https://doi.org/10.1007/s00371-021-02189-0>. 2021.

COSTA, Denis C. L., NUNES, Marcus V.A., VIEIRA, João P.A. & BEZERRA, Ubiratan H. Decision tree-based security dispatch application in integrated electric power and natural-gas networks. *Electric Power Systems Research* 14. 442–449. 2016.

COSTA, Denis. C. L.; COSTA, Heictor. A. de O.; CASTRO, Ana P. S.; CRUZ, Edson C.; AZANCORT NETO, Júlio L.; CRUZ, Brennus C. C. da. As dimensões das Modelagens Matemática e Computacional prescrevidas à Gestão Ambiental. *Research, Society and Development, [S. l.]*, v. 9, n. 10, p. e6939109013. DOI: 10.33448/rsd-v9i10.9013. 2020.

COSTA, Denis C. L.; MENESES, L. A. de; LIMA, M. L. V. de; COSTA, Heictor A. de O.; REIS, A. C. F.; PINHEIRO, H. F. B.; COSTA, E. F. da; SILVA, A. R. dos S. da; REIS, A. C. F.; RAIOL, F. M.; SANTOS, R. C. P. dos. Thermoelectric generation with reduced pollutants made possible by bio-inspired computing. *Research, Society and Development, [S. l.]*, v. 11, n. 1, p. e7611124568, 2022. DOI: 10.33448/rsd-v11i1.24568. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/24568>. 2022a.

COSTA, Denis C. L. et al. A Interdisciplinaridade entre as Leis de Kirchhoff, Sistemas Lineares e Lógica de Programação. In: *Anais do Seminário Internacional de Linguagens, Culturas, Tecnologias e Inclusão*. Anais. Castanhal (PA) IFPA, 2022. DOI: 10.29327/163666.2-22. Disponível em: < <https://www.even3.com.br/anais/iisilicti/479506-a-interdisciplinaridade-entre-as-leis-de-kirchhoff-sistemas-lineares-e-logica-de-programacao/> >. 2022b.

COSTA, Heictor A. de O. et al. A Modelagem Matemática-Computacional Potencializando a Gestão Energética. In: *Anais do Seminário Internacional de Linguagens, Culturas, Tecnologias e Inclusão*. Anais Castanhal (PA) IFPA, 2022. DOI:

10.29327/163666.2-23. Disponível em: < <https://www.even3.com.br/anais/iisilicti/480582-a-modelagem-matematica-computacional-potencializando-a-gestao-energetica/> >. 2022.

DREßLER, Christian; SCHERRER, Arne; AHLERT, Paul e SEBASTIANI, Daniel. Efficient Representation of the Linear Density-Density Response Function. *Journal of Computational Chemistry* published by Wiley Periodicals, Inc. DOI: 10.1002/jcc.26046. 2019.

DREßLER, Christian; SEBASTIANI, Daniel. Polarization Energies from Efficient Representation of the Linear Density–Density Response Function. *Advanced Theory and Simulations*. 2021, 4, 2000260. DOI: 10.1002/adts.202000260. 2021.

PUGA, L. Z. Revolution Solids in Engineering Course with Matlab. *Research, Society and Development, [S. l.]*, v. 4, n. 2, p. 102-115. DOI: 10.17648/rsd-v4i2.72. 2017.

PSTCA. Power Systems Test Case Archive. Available: <http://www.ee.washington.edu/research/ps-tca/>. 1993.

VIGNALE, Giovanni. Sum rule for the linear density response of a driven electronic system. *Physics Letters* - Elsevier. 206-210. 1995.



ANÁLISE CINÉTICA ENZIMÁTICA IMPLEMENTADA PELA MATEMÁTICA-COMPUTACIONAL

**Adriane Cristina Fernandes Reis¹, Denis Carlos Lima Costa²,
Mayra Xavier dos Santos³, André Renan dos Santos da Silva⁴,
Erick Freitas da Costa⁵, David Daniel Lira de Santana⁶**

¹Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (adriane.fr22@gmail.com)

²Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (denis.costa@ifpa.edu.br)

³Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (xaviermayra01@gmail.com)

⁴Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (andre.renan13@hotmail.com)

⁵Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (efreitas256@gmail.com)

⁶Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (davidsantana1113@gmail.com)

Resumo: O presente trabalho faz um ensaio sobre o movimento das enzimas e o reconhecimento de padrões dessa cinética. A pesquisa é fundamentada na equação de Michaelis-Menten e aponta outros modelos para ajustar a relação entre a velocidade inicial da reação em função da concentração do substrato. São indicados os modelos Linear, Quadrático, Cúbico, Exponencial, de Gauss, Potência e de Fourier. Os resultados obtidos à acurácia foram sempre superiores a 90% e maiores que 80% para a precisão.

Palavras-chave: Cinética Química; Enzimas; Modelagem Matemática-Computacional; Função de Gauss; Função de Fourier.

INTRODUÇÃO

As enzimas são proteínas catalíticas onipresentes nos sistemas vivos e desempenham função essencial em, praticamente, todos os processamentos celulares, como, por exemplo, o metabolismo, o transporte ativo, o sensoramento, a regulação, a comunicação e transdução e a integração de sinais (Wang et al 2020).

As enzimas atuam como catalisadores para acelerar a taxa de reações químicas nas células vivas. Na maioria dos casos, eles convertem um produto químico, o substrato, em outro, o produto (Abis et al, 2019).

A maior parte dos ensaios enzimáticos mensura as concentrações de substrato, e/ou produto, em relação ao tempo (Stroberg and Schnell, 2016).

As taxas de desaparecimento, e/ou aparecimento, que viabilizam a velocidade da enzima, podem ser estipuladas por modelos matemáticos, como o apresentado por Michaelis-Menten (1913), destacado na Equação (1)

$$v = \frac{v_m[S]}{k_s + [S]} \quad (1)$$

Em que,

$v \rightarrow$ é a velocidade inicial da reação;

$v_m \rightarrow$ é a velocidade inicial máxima da reação;

$k_s \rightarrow$ é a constante de meia saturação;

$[S] \rightarrow$ é a concentração do substrato.

Este trabalho apresenta um conjunto de sistemas matemáticos capaz de estimar a velocidade inicial da reação em função da concentração do substrato. Esses sistemas foram desenvolvidos a partir da Modelagem Matemática-Computacional fundamentada nos modelos Linear, Quadrático, Cúbico, Exponencial, de Gauss, Potência e de Fourier.

As expressões obtidas proporcionam um elevado nível de segurança e estabilidade das soluções, pois conferem acurácia e precisão acima de 80%.



MATERIAL E MÉTODOS

As estratégias utilizadas para modelar a relação $v = f(S)$ estão amparadas no Método de Regressão Linear (MRL) e Método de Regressão Não-Linear (MRNL), destacados por Costa et al (2020).

Apresenta-se uma matriz de critérios em que S_i representa a variável independente e v_i , representa a variável dependente, uma vez que $v_i = f(S_1, S_2, S_3, \dots, S_i)$, sendo $i = 1, 2, 3, \dots, 7$.

A Tabela 1 apresenta os valores dos pares ordenados empregados no reconhecimento do melhor padrão de comportamento entre a velocidade inicial da reação (v) e a concentração do substrato ($[S]$).

Tabela 1. Valores de $[S]$ e v .

Concentração - $[S]$	Velocidade inicial - (v)
1.30	0.070
1.80	0.130
3.00	0.220
4.50	0.275
6.00	0.335
8.00	0.350
9.00	0.360

Fonte: Chapra, 2018.

A linguagem de programação escolhida para realizar a Modelagem Matemática-Computacional, foi a MATLAB – MATrix LABoratory ((Mathworks, 2021). Para Costa et al (2021), essa linguagem usa, sistematicamente, os conceitos matemáticos elementares de Funções Polinomiais e Não-Polinomiais, Matrizes, Álgebra Linear, Cálculo Diferencial e Integral. Esses conceitos são essenciais para a implementação das simulações computacionais.

O gráfico da Figura 1 mostra como os valores de $[S]$ e v , expostos na Tabela 1, comportam-se.

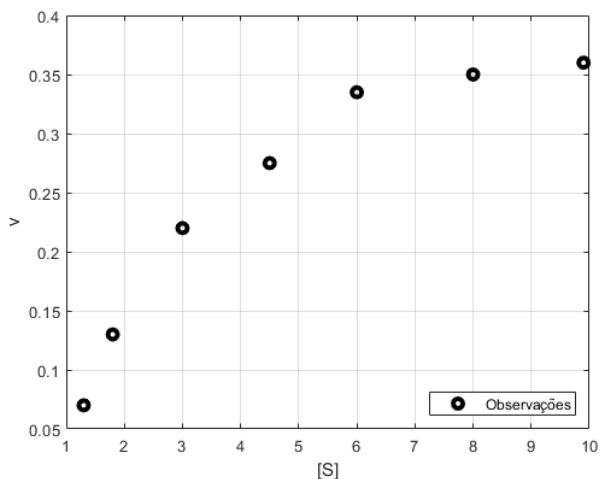


Figura 1. Valores observados de $[S]$ e v .

A partir da incursão das informações, presentes na Tabela 1, em ambiente MATLAB, foi possível estimar as relações funcionais que vinculam $[S]$ e v . As melhores funções encontradas foram: Função Linear, Função Quadrática, Função Cúbica, Função Exponencial, Função de Gauss, Função Potência e Função de Fourier.

RESULTADOS E DISCUSSÃO

O primeiro modelo determinado foi uma Função Linear, expressada na Equação (2).

$$v(S) = 0.03538[S] + 0.06693 \quad (2)$$

Os valores de acurácia e precisão do modelo Linear são, respectivamente, $ac = 92.22\%$ e $pr = 83.53\%$. A representação gráfica do modelo Linear está exposta na Figura 2.

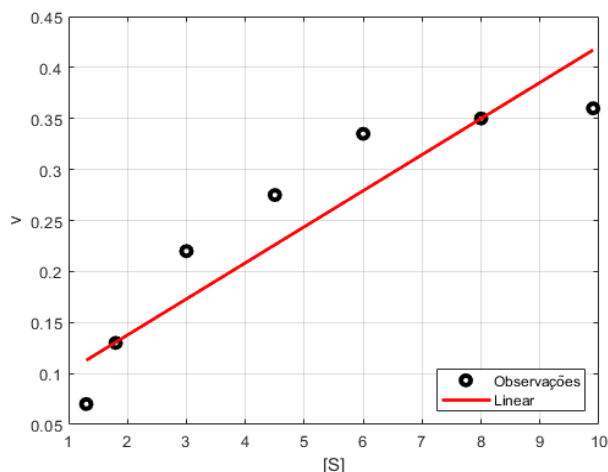


Figura 2. Modelo Linear para $v = f(S)$.

O segundo modelo determinado foi uma Função Quadrática, expressada na Equação (3).

$$v(S) = -0.005545[S]^2 + 0.0931[S] - 0.02659 \quad (3)$$

A representação gráfica do modelo Quadrático está exposta na Figura 3.

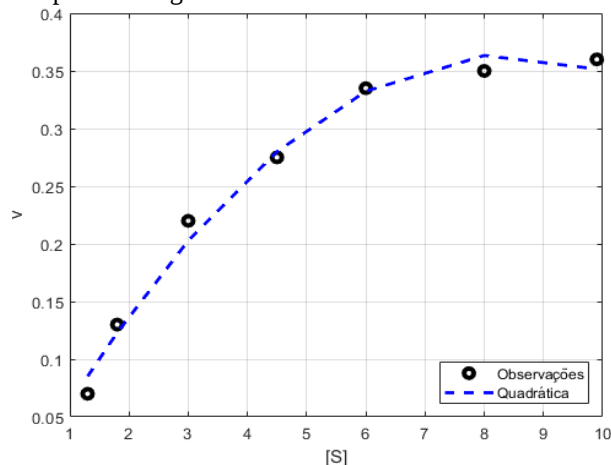


Figura 3. Modelo Quadrático para $v = f(S)$.



Os valores de acurácia e precisão do modelo Quadrático são, respectivamente, $ac = 92.36\%$ e $pr = 98.89\%$.

O terceiro modelo determinado foi uma Função Cúbica, expressada na Equação (4).

$$v(S) = 0.0005768[S]^3 - 0.01522[S]^2 + 0.1386[S] - 0.07987 \quad (4)$$

A representação gráfica do modelo Cúbico está exposta na Figura 4.

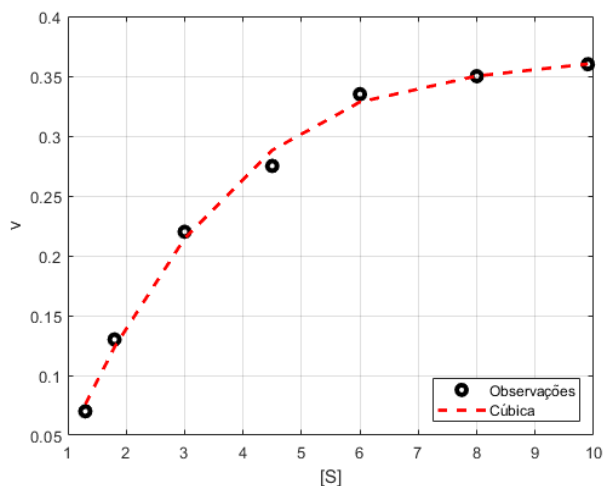


Figura 4. Modelo Cúbico para $v = f(S)$.

Os valores de acurácia e precisão do modelo Cúbico são, respectivamente, $ac = 92.27\%$ e $pr = 99.58\%$.

O quarto modelo determinado foi uma Função Exponencial, expressada na Equação (5).

$$v(S) = 0.04098e^{-0.008505[S]} - 0.5239e^{-0.3513[S]} \quad (5)$$

A representação gráfica do modelo Exponencial está exposta na Figura 5.

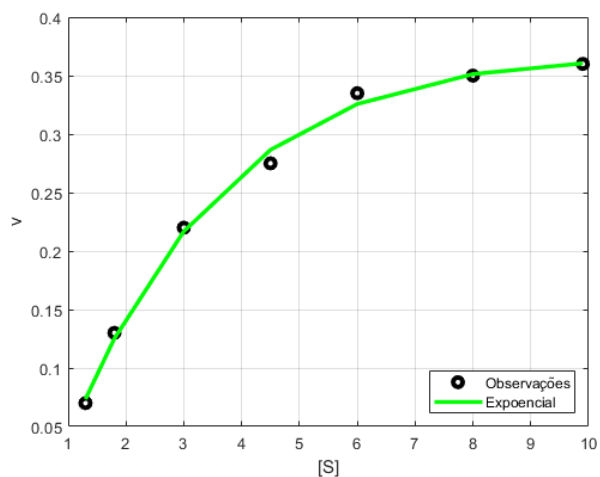


Figura 5. Modelo Exponencial para $v = f(S)$.

Os valores de acurácia e precisão do modelo Exponencial são, respectivamente, $ac = 92.26\%$ e $pr = 99.66\%$.

O quinto modelo determinado foi uma Função de Gauss, expressada na Equação (6).

$$v(S) = 0.3593e^{-\left(\frac{[S]-9.458}{4.956}\right)^2} + 0.1627e^{-\left(\frac{[S]-4.068}{2.864}\right)^2} \quad (6)$$

A representação gráfica do modelo de Gauss está exposta na Figura 6.

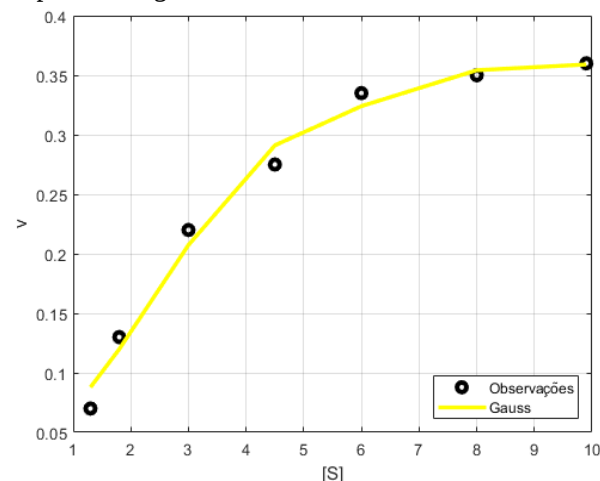


Figura 6. Modelo de Gauss para $v = f(S)$.

Os valores de acurácia e precisão do modelo de Gauss são, respectivamente, $ac = 92.50\%$ e $pr = 98.69\%$.

O sexto modelo determinado foi uma Função Potência, expressada na Equação (7).

$$v(S) = -0.5825[S]^{-0.4293} + 0.5869 \quad (7)$$

A representação gráfica do modelo de Potência está exposta na Figura 7.

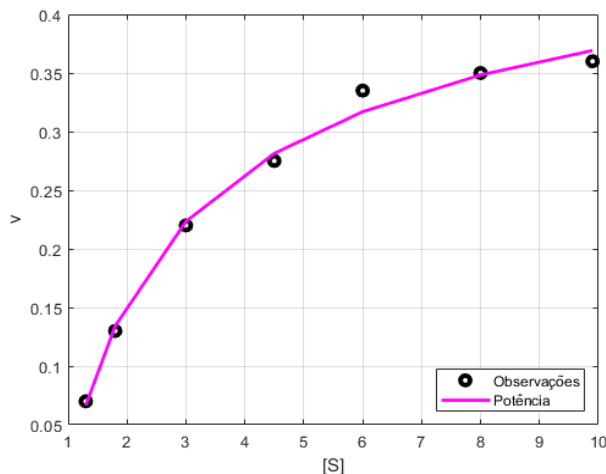


Figura 7. Modelo de Potência para $v = f(S)$.

Os valores de acurácia e precisão do modelo de Potência são, respectivamente, $ac = 92.29\%$ e $pr = 99.36\%$.



O sétimo modelo determinado foi uma Função de Fourier, expressada na Equação (8).

$$v(S) = -3.181 \times 10^6 + 3.181 \times 10^6 \cos(-5.91 \times 10^{-5}[S]) - 1579 \sin(-5.91 \times 10^{-5}[S]) \quad (8)$$

A representação gráfica do modelo de Fourier está exposta na Figura 8.

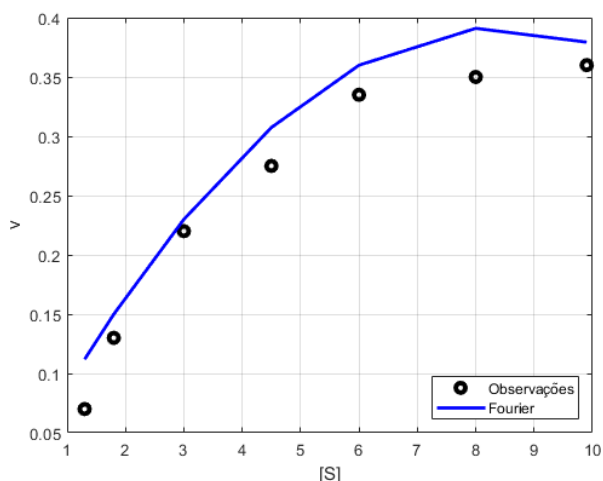


Figura 8. Modelo de Fourier para $v = f(S)$.

Os valores de acurácia e precisão do modelo de Fourier são, respectivamente, $ac = 92.31\%$ e $pr = 92.21\%$.

O gráfico da Figura 9 exibe a comparação entre todos os modelos gerados, com base nas observações declaradas na Tabela 1, baseados na metodologia Matemática-Computacional disponível na Linguagem MATLAB de Computação, cujo comando é dado por

$$cftool(S, v)$$

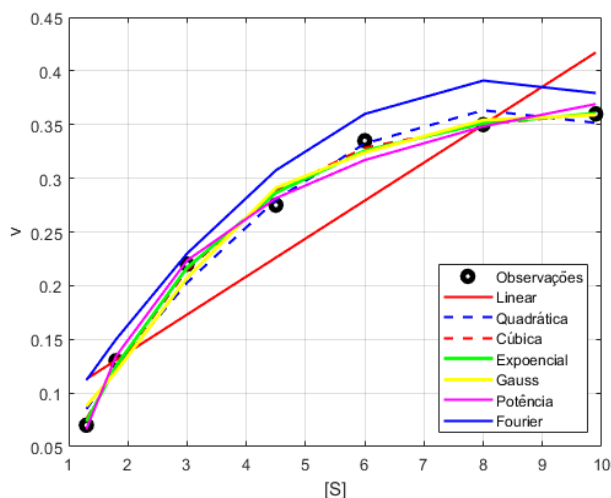


Figura 9. Comparação dos modelos estimados para $v = f(S)$.

Em que,

$v \rightarrow$ é a velocidade inicial da reação;

$[S] \rightarrow$ é a concentração do substrato.

Chapra (2018), apresenta o valor de $v_m = 0.4083$ à velocidade inicial máxima da reação e $k_s = 2.8127$ à constante de meia saturação. Segundo o autor, embora o modelo Michaelis-Menten forneça um bom ponto de partida, ele foi refinado e estendido para incorporar recursos adicionais de cinética enzimática.

A Figura 10 exibe a representação gráfica do modelo de Michaelis-Menten.

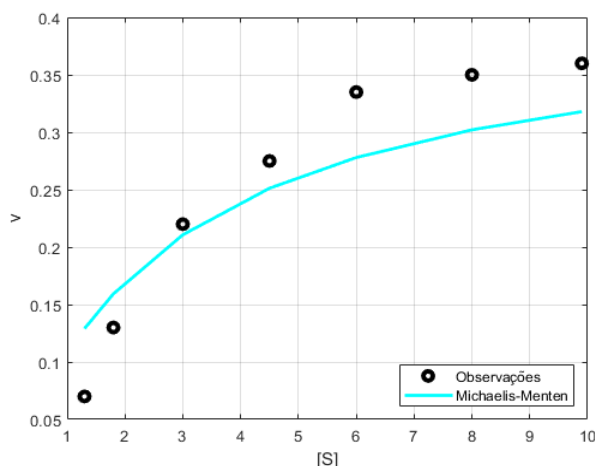


Figura 10. Modelo de Michaelis-Menten para $v = f(S)$.

A Tabela 2 apresenta os valores da acurácia e da precisão, calculados para cada modelo.

Tabela 2. Estabilidade e segurança das estimações.

Modelos	Acurácia (%)	Precisão (%)
Michaelis-Menten	96.90	60.33
Linear	92.22	83.53
Quadrático	92.36	98.89
Cúbico	92.27	99.58
Exponencial	92.26	99.66
Gauss	92.50	98.69
Potência	92.29	99.36
Fourier	92.31	92.21

Todos os modelos encontrados estão em uma escala superior a 90% para a acuraria e 80% para a precisão. Ou seja, excelentes estimações. Entretanto, o modelo representado pela Função Exponencial possibilita o melhor ajuste entre os demais, tendo em vista que esse



modelo apresenta a menor média dos resíduos para os seus valores estimados.

CONCLUSÃO

A análise da Cinética Química é essencial para estudos de fase gasosa, fase condensada e cinética de reação de polímeros, bem como cinética bioquímica e de superfície. Modelos capazes de realizar medições experimentais cuidadosas de cinética de reação, em sistemas simples e complexos são sempre aprimorados com os excelentes arranjos entre a Matemática e a Computação.

Este artigo se propôs a publicar modelos matemáticos, fundamentados em Função Linear, Função Quadrática, Função Cúbica, Função Exponencial, Função de Gauss, Função Potência e Função de Fourier, qualificados a interpretar o comportamento da Cinética Enzimática, em sua relação entre velocidade inicial da reação e a concentração do substrato.

O processo desenvolvido na pesquisa beneficiou-se dos métodos matemáticos, implementados em Linguagem MATLAB de Computação. Os resultados obtidos demonstram que, apesar de o modelo de Michaelis-Menten exibir uma excepcional interpretação, os modelos manifestados neste trabalho evidenciaram um melhor conjunto entre a acurácia e a precisão, à predição segura e estável, de novos valores da relação $v = f(S)$.

Com consequência desses resultados, recomenda-se para próximas pesquisas, a formulação de modelos que determinem os valores da velocidade inicial máxima da reação e da constante de meia saturação, mais precisos e estáveis.

AGRADECIMENTOS

Ao Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA *Campus Ananindeua*, pela administração do Bacharelado em Ciência e Tecnologia desse *campus*. E ao grupo de pesquisa Gradiente de Modelagem Matemática e Simulação Computacional – GM²SC, pela gestão e motivação científica de Discentes e Docentes.

REFERÊNCIAS

ABIS, G.; PACHECO-GÓMEZ, R.; BUI, T. T. T.; CONTE, M. R. Isothermal titration calorimetry enables rapid characterization of enzyme kinetics and inhibition for the human soluble epoxide hydrolase. *Anais Chem.* 91, 14865–14872. DOI: 10.1021/acs.analchem.9b01847. 2019.

COSTA, Denis. C. L.; COSTA, Heictor. A. de O.; CASTRO, Ana P. S.; CRUZ, Edson C.; AZANCORT NETO, Júlio L.; CRUZ, Brennus C. C. da. As dimensões das Modelagens Matemática e Computacional prescrevidas à Gestão Ambiental. *Research, Society and Development*, [S. l.], v. 9, n. 10, p. e6939109013. DOI: 10.33448/rsdv9i10.9013. 2020.

COSTA, Denis C. L.; COSTA, Heictor A. de O.; SILVA, Hugo C. M. da. SILVA, Silvio T. T. da. *Matemática Computacional Aplicada à Ciência e Tecnologia*. 1st ed. Belém-PA, SINEPEM-IFPA. 2021.

CHAPRA, Steven C. *Applied Numerical Methods with MATLAB for Engineers and Scientists*. Berger Chair in Computing and Engineering, Tufts University. Fourth edition. New York, NY: McGraw-Hill Education. 2018.

MATHWORKS. MathWorks leading developer of Mathematical Computing software for Engineers and Scientists. <https://www.mathworks.com/>. 2021.

MICHAELIS, L. e MENTEN, M. Die kinetik der invertinwirkung, *Biochemistry Zeitung* 49, 333-369. 1913.

STROBERG, W.; SCHNELL, S. On the estimation errors of K_M and V from time-course experiments using the Michaelis-Menten equation. *Biophys Chem.* 219:17-27. DOI: 10.1016/j.bpc.2016.09.004. PMID: 27677118. 2016.

WANG, Yun; WANG, Guanyu; MOITESSIER, Nicolas; MITTERMAIER, Anthony K. Enzyme Kinetics by Isothermal Titration Calorimetry: Allostery, Inhibition, and Dynamics. *Frontiers in Molecular Biosciences | Biological Modeling and Simulation*. <https://doi.org/10.3389/fmolb.583826>. 2020.



INTELIGÊNCIA DE ENXAME APLICADA NA OTIMIZAÇÃO DE SISTEMA DE ENERGIA

Heictor Alves de Oliveira Costa¹, Denis Carlos Lima Costa², Ariane Cristina Fernandes Reis³, Fernando José Aguirre Ramos da Silva⁴, Huan F. Brasil Pinheiro⁵, Rosângela dos Santos Azevedo⁶, Lair Aguiar de Meneses⁷, Aguinaldo de Jesus Moraes Marques⁸

¹Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (heictor.costa@itec.ufpa.br)

²Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (denis.costa@ifpa.edu.br)

³Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (arianecfr@gmail.com)

⁴Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (aguirrefernando.ifpa@gmail.com)

⁵Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (huanbrasil@gmail.com)

⁶Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (992120536rosangela@gmail.com)

⁷Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (lair.meneses@ifpa.edu.br)

⁸Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – GM²SC
Ananindeua, Brasil (aguinaldoj2m@gmail.com)

Resumo: A progressiva evolução dos Sistemas Tecnológicos impulsiona a crescente necessidade por tomadas de decisão imediatas e seguras. Diante disso, este artigo preconiza a Otimização por Enxame de Partículas (*Particle Swarm Optimization - PSO*), como um mecanismo capaz de encontrar o melhor conjunto de soluções de modelos matemáticos relacionados aos Sistemas de Energia. A estratégia foi implementada em Linguagem MATLAB e comprovada mediante os dados do sistema elétrico de 14 barras.

Palavras-chave: Otimização; Enxame de Partículas; Sistema de Energia; Potência Elétrica.

INTRODUÇÃO

A Computação Bioinspirada está, cada vez mais, intrínseca em nossa realidade. Sistemas Inteligentes administrados por Redes Neurais Artificiais, Algoritmo Genético, Árvore de Decisão, entre outros, têm suas aplicações expandidas para inúmeras áreas da Ciência e Tecnologia, ampliando, desta maneira, as funcionalidades da Inteligência Artificial.

Costa et al (2016), por exemplo, apresentam um Sistema Integrado das Redes Elétrica e de Gás Natural, cuja segurança foi estimada pela estratégia de Árvore de Decisão.

Ainda sobre Costa et al (2016), os autores otimizaram o despacho seguro do Sistema Integrado de Redes, utilizando, para isso, o Algoritmo Genético.

Costa et al (2021a), usaram a interdisciplinaridade como critério para a otimização de Redes Elétrica e de Gás Natural, viabilizada pelo Algoritmo Genético.

Costa et al (2021b), apresentam o Cálculo Diferencial de Ordem Não-Inteira, combinado com a técnica de Árvore de Decisão, como ferramenta destinada ao gerenciamento de Redes Inteligentes.

Costa et al (2021c), em seu artigo, expõem uma metodologia que associa os mecanismos do Algoritmo Genético e o do Enxame de Partículas, tendo como finalidade otimizar o Sistema Elétrico.

Costa et al (2022), aplicaram a técnica do Algoritmo Genético com o objetivo de reduzir a quantidade de poluentes provocada pela geração de energia elétrica em termoelétricas a gás natural.

Esse artigo visa, como principal propósito, a otimização de um Sistema Energia, empregando a metodologia Bioinspirada de Enxame de Partículas, considerando que a função *fitness* é uma função de múltiplas variáveis que correlaciona, isocronicamente,



a Potência Ativa (P), a Potência Reativa (Q) e a Potência Aparente (P_{ap}).

Para realização deste objetivo, a pesquisa se beneficiou da Modelagem Matemática-Computacional dissertada por Costa et al (2021c) implementada em Linguagem MATLAB de Computação.

MATERIAL E MÉTODOS

O caso base, utilizado nesta pesquisa, refere-se ao Sistema de 14 barras do IEEE – *Institute of Electrical and Electronics Engineers*. A Figura 1 representa este sistema, que considera a barra 1 como barra de referência (PSTCA 1993).

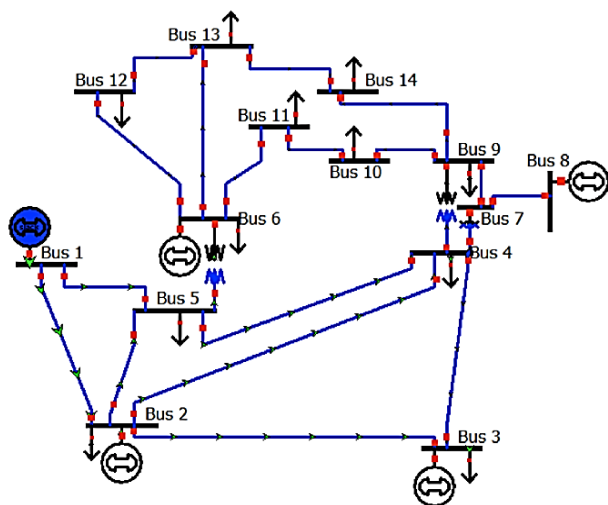


Figura 1. Rede Elétrica – Sistema de 14 barras.

Essa rede elétrica dispõe de um conjunto de Potências Elétricas que pode ser estimado mediante a aplicação da Equação (1)

$$P_{ap}(P, Q) = 16.76 + 12.97P + 8.676Q + 2.564P^2 - 0.2088PQ - 1.174P^3 - 2.325P^2Q \quad (1)$$

Sendo P_{ap} a Potência Aparente (MVA), P a Potência Ativa (MW) e Q a Potência Reativa (MVA).

A Equação (1) será aplicada, no algoritmo computacional, como função objetivo ou função *fitness*. A partir da incorporação das restrições do Sistema Elétrico de Potência, será possível encontrar os valores ótimos para o conjunto de Potências Elétricas.

Neste trabalho a *Particle Swarm Optimization* (PSO) ou Otimização por Enxame de Partículas (OEP) é a estratégia, Bioinspirada, implementada para a determinação das melhores Potências Elétricas.

A OEP é uma metodologia metaheurística de população cooperadora, elaborada com a finalidade de

simular o comportamento de espécies animais que executam tarefas em grupo, tendo como fundamentação o paradigma social retratado por J. Kennedy e R. Eberhart (1995).

Para Sanches et al (2012), a metaheurística é uma técnica inteligente que, mediante um conjunto de etapas definidas, é eficiente para lidar com problemas matemáticos de otimização. Essa magnífica ferramenta, lida de forma eficaz com informações aproximadas, e/ou imprecisas, tendo como objetivo, encontrar uma solução de qualidade para o problema investigado.

O fluxograma na Figura 2 ilustra como a OEP foi implementado neste trabalho.

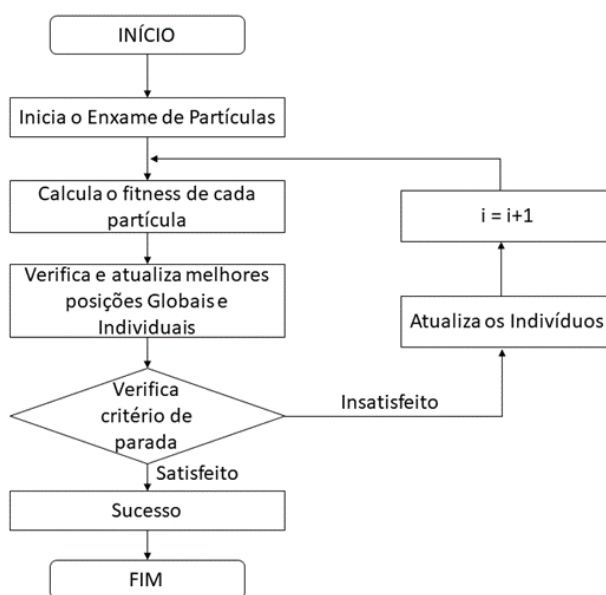


Figura 2. Funcionamento da Otimização do Enxame de Partículas (Adaptado de Costa et al, 2021c).

A Figura 2 mostra a sequência de passos da OEP. As principais fases estão relacionadas a uma estrutura de repetição *for*, que termina assim que o critério de parada for alcançado.

Nesta pesquisa, a Inteligência Artificial (IA) foi treinada para interromper a estrutura de repetição quando o número de gerações, previamente estabelecido, for atendido. Desta forma, A IA será capaz de atuar em um intervalo de tempo mais previsível, uma vez que, depender da OEP para obter uma alta precisão no valor de *fitness* (aptidão) demandaria um intervalo de tempo impreciso para o processamento, e isso prejudicaria seu propósito de otimização do processo.

Os valores da Potência Ativa (P), da Potência Reativa (Q) e da Potência Aparente (P_{ap}), evidenciados na Tabela 1, foram utilizados na modelagem da Equação (1). Esses valores são as partículas que compõem o



enxame, inserido no código-fonte, representado pela Figura 2.

Cada partícula representa um indivíduo na população e cada um tem um valor de coordenada em cada uma das dimensões no espaço de domínio.

Tabela 1. Valores das Potências Elétricas.

Barra	P(MW)	Q(MVA)	P _{ap} (MVA)
01	19,68	56,67	59,99
02	30,10	22,52	37,59
03	20,09	0,27	20,09
04	39,41	6,13	39,88
05	32,24	6,23	32,84
06	17,76	4,72	18,38
07	-30,88	1,49	30,92
08	24,77	7,86	25,99
09	14,32	6,66	15,79
10	22,36	31,76	38,84
11	10,66	-0,26	10,66
12	8,12	1,98	8,36
13	19,42	5,38	20,15
14	-0,00	-2,07	2,07

Fonte: PSTCA, 1993.

Essas partículas são caracterizadas, na dimensão computacional, por múltiplas variáveis, tais como $[a_1, a_2, \dots, a_n]$. Isso permite que as partículas sejam interpretadas como vetores, sendo que, todo vetor de partícula deverá ser formatado por suas posições e velocidades relacionadas.

A população é iniciada como um aglomerado de partículas, uniformemente, distribuídas por todo o espaço de busca, com valores aleatórios dentro dos intervalos delimitados, conforme indica a Equação (2).

$$a_n x = \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,m} \end{bmatrix}$$

$$v_{i+1} = wv_i + c_1 \cdot r_1 \cdot (p_i - x_i) + c_2 \cdot r_2 \cdot (g - x_i) \quad (2)$$

Sendo, $a_n x$ espaço de busca da n – ésima partícula a , na posição x ; v_{i+1} é a velocidade de cada partícula, a cada nova iteração.

Outro relevante fator é o wv_i , denominado termo de inércia. Ele é o encarregado por manter a partícula na direção atual, sendo w o coeficiente responsável por definir a força deste elemento.

A expressão $c_1 \cdot r_1 \cdot (p_i - x_i)$ é definida como o elemento cognitivo. Ele regula a prioridade da melhor posição da partícula. O r_1 é uma quantidade randômica entre 0 e 1; c_1 é o coeficiente do elemento cognitivo e p_i representa a melhor posição que a partícula a_i ocupou.

Há, também, o chamado termo social, estipulado por $c_2 \cdot r_2 \cdot (g - x_i)$. Ele exerce uma função oposta ao termo cognitivo. O termo social prioriza o melhor valor de posição encontrado pela população, em que r_2 é um número aleatório entre 0 e 1, c_2 o coeficiente do termo social e g caracteriza a melhor posição global do enxame.

Para cada velocidade atualizada da partícula, o algoritmo aplicará seu efeito na população. Desta forma, a posição de cada partícula será alterada de acordo com sua nova velocidade e trajetória. A posição x de uma partícula em cada iteração i é dada pela Equação (3).

$$x_{i+1} = x_i + v_{i+1} \quad (3)$$

Sendo assim, a posição de cada partícula é ajustada seguindo um movimento não uniforme, baseado na mecânica newtoniana, conforme a Expressão (4).

$$\begin{aligned} \min P_{ap}(P, Q) \\ P, Q \in \mathbb{R}^n \end{aligned} \quad (4)$$

Sujeito a

$$\begin{cases} E_i(P, Q) = 0 \\ I_i(P, Q) \geq 0 \end{cases}$$

Sendo E_i e I_i os vetores que contém o conjunto de restrições que as variáveis P e Q devem satisfazer.

RESULTADOS E DISCUSSÃO

O Processo de Otimização significa determinar o melhor conjunto de resultados, considerando as restrições do fenômeno estudado. Inicialmente identificou-se o propósito primordial da análise, representado pela função objetivo, dada pela Equação (1). A seguir, foram detectas as restrições que as variáveis que modelam a função objetivo estão sujeitas.

Na execução da Otimização por Enxame de Partículas, aplicada nesta pesquisa, foram incorporadas as seguintes restrições operacionais do Sistema Elétrico de 14 barras:

$$-30.88 \leq P \leq 39.41$$

$$-2.07 \leq Q \leq 56.67$$



Para melhor visualização da otimização, a Figura 3 apresenta o comportamento da função objetivo.

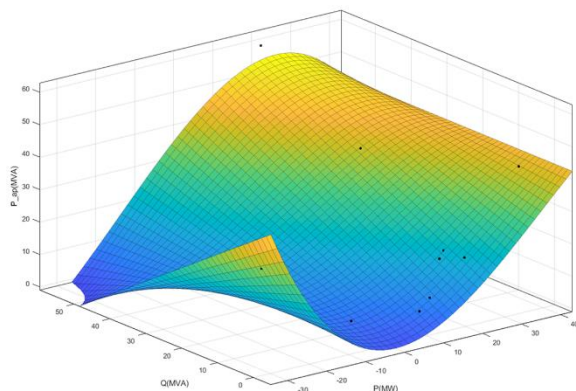


Figura 3. Representação gráfica da função objetivo.

Na execução da OEP foram utilizados os seguintes valores:

$$0.9 \leq w \leq 0.4$$

$$c_1 = 2.00$$

$$c_2 = 2.05$$

$$0.0 \leq r_1 \leq 1.0$$

$$0.0 \leq r_2 \leq 1.0$$

$$N_{\text{máx}} \text{ iterações} = 1000$$

$$N_{\text{partículas}} = 100$$

A Otimização por Enxame de Partículas – OEP (PSO – *Particle Swarm Optimization*), inspira-se no comportamento social das aves e dos peixes. Os indivíduos da população são representados por partículas no espaço de busca, as quais se movimentam com o objetivo de convergir para a posição de uma partícula ótima (Paiva).

As Figuras 4, 5 e 6 expõem o comportamento das partículas durante o processo de otimização.

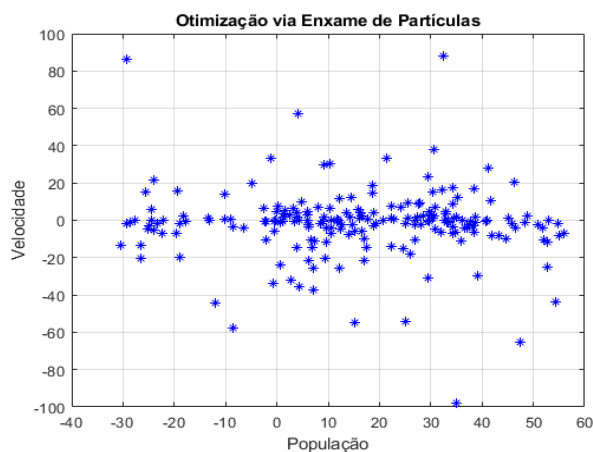


Figura 4. Movimento das partículas – Espaço I.

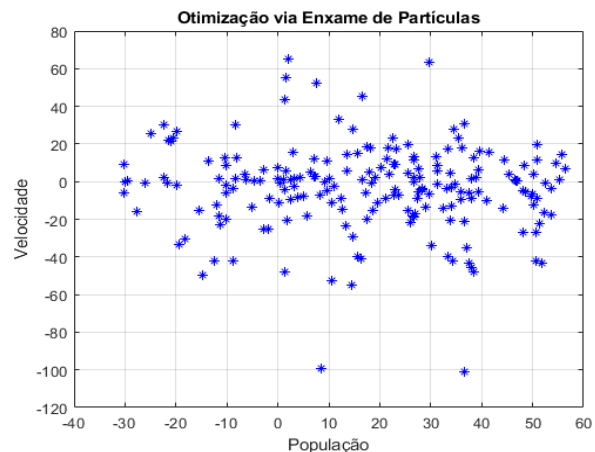


Figura 5. Movimento das partículas – Espaço II.

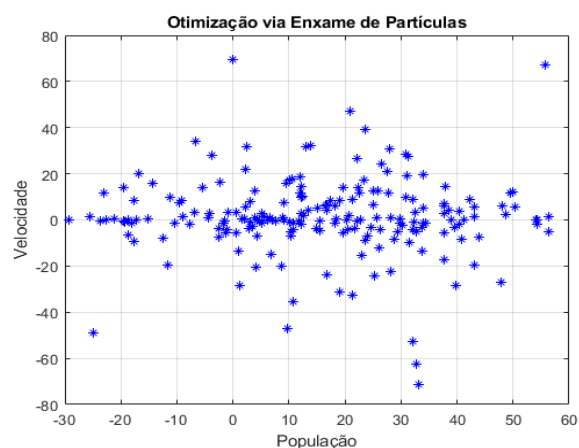


Figura 6. Movimento das partículas – Espaço III.

A Natureza é composta por incontáveis sistemas físicos que, frequentemente, tendem a convergir para um estado de energia mínima. A corrente elétrica, por exemplo, procura o caminho de menor resistência em seu percurso, a fim de minimizar as perdas com a dissipação térmica.

A Figura 7 apresenta a dinâmica da OEP na busca por minimizar a função objetivo (função *fitness*).

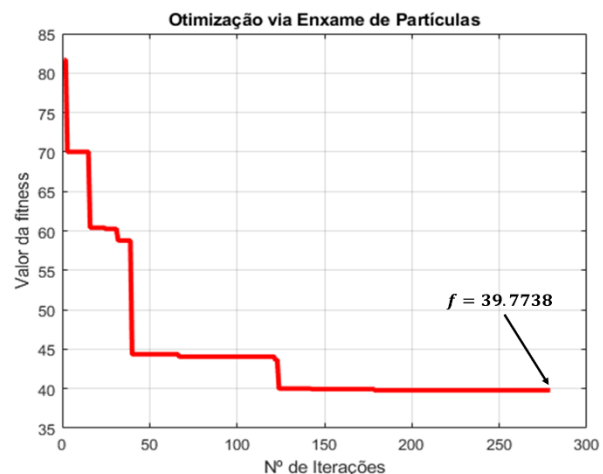


Figura 7. Decaimento da função *fitness*.



A solução ótima, encontrada pela Otimização por Enxame de Partículas, está representada na Tabela 2.

Tabela 2. Valores das Potências Elétricas.

Variáveis	Valor ótimo	Limite inferior	Limite superior
P (MW)	18.9650	-30.88	39.41
Q (MVA)	34.1170	-2.07	56.67
P_{ap} (MVA)	39.7738	1.94	59.99

As variáveis independentes, analisadas no sistema de 14 barras, são a Potência Ativa (P) e a Potência Reativa (Q). Elas compõem a população de partículas do enxame, cujas posições, no espaço de busca, podem ser vislumbradas na Figura 8.

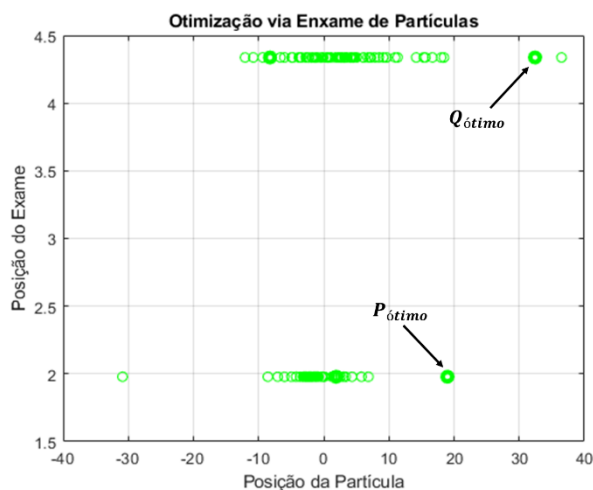


Figura 8. Variáveis no espaço de busca.

O gráfico, apresentado pela Figura 9, mostra o comportamento da função *fitness* (função objetivo), destacando o valor da solução ótima determinada pela OEP.

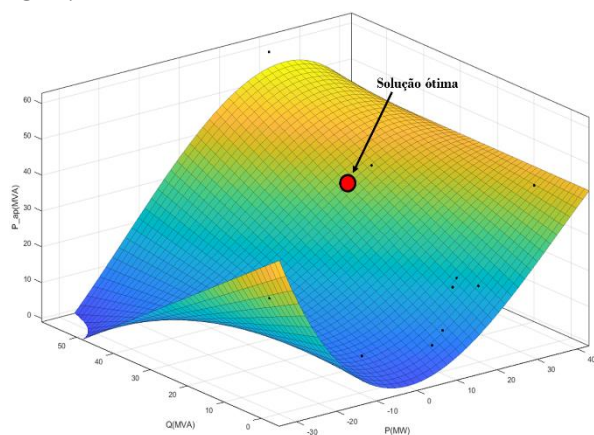


Figura 9. Representação gráfica da solução ótima.

CONCLUSÃO

A busca pela melhor combinação entre eficácia e eficiência faz com que, intuitivamente, apliquemos a otimização em nossos processos. Essa intuição ocorre porque a Natureza está sempre otimizando sua evolução.

Na geração de energia, não poderia ser diferente. Inúmeros métodos de otimização são aplicados para melhorar nossos sistemas energéticos.

Este trabalho propôs a estratégia de Enxame de Partículas, pois trata-se de uma metodologia fundamentada nos aspectos naturais de espécies que vivem em grupo, com uma dinâmica social capaz de modelar o comportamento coletivo a partir das características globais.

Para isso, foram identificados a função objetivo, as variáveis de controle e as restrições operacionais de um sistema elétrico de 14 barras.

A partir dessas informações, aplicou-se a Otimização via Enxame de Partículas, que determinou o melhor conjunto para as variáveis Potência Elétrica Ativa e Potência Elétrica Reativa.

Esse conjunto solução ótima atende os limites operacionais da rede elétrica estudada, garantindo ao operador do sistema elétrico, uma redução de custo e confiabilidade na tomada de decisão.

Espera-se que, para futuros trabalhos, seja possível aplicar a Otimização via Enxame de Partículas em termoeletricas com o objetivo de reduzir a emissão de poluentes, mantendo a segurança no despacho elétrico

AGRADECIMENTOS

Ao Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA *Campus Ananindeua*, pela administração do Bacharelado em Ciência e Tecnologia desse *campus*. E ao grupo de pesquisa Gradiente de Modelagem Matemática e Simulação Computacional – GM²SC, pela gestão e motivação científica de Discentes e Docentes.

REFERÊNCIAS

- COSTA, Denis C. L., NUNES, Marcus V.A., VIEIRA, João P.A. & BEZERRA, Ubiratan H. Decision tree-based security dispatch application in integrated electric power and natural-gas networks. *Electric Power Systems Research* 14. 442–449. 2016.
- COSTA, Heictor A. de O.; COSTA, Denis C. L.; MENESES, Lair A. de. Interdisciplinarity Applied to the Optimized Dispatch of Integrated Electricity and Natural Gas Networks using The tGenetic Algorithm. *Research, society and development, [s.*



L.J., v. 10, n. 2, p. E42110212641, 2021. DOI: 10.33448/rsd-v10i2.12641. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/12641>. 2021.

COSTA, Heictor A. de O.; GOMES, L. L.; COSTA, Denis C. L.; ROCHA, E. M.; FRANCÊS, C. R.; ANDRADE, S. H. Fractional order differential calculus applied on decision making system to smart grid management via decision trees. Research, Society and Development, [S. l.], v. 10, n. 16, p. e38101623387, 2021. DOI: 10.33448/rsd-v10i16.23387. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/23387>. 2021.

COSTA, Heictor A. de O.; GOMES, L. L.; COSTA, Denis C. L. Genetic algorithm and particle swarm applied in electric system optimization. Research, Society and Development, [S. l.], v. 10, n. 10, p. e166101018871, 2021. DOI: 10.33448/rsd-v10i10.18871. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/18871>. 2021.

COSTA, Denis C. L.; MENESES, L. A. de; LIMA, M. L. V. de; COSTA, Heictor A. de O.; REIS, A. C. F.; PINHEIRO, H. F. B.; COSTA, E. F. da; SILVA, A. R. dos S. da; REIS, A. C. F.; RAIOL, F. M.; SANTOS, R. C. P. dos. Thermoelectric generation with reduced pollutants made possible by bio-inspired computing. Research, Society and Development, [S. l.], v. 11, n. 1, p. e7611124568, 2022. DOI: 10.33448/rsd-v11i1.24568. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/24568>. 2022.

KENNEDY, James and EBERHART, Russell. Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, Western Australia, Perth, 1995. <https://www.doi.org/10.1109/ICNN.1995.488968>, 1995.

PAIVA, Fábio A. P. de. Otimização por enxame de partículas – Editora IFRN. Natal: IFRN. Disponível para download em: <http://memoria.ifrn.edu.br>. 2018.

PSTCA. Power Systems Test Case Archive. Available: <http://www.ee.washington.edu/research/ps-tca/>. 1993.

SANCHES, D. S.; MARQUES, R. A. C.; SILVA, M. Integrando relevantes aspectos de Algoritmos Multiobjetivo para treinamento do problema de restabelecimento de Energia em Sistemas de distribuição de grande porte. XIX Congresso Brasileiro de Automática, [S.l.], 2012.