

Solução numérica da equação de Laplace através do método da relaxação

Numerical solution of Laplace's equation through the relaxation method

I. Guilhon^{*1}, Tiago D. Admiral²

¹Instituto Tecnológico de Aeronáutica, Laboratório de Pesquisa em Educação Científica e Tecnológica, 12228-900, São José dos Campos, SP, Brasil.

²Instituto Federal de Educação, Ciência e Tecnologia Fluminense, Campos dos Goytacazes, RJ, Brasil.

Recebido em 09 de setembro de 2022. Revisado em 25 de outubro de 2022. Aceito em 03 de novembro de 2022.

A equação de Laplace é uma equação diferencial que pode descrever diferentes fenômenos físicos. Neste trabalho discutimos como ela pode ser resolvida numericamente através do método da relaxação. O operador laplaciano é escrito em termos de diferenças finitas, estabelecendo uma relação entre o valor da função em um ponto e a média aritmética da função em pontos vizinhos. A relação obtida é aplicada iterativamente e soluções numéricas são obtidas. Aplicamos o algoritmo para a solução da equação de calor considerando estados estacionários e para a determinação de potencial eletrostático entre dois cilindros condutores coaxiais. O acordo entre soluções analíticas e numéricas é muito bom, o algoritmo é computacionalmente barato e pode ser implementado com conhecimentos básicos de algoritmos em poucas linhas de código. Discutimos como essa ferramenta didática pode ser aplicada em diferentes contextos de ensino.

Palavras-chave: Equação de Laplace, Simulações Computacionais, Python, Equações Diferenciais.

Laplace's equation is a differential equation that can describe different physical phenomena. In this work we discuss how it can be numerically solved using the relaxation method. The Laplacian operator is written in terms of finite differences, establishing a relationship between the value of the function at one point and the arithmetic mean of the function at neighboring points. The obtained relationship is applied iteratively and numerical solutions are obtained. We apply the algorithm for the solution of the heat equation considering steady states and for the determination of electrostatic potential between two coaxial conducting cylinders. The agreement between analytical and numerical solutions is very good, the algorithm is computationally cheap and can be implemented with basic knowledge of algorithms in a few lines of code. We discuss how this didactic tool can be applied in different teaching contexts.

Keywords: Laplace's equation, Computational Simulations, Python, Differential Equations.

1. Introdução

A formação de estudantes nos cursos de física, engenharia e áreas correlatas costuma dar maior ênfase em problemas analíticos. Técnicas de solução analíticas de equações diferenciais, sejam ordinárias ou parciais, são amplamente exploradas, enquanto técnicas numéricas muitas vezes ficam restritas a disciplinas específicas que envolvem cálculo numérico ou programação. Dessa forma, técnicas de solução numéricas de equações diferenciais costumam ser vistas de forma menos aprofundada e descontextualizada [1].

Entretanto, diversos sistemas físicos de interesse podem não possuir uma solução analítica simples, tornando-se imprescindível o domínio de técnicas de soluções numéricas para que o estudante esteja preparado para lidar com problemas reais. Diretrizes e bases curriculares têm sido elaboradas tendo em vista o desenvolvimento de competências e habilidades do estudante para atuar

nesse tipo de situação, integrando conhecimentos de diferentes áreas [2, 3]. Um exemplo de grande interesse na física e engenharia é a solução da equação de Laplace, que pode ser aplicada, entre outros contextos, a problemas de equação de calor e determinação de potencial elétrico. Para ambos os casos a maior parte dos livros na literatura [4–6], aborda prioritariamente as soluções analíticas das respectivas equações. O que reforça a necessidade da criação e difusão de materiais didáticos para uma abordagem numérica desses problemas.

Nesse contexto, é de suma importância o desenvolvimento de ferramentas capazes de trazer problemas numéricos à sala de aula de maneira didática, aproveitando ambientes e softwares de programação cada vez mais acessíveis. É fato conhecido que a utilização de simulações e softwares de soluções numéricas para propósitos didáticos já se mostraram ferramentas valiosas no ensino de física [7].

Neste artigo apresentamos uma implementação do método da relaxação para solução numérica da equação de Laplace, utilizando linguagem *Python* [8]. A estratégia

* Endereço de correspondência: guilhon@ita.br

didática escolhida baseia-se na simplicidade da implementação dos códigos, que exigem apenas noções básicas de programação. Os códigos apresentados podem ser facilmente reproduzidos e adaptados para diferentes contextos e condições de contorno. Concentramos nossa discussão em problemas com soluções analíticas conhecidas a fim de comparar o desempenho do método numérico proposto frente às respectivas soluções exatas de cada sistema.

Na seção 2, discutimos métodos de soluções analíticas da equação de Laplace para solução da equação de calor e determinação de potencial elétrico. Na seção 3 apresentam-se noções elementares de cálculo de derivadas por diferenças finitas e a implementação do método de relaxação. Na seção 4, apresentamos como diferentes problemas físicos podem ser investigados com o método bastando alterar as condições de contorno de forma conveniente. A comparação dos resultados analíticos e numéricos é feita sempre que possível. Sugestões de parâmetros de simulação e indicações dos custos computacionais associados são fornecidas ao longo do texto. Finalmente, concluímos o texto trazendo algumas considerações e perspectivas para o uso de simulações como essas em salas de aula.

2. Problemas de Interesse

2.1. Equação de Calor

Entre o fim do século XVIII e início do século XIX, Jean Baptiste Joseph Fourier dedicou-se ao estudo da descrição matemática da condução de calor. Influenciado pelo desenvolvimento do cálculo e suas aplicações, no ano de 1807, apresentou seu primeiro manuscrito voltado ao tema, com título *Mémoire sur la propagation de la chaleur dans les corps solides*. Tendo se debruçado sobre o tema por mais alguns anos, Fourier publicou a versão final [9] do seu trabalho sobre a condução de calor no ano de 1822, sob o título *Théorie analytique de la chaleur*, nesse trabalho ele não apenas apresentou a formulação da equação de calor em sua forma de equação diferencial parcial, como conhecemos hoje, como também descreveu a solução para algumas condições específicas, utilizando a técnica de soma de séries trigonométricas.

A equação de calor, como a conhecemos hoje, pode ser escrita como

$$\frac{\partial \theta}{\partial t} = \eta \nabla^2 \theta, \quad (1)$$

em que $\theta(\vec{r}, t)$ representa o campo de temperatura. Na condição estacionária, $\frac{\partial \theta}{\partial t} = 0$. Segue, portanto, que nesse caso a equação de calor reduz-se à equação de Laplace

$$\nabla^2 \theta = 0. \quad (2)$$

Quando tomada a situação de propagação de calor unidimensional, em regime permanente, teremos que a

solução geral da equação (2) é trivial e se resume a uma função linear, geralmente estudada no ensino médio [10].

O problema bidimensional oferece uma maior riqueza de soluções possíveis. Suponha uma chapa metálica de altura b , comprimento a e de espessura desprezível, localizada sobre o plano XY. Admita que a chapa está sujeita a seguinte condição de contorno de Dirichlet [6]: uma de suas arestas é mantida a uma temperatura T_0 , enquanto as demais são mantidas na temperatura $T = 0$ u.a. A situação é ilustrada na Figura 1.

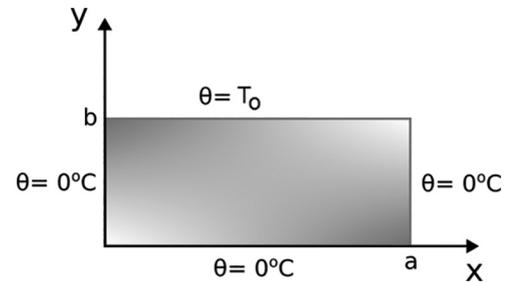


Figura 1: Condições de contorno escolhidas para a determinação do campo de temperatura estacionário de uma chapa bidimensional.

Soluções analíticas do problema são usualmente obtidas pela técnica de separação de variáveis. Assumindo que a solução da equação (2) possa ser expressa pelo produto de funções de uma única variável, isto é, $\theta(x, y) = X(x)Y(y)$, deduz-se que

$$Y \frac{d^2 X}{dx^2} + X \frac{d^2 Y}{dy^2} = 0. \quad (3)$$

Considerando que a solução é não nula, podemos dividir toda equação (3) por XY , que resulta em:

$$\frac{1}{X} \frac{d^2 X}{dx^2} + \frac{1}{Y} \frac{d^2 Y}{dy^2} = 0. \quad (4)$$

Consequentemente, pode-se escrever

$$\frac{1}{X} \frac{d^2 X}{dx^2} = -k^2 \quad e \quad \frac{1}{Y} \frac{d^2 Y}{dy^2} = k^2, \quad (5)$$

em que k é uma constante de dimensões físicas adequadas. As equações diferenciais ordinárias obtidas admitem soluções gerais dadas por $X(x) = A \cos(kx) + B \sin(kx)$ e $Y(y) = C e^{-ky} + D e^{ky}$, respectivamente [6]. Consequentemente, chega-se à seguinte solução da equação diferencial parcial

$$\theta(x, y) = [A \cos(kx) + B \sin(kx)] \cdot [C e^{-ky} + D e^{ky}]. \quad (6)$$

Utilizando as condições de contorno estabelecidas inicialmente, podemos ajustar os valores das constantes para nosso problema, em particular. Utilizando a condição $\theta(0, y) = 0$, teremos obrigatoriamente que $A = 0$.

Ao considerarmos a condição para a qual $\theta(x, 0) = 0$, teremos que as constantes C e D devem ser valores

simétricos. Dessa maneira essas constantes podem ser absorvidas em B , e nossa solução geral reduz-se a

$$\theta(x, y) = B \operatorname{sen}(kx)(e^{ky} - e^{-ky}). \tag{7}$$

O fator $(e^{ky} - e^{-ky})$ pode ser substituído por $2 \operatorname{senh}(ky)$, e o valor 2 por sua vez também pode ser absorvido pela constante B , a ser ajustada conforme a condição de contorno.

A condição que garante que $\theta(a, y) = 0$, logo, para que essa condição seja satisfeita, os valores de k devem ser tais que $k = \frac{n\pi}{a}$, com $n = 1, 2, 3, \dots$. Considerando esse fato nossa solução poderá ser reescrita sob a seguinte forma:

$$\theta(x, y) = B \operatorname{sen}\left(\frac{n\pi x}{a}\right) \operatorname{senh}\left(\frac{n\pi y}{a}\right). \tag{8}$$

Como uma última condição de contorno sabemos que $\theta(x, b) = T_o$, somado-se a esse fato, temos que a solução geral é a combinação linear dos valores de n , o que nos leva à expressão

$$T_o = \sum_{n=1}^{\infty} B_n \operatorname{sen}\left(\frac{n\pi x}{a}\right) \operatorname{senh}\left(\frac{n\pi b}{a}\right). \tag{9}$$

A fim de encontrar os valores dos coeficientes B_n vamos utilizar a propriedade de ortogonalidade de funções, para tanto multiplicaremos a equação (9) pela função $\operatorname{sen}\left(\frac{n'\pi x}{a}\right)$, e integraremos ambos membros no intervalo θ a a . A ortogonalidade entre as diferentes funções seno, permite chegarmos à seguinte expressão para os coeficientes B_n :

$$B_n = \frac{2T_o}{\pi} \frac{(-1)^{n'+1} + 1}{n'}. \tag{10}$$

O que nos leva à expressão

$$\sum_{n=1}^{\infty} \frac{2T_o}{\pi} \frac{(-1)^{n+1} + 1}{n} \operatorname{sen}\left(\frac{n\pi x}{a}\right) = T_o. \tag{11}$$

Comparando as equações (9) e (11), podemos isolar os coeficientes B_n e escrever a solução final da equação:

$$\begin{aligned} \theta(x, y) &= \frac{2T_o}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^{n+1} + 1}{n} \\ &\times \operatorname{sen}\left(\frac{n\pi x}{a}\right) \frac{\operatorname{senh}\left(\frac{n\pi y}{a}\right)}{\operatorname{senh}\left(\frac{n\pi b}{a}\right)}. \end{aligned} \tag{12}$$

A equação (12) é a série que converge para a solução o campo de temperatura $\theta(x, y)$ no regime estacionário, obtido a partir da solução da equação de Laplace.

2.2. Potencial elétrico V

Considerando que o campo elétrico \vec{E} é um campo conservativo, sabemos que podemos associar à ele um potencial escalar V , tal que [5]

$$\nabla V = -\vec{E}. \tag{13}$$

Tomando-se o divergente em ambos os membros da equação (14), teremos no membro da esquerda o Laplaciano do potencial V , ou seja, o divergente do gradiente de V . Já no membro da direita, a divergência do campo \vec{E} pode ser substituída pela sua equivalência obtida na Lei de Gauss na forma diferencial. Tem-se, portanto, que

$$\nabla^2 V = -\frac{\rho(\vec{r})}{\epsilon_0}. \tag{14}$$

A equação (14) é chamada de equação de Poisson, e relaciona o operador Laplaciano do potencial elétrico com a densidade volumétrica de carga. Ao considerarmos uma região fechada no espaço na qual não há distribuição de carga em seu interior ($\rho(\vec{r}) = 0$), a equação de Poisson, reduz-se a

$$\nabla^2 V = 0, \tag{15}$$

ou seja, se reduz a equação de Laplace.

A fim de investigar em detalhes a solução da equação de Laplace em um problema particular na eletrostática, consideremos dois cilindros condutores muito longos, concêntricos, com seus eixos principais coincidindo com o eixo z . O cilindro interno é mantido a um potencial elétrico $V_1 = V_0$ e tem raio R_1 ; o externo é aterrado ($V_2 = 0V$) e tem raio R_2 .

Como não há distribuição de cargas elétricas na região do espaço compreendida entre os cilindros, o potencial elétrico nessa região satisfaz a equação de Laplace. Adotando o sistema coordenadas cilíndricas para aproveitar a simetria, verifica-se que [5]

$$\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial V}{\partial r} \right) = 0. \tag{16}$$

A equação 16 pode ser resolvida por integração direta [6]. Sua solução geral é dada pela expressão

$$V(r) = c_1 \ln(r) + c_2, \tag{17}$$

em que c_1 e c_2 são duas constantes físicas com unidade de potencial elétrico ajustadas para satisfazer as condições de contorno em $r = R_1$ e $r = R_2$. Aplicando as condições de contorno anteriormente descritas, chega-se ao seguinte resultado analítico

$$V(r) = V_0 \frac{\ln(r/R_2)}{\ln(R_1/R_2)}. \tag{18}$$

Percebemos que os valores de potencial começam com V_0 superfície do cilindro interno, que é uma superfície equipotencial e, conforme nos afastamos do cilindro do interior, os valores de potenciais caem, convergindo para zero para $r = R_2$.

3. Solução Numérica de Equações Diferenciais

3.1. Método de diferenças finitas

O método de diferenças finitas permite realizar estimativas dos valores de derivadas de uma função a partir de

amostragem dos seus valores em diferentes pontos [11]. Assumiremos a seguir diferenças finitas dos valores da função assumindo um passo $h > 0$ constante.

Assumindo que uma dada função $f(x)$ é diferenciável num certo intervalo $[x_1, x_2]$, temos que a aproximação da derivada da função $f(x)$ para um valor x_0 tal que $x_1 < x_0 < x_2$, é dada pela equação

$$\frac{df}{dx}(x_0) = \frac{f(x_0 + h) - f(x_0)}{h}. \quad (19)$$

O valor do passo h deve ser definido segundo uma escolha de compromisso entre a precisão da estimativa do valor da derivada no ponto em questão e o número de iterações que serão necessárias.

Procedimentos análogos podem ser aplicados para funções de várias variáveis e derivadas de ordem superior [11]. Imaginemos uma função $f(x, y)$, também diferenciável num certo intervalo $[x_1, x_2]$, de forma análoga suas derivadas parciais podem ser estimadas a partir das equações [12]

$$\frac{\partial^2 f}{\partial^2 x}(x_0, y_0) = \frac{f(x_0 + h, y_0) - 2f(x_0, y_0) + f(x_0 - h, y_0)}{h^2} \quad (20)$$

e

$$\frac{\partial^2 f}{\partial^2 y}(x_0, y_0) = \frac{f(x_0, y_0 + h) - 2f(x_0, y_0) + f(x_0, y_0 - h)}{h^2}. \quad (21)$$

Somando as equações (20) e (21), determinamos uma expressão numérica capaz de determinar numericamente o valor do laplaciano de uma função de duas variáveis descrita em coordenadas cartesianas

$$\nabla^2 f = \frac{f(x + h, y) + f(x - h, y) + f(x, y + h) + f(x, y - h) - 4f(x, y)}{h^2}. \quad (22)$$

No caso da equação de Laplace, $\nabla^2 f = 0$, a equação (22) pode ser reescrita de modo a fornecer o valor de $f(x, y)$ em função dos valores da função nos pontos vizinhos, isto é,

$$f(x, y) = \frac{f(x + h, y) + f(x - h, y) + f(x, y + h) + f(x, y - h)}{4}. \quad (23)$$

Considerando uma rede de pontos de passo fixo, a equação (23) pode ser interpretada como uma expressão discreta da propriedade geral de soluções bidimensionais da equação de Laplace [5]

$$f(x, y) = \frac{1}{2\pi R} \oint_C f dl, \quad (24)$$

em que $f(x, y)$ é integrada ao longo de uma circunferência em torno de (x, y) . O tipo de superfície gerada pela solução da equação de Laplace apresenta como propriedade a não existência de máximos e mínimos locais,

exceptuando-se os extremos de cada intervalo analisado, portanto essas superfícies não apresentam picos ou vales ao longo de seu domínio [5]. Essa propriedade também é válida, *mutatis mutandis*, para os casos unidimensional e tridimensional.

3.2. Método da relaxação

A implementação proposta do método da relaxação a equações diferenciais consiste na aplicação da equação (23) a todos os pontos amostrados no domínio da função $f(x, y)$, seguida da imposição das condições de contorno do problema. Para problemas de condição de contorno fixa de Dirichlet, uma única aplicação das condições de contorno é o suficiente. Para atacar problema com condições de contorno mais gerais, como de Neumann, podemos reaplicar as condições de contorno ao final de cada iteração.

O processo deve ser repetido um número suficientemente grande de vezes até que a função obtida convirja para a solução analítica dentro de uma margem aceitável de precisão. Esse algoritmo de relaxação pode ser implementado de maneira genérica, utilizando função da biblioteca *numpy* [13], como mostrado no código a seguir:

```

1 import numpy as np
2 NInt=300
3 N=101
4 a = np.linspace(-1, 1, N)
5 b = np.linspace(-1, 1, N)
6 xa, ya = np.meshgrid(a, b)
7 V = np.zeros_like(xa)
8
9 condContorno(V)
10 for k in range(NInt):
11     for i in range(1, N-1):
12         for j in range(1, N-1):
13             V[i, j] = 0.25*(V[i-1, j] + V[i+1, j]
14             + V[i, j-1] + V[i, j+1])
15     condContorno(V)

```

Código 1: Exemplo de implementação do algoritmo de relaxação em linguagem *Python*.

em que *NInt* é o número de iterações que serão realizadas em uma rede quadrada de $N \times N$ ($N=101$) pontos na região $-1 < x < 1$ e $-1 < y < 1$. A função $V(x, y)$ é a função de duas variáveis solução da equação de Laplace no domínio considerado. A aplicação da condição de contorno é representada pela função *condContorno* (*V*).

A implementação da condição de contorno varia conforme o problema considerado. Para o problema da determinação do campo de temperaturas de uma placa retangular, *e.g.*, pode-se implementar a função *condContorno* (*V*) como simples atribuição de temperaturas na fronteira da chapa. Caso a geometria do problema não seja retangular, basta aplicar valores correspondentes a condição de contorno na fronteira do domínio de interesse. Às posições que não pertencem ao domínio da função $V(x, y)$, podem atribuídos quaisquer valores.

É evidente que essas implementações não são únicas, tampouco de desempenho computacional otimizado. Procura-se neste trabalho evitar complicações didáticas associadas à reescrita do operação Laplaciano em coordenadas cilíndricas ou a necessidade de um conhecimento de computação além de um laço `for` simples. Os trechos de código foram escritos em *Python*, mas podem ser traduzidos para outras linguagens sem grandes dificuldades. A biblioteca *Numpy* utilizada serve para criação e manipulação de vetores e matrizes.

O número de iterações necessárias para atingir um determinado grau de precisão pode ser significativamente reduzido caso a função os valores iniciais de $V(x, y)$ sejam próximos da solução da equação de Laplace. De maneira geral, algum tipo de interpolação ou média dos valores das condições de contorno são escolhas eficientes para as condições iniciais. O processo de convergência pode ser também acelerado mediante uso de expressões de diferenças finitas que utilizem mais pontos amostrados. Por simplicidade, restringimo-nos à amostragem do valor da função no ponto de interesse e de primeiros vizinhos em cada direção.

O custo computacional da simulação varia principalmente com o número de pontos da rede de interesse, N^2 nos exemplos fornecidos anteriormente. A elevação do parâmetro N pode implicar em um aumento do número de interações N_{int} necessário para atingir uma precisão desejada. Em todos os testes realizados ao longo do desenvolvimento do trabalho, as simulações mantiveram-se sempre da ordem de segundos, o que permite a adaptação de parâmetros e condições de contorno em sala de aula.

A dimensionalidade do problema é um parâmetro importante para a formulação da resolução do problema. Escolhemos focar nossa atenção em problemas bidimensionais, que proporcionam um bom equilíbrio entre complexidade e custo computacional. A resolução de problemas tridimensionais exige adaptações no código. Por outro lado, problemas unidimensionais podem ser resolvidos com os códigos apresentados, mediante aplicação de condições de contorno que dependam apenas de uma coordenada, conforme exemplificado na próxima seção.

4. Resultados e Discussões

Nesta seção aplicamos o método da relaxação a três problemas de física básica distintos. O primeiro consiste no problema da determinação do campo de temperatura de uma barra condutora de calor entre dois reservatórios térmicos. O segundo problema é a determinação do campo de temperatura de uma chapa, que já foi discutido em detalhes. A aplicação do algoritmo proposto nesse exemplo é imediata. Por fim, ilustramos como geometrias não retangulares podem ser tratadas a partir da determinação do potencial elétrico entre dois cilindros concêntricos.

4.1. Campo de temperatura $\theta(x)$ de uma barra unidimensional

Como uma primeira aplicação do método da relaxação, escolhamos o problema da determinação do campo de temperaturas de uma barra sujeita a um gradiente de temperatura apenas em uma direção, i.e., $\theta(x, y = 0) = 0$ e $\theta(x, y = b) = T_0$. Nas outras duas arestas laterais, $y = 0$ e $y = a$, não há fluxo de calor para fora da chapa. Para tratar com sistemas desse tipo, impõe-se a condição de contorno de Neumann

$$\frac{\partial \theta}{\partial y} = 0. \tag{25}$$

A condição é equivalente a deixar os valores de temperaturas livres, copiando-os, em cada iteração, aos valores das linhas vizinhas. Assumindo $T_0 = 100^\circ\text{C}$, a condição de contorno da situação descrita pode ser implementada em linguagem *Python* conforme mostrado no Código 2.

```

1 def condCotorno_barra1D(V):
2     V[0,:] = 100
3     V[N-1,:] = 0
4     V[:,0] = V[:,1]
5     V[:,N-1] = V[:,N-2]
```

Código 2: Exemplo de implementação de condição de contorno para problema de determinação do campo de temperatura de uma barra unidimensional em linguagem *Python*. Um extremo da barra é mantido em 100°C e o outro em 0°C . A rede de pontos é assumida como $N \times N$ e os índices assumem valores entre 0 e $N - 1$.

O campo de temperatura é inicializado em um valor constante e igual 50°C . A escala de comprimento é escolhida como $a = b = 10$ cm. São realizadas 1000 iterações, chegando ao campo de temperatura mostrado pela Figura 2. Um excelente acordo entre o resultado obtido e o comportamento linear previsto pela solução analítica é verificado.

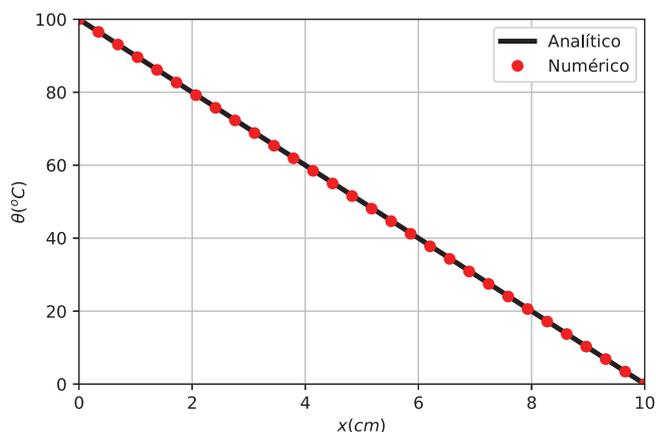


Figura 2: Solução do campo de temperatura para o problema unidimensional de condução de calor. Simulação feita com $N_{int} = 1000$ iterações e rede de 30×30 pontos ($N = 30$).

4.2. Determinação de um campo de temperatura $\theta(x, y)$ de uma chapa

Assumimos agora o problema de determinação do campo de temperatura da uma chapa condutora de calor bidimensional, sujeita às condições de contorno de Dirichlet descrita na seção 2.1. Assumindo $T_0 = 100^\circ\text{C}$, a condição de contorno da situação descrita pode ser implementada em linguagem *Python* conforme mostrado no Código 3. A solução analítica desse problema, conforme visto anteriormente, é expressa pela equação (12).

```

1 def condCotorno_barra2D(V):
2     V[0,:] = 100 #oC
3     V[N-1,:] = 0 #oC
4     V[:,0] = 0 #oC
5     V[:,N-1] = 0 #oC

```

Código 3: Exemplo de implementação de condição de contorno para problema de determinação do campo de temperatura de uma chapa condutora de calor em linguagem *Python*. Uma aresta da barra é mantida em 100°C e as demais em 0°C . A rede de pontos é assumida como $N \times N$ e os índices assumem valores entre 0 e $N - 1$.

Aplicamos o método da relaxação para determinar numericamente o campo de temperatura $\theta(x, y)$ da chapa resolvendo a equação de Laplace. Realizamos $N_{int} = 1000$ iterações e consideramos uma rede de 51×51 pontos. A Figura 3 demonstra o acordo entre as soluções numérica e analítica é demonstrada para dois cortes paralelos as arestas da chapa e passando pelo seu centro. Verifica-se um excelente acordo entre os campos de temperatura analítico e o numericamente obtido.

O comportamento global do campo de temperatura pode ser também investigado com o uso de um mapa de cores, como o fornecido na Figura 4. A cor branca corresponde à temperatura de 100°C , correspondendo à aresta superior; e a cor preta, à temperatura de 0°C ; que ocorre nas demais arestas. O caminho $x = 5$ apresenta uma variação monotônica de temperatura, conforme Figura 3(a), enquanto o caminho $y = 5$ apresenta um máximo de temperatura no ponto equidistante das arestas laterais da chapa, conforme Figura 3(b). Não se percebe diferenças significativas entre os mapas de cores obtidos das soluções analítica e numérica.

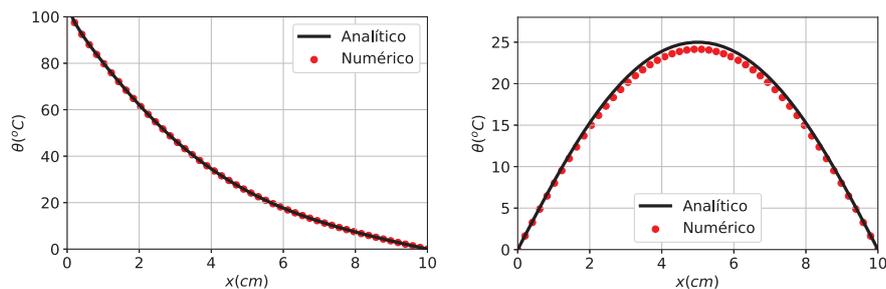


Figura 3: Comparação da solução numérica do campo de temperatura para o problema bidimensional de condução de calor mantendo a aresta $y(10) = 100^\circ\text{C}$ e demais arestas em 0°C . A figura à esquerda representa um corte na linha $x = 5$ e a figura à direita, na linha $y = 5$. Simulação feita com $N_{int} = 1000$ iterações e rede de 51×51 pontos.

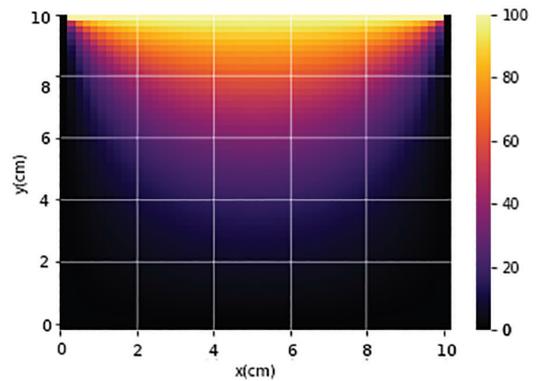


Figura 4: Mapa de cores do campo de temperatura de uma chapa quadrada homogênea com uma aresta mantida em 100°C e as demais mantidas em 0°C .

4.3. Determinação do potencial elétrico entre cilindros coaxiais

Nesta seção discutiremos o problema de determinação do potencial elétrico entre dois condutores cilíndricos concêntricos. O cilindro interior é mantido no potencial elétrico V_0 , enquanto o condutor externo é aterrado. Assumindo $V_0 = 20\text{ V}$, raio do cilindro interno $R_1 = 0,1\text{ m}$ e o do cilindro externo $R_2 = 0,8\text{ m}$, a condição de contorno da situação descrita pode ser implementada em linguagem *Python* conforme mostrado no Código 4. A solução analítica do problema, conforme discutido anteriormente, é expressa pela equação (18).

```

1 def condCotorno_cilindros(V):
2     R1=0.1
3     R2=0.8
4     global xa, ya, N
5     for i in range(N):
6         for j in range(N):
7             x, y = xa[i, j], ya[i, j]
8             if (x**2+y**2 < R1**2):
9                 V[i, j] = 20
10            if (x**2+y**2 > R2**2):
11                V[i, j] = 0

```

Código 4: Exemplo de implementação de condição de contorno para problema de determinação do potencial elétrico entre dois cilindros condutores concêntricos em linguagem *Python*. O cilindro interno tem raio de $0,1\text{ m}$ e é mantido em 20 V ; o segundo tem raio de $0,8\text{ m}$ e é aterrado.

Uma vez que estamos interessados na região compreendida entre os dois condutores, impomos o potencial elétrico $V = V_0$ em toda região interior ao cilindro de menor raio e potencial elétrico $V = 0$ em toda a região externa aos cilindros. A estratégia de solução descrita dispensa a reformulação do código para considerar coordenadas cilíndricas explicitamente.

Aplicamos o método da relaxação para determinar numericamente o potencial elétrico $V(r)$ do sistema descrito. Realizamos $N_{int} = 1000$ iterações e consideramos uma rede de 101×101 pontos. A simulação desse sistema em um computador pessoal de processador Intel i7 de sétima geração levou 23,6 s. O aumento do custo computacional comparado com as demais simulações advém do aumento do número de pontos da rede, a fim de garantir uma amostragem mais fina em torno do cilindro interno, e do fato da aplicação da condição de contorno nesse caso considerar toda a rede de pontos e não a sua fronteira.

O acordo entre as soluções numérica e analítica é demonstrada para um corte radial no sistema. Os resultados dos potenciais obtidos pela expressão analítica e método numérico são mostrados na Figura 5. O acordo entre os dois resultados é bastante satisfatório.

O comportamento da solução na região de interesse pode ser investigada com o mapa de cores ou com um gráfico tridimensional, conforme painéis disponíveis na Figura 6 à esquerda e à direita, respectivamente. Na

região próxima ao cilindro central o potencial é fixado em $V_0 = 20V$, cai de forma abrupta na vizinhança do cilindro interior, diminuindo sua taxa de decrescimento nas vizinhanças do cilindro externo aterrado ($V = 0$). As regiões de maior gradiente de campo elétrico correspondem àquelas em que o campo é mais intenso.

O método numérico tem a vantagem de ser facilmente adaptado para sistema com baixa simetria, como cilindros não coaxiais ou armaduras de diferentes formatos. Quando a solução analítica não é conhecida, é desejável que um parâmetro de precisão seja considerado como critério de parada para o numérico de interações e que N_{int} seja mantido apenas como um número limite de interações. O número de pontos da rede, $N \times N$, deve ser também ajustado a fim de se garantir uma boa amostragem de regiões com forte variação da função desejada.

5. Conclusões

Apresentamos o método da relaxação para resolução da equação de Laplace. A título de exemplo, resolvemos o problema da determinação de campos de temperaturas considerando o fluxo de fluxo estacionário através de uma barra unidimensional e através de chapa bidimensional sujeita a condições de contorno de Dirichlet, assim como a determinação do potencial elétrico entre dois cilindros coaxiais. Os problemas foram escolhidos a fim de demonstrar o baixo custo computacional do método assim como sua possibilidade de aplicação em diferentes áreas da física. A facilidade de adaptação da ferramenta proposta para problemas distintos, bem como liberdade de aplicação de condições de contorno de diferentes naturezas, permite a exploração de diferentes situações físicas em contexto didático.

Os dados obtidos pelo método numérico foram comparados com a solução analítica sempre que possível, mostrando um ótimo acordo. O comportamento das soluções foram investigados com diferentes ferramentas de visualização e diferentes seções do domínio. A eficiência numérica das simulações ou o grau de refinamento das redes de pontos podem ser retrabalhados para atingir melhores acordos, caso isso seja considerado conveniente.

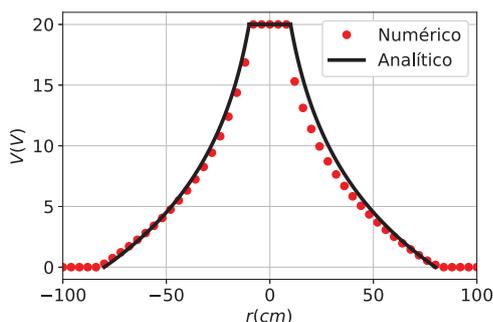


Figura 5: Solução para dois cilindros coaxiais. Simulação feita com $N_{int} = 1000$ iterações e rede de 101×101 pontos.

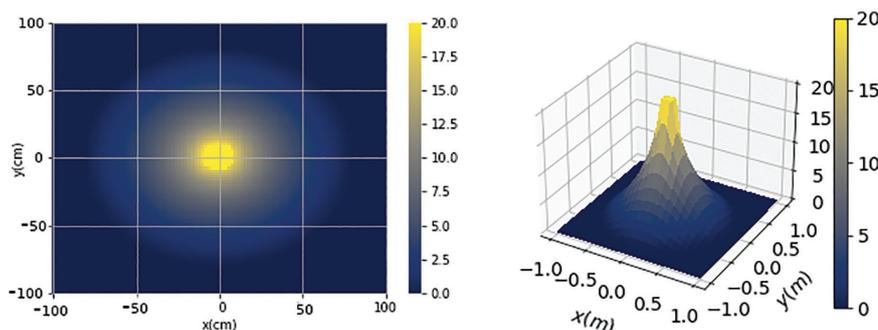


Figura 6: Mapa de cores, à esquerda, e gráfico tridimensional do potencial do sistema de cilindros coaxiais. Valores todos fornecidos em volts. Simulação feita depois de 1000 iterações e rede de 101×101 pontos.

Ao longo do texto indicamos certos caminhos para tal, mas cremos que, como ferramenta didática, eles fogem do escopo desse trabalho.

O bom desempenho demonstrado pelas simulações e o baixo custo computacional do método, que pode ser executado em computadores pessoais convencionais, demonstram ser o método da relaxação uma ferramenta didática viável em diferentes contextos. O grau de detalhamento e discussão dos códigos pode ser ajustado de acordo com o nível de conhecimento dos estudantes. Dessa forma, apresentamos um caminho didático para ir além de uma abordagem estritamente analítica para a investigação de diferentes problemas em sala de aula. Exploramos em maior detalhes problemas clássicos, com solução analítica bastante conhecida, mas a aplicação do método numérico para problemas de baixa simetria não apresenta grande dificuldade.

Agradecimentos

Os autores agradecem à ITAEx – Ex-Alunos Apoiando o ITA – pelo apoio e financiamento do projeto 221043: Metodologias ativas e experimentos em sala de aula de FIS-32, e ao Laboratório de Pesquisa em Educação Científica e Tecnológica (LPECT) do ITA.

Referências

- [1] F. Caruso e V. Oguri, *Rev. Bras. Ensino Fís.* **36**, 2310 (2014).
- [2] <http://basenacionalcomum.mec.gov.br/>
- [3] MINISTÉRIO DA EDUCAÇÃO, *Parecer CNE/CP 9/2001*, 8 de maio de 2001. Distrito Federal, 2001. Disponível em: <http://portal.mec.gov.br/cne/arquivos/pdf/009.pdf>
- [4] S.J. Farlow, *Partial differential equations for scientists and engineers* (Courier Corporation, Massachusetts, 1993).
- [5] D.J. Griffiths, *Introduction to electrodynamics* (Cambridge University Press, Cambridge, 2017).
- [6] W.E. Boyce, R.C. DiPrima e D.B. Meade, *Elementary Differential Equations and Boundary Value Problems* (John Wiley & Sons, Inc., Nova Jersey, 2001).
- [7] A. Medeiros e C.F. Medeiros, *Rev. Bras. Ensino Fís.* **24**, 77 (2002).
- [8] G. Van Rossum e F.L. Drake, *Python tutorial* (Centrum voor Wiskunde en Informatica Amsterdam, Amsterdam, 1995).
- [9] A. Pifer e K.M. Aurani, *Rev. Bras. Ensino Fís.* **37**, 1603 (2015).
- [10] H.S. Amorim, M.A. Dias e V. Soares, *Rev. Bras. Ensino Fís.* **37**, 4310 (2015).
- [11] J.C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations* (Society for Industrial and Applied Mathematics, Pensilvânia, 2004), 2 ed.
- [12] B. Fornberg, *Mathematics of computation* **51**, 699 (1988).
- [13] C.R. Harris, K.J. Millman, S.J. Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith et al., *Nature* **585**, 357 (2020).