

UNIVERSIDADE DE SÃO PAULO  
Instituto de Física de São Carlos

## **Potenciais e campos perto de cargas elétricas**

Aluno: Matheus Fernandes Sousa Lemes (Nº USP: 9866506)  
Curso: Bacharelado em Física  
Disciplina: Eletromagnetismo computacional  
Orientador: Prof. Dr. Guilherme Matos Sipahi

São Carlos - 2020

# Sumário

<b>1</b>	<b>Introdução</b>	<b>2</b>
1.1	A equação de Poisson . . . . .	2
1.2	Método das diferenças finitas . . . . .	2
1.3	Algoritmo de relaxação de Jacobi . . . . .	3
<b>2</b>	<b>Potencial de uma carga pontual</b>	<b>4</b>
2.1	Esquemático do problema . . . . .	4
2.2	Solução numérica . . . . .	5
2.3	Variando a localização da carga pontual . . . . .	6
<b>3</b>	<b>Carga pontual em coordenadas esféricas</b>	<b>7</b>
3.1	Discretização em coordenadas esféricas . . . . .	7
3.2	Solução numérica . . . . .	8
<b>4</b>	<b>Performance do algoritmo SOR para uma carga pontual em duas e três dimensões</b>	<b>8</b>
4.1	SOR em duas dimensões . . . . .	9
4.2	SOR em três dimensões . . . . .	10
<b>5</b>	<b>Conclusão</b>	<b>11</b>

# 1 Introdução

## 1.1 A equação de Poisson

Esse trabalho foi focado em solucionar os exercícios referentes a seção 5.2 do livro do Giordano [1], em que queremos determinar o potencial e campo elétrico para regiões do espaço que contenham uma densidade de carga volumétrica. A lei de Gauss em sua forma diferencial é dada por

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}. \quad (1)$$

Usualmente é mais fácil trabalharmos com o potencial elétrico, visto que ele é um escalar. Como não há campos magnéticos variando no tempo no nosso problema de interesse, temos que  $\nabla \times \mathbf{E} = 0$ . Isso nos permite dizer que o campo elétrico pode ser escrito como o gradiente de uma função escalar, i.e.,  $\mathbf{E} = -\nabla V$ . Logo, a lei de Gauss se torna

$$\nabla^2 V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} = -\frac{\rho}{\epsilon_0}, \quad (2)$$

que é a conhecida *equação de Poisson*.

Dada as condições de contorno do problema, podemos determinar o potencial elétrico. Por fim, podemos utilizar  $V(x, y, z)$  para calcular o campo elétrico, visto que

$$E_\alpha = -\frac{\partial V_\alpha}{\partial \alpha} \quad (3)$$

com  $\alpha = x, y, z$ .

## 1.2 Método das diferenças finitas

Para resolver a equação de Poisson numericamente é necessário que discretizemos o nosso espaço, em que os pontos serão especificados pelos números inteiros  $i, j, k$ , com  $x = i\Delta x$ ,  $y = j\Delta y$  e  $z = k\Delta z$ . Logo, podemos escrever o potencial como  $V(x, y, z) \equiv V(i, j, k)$ .

Um dos métodos mais famosos para resolver equações diferenciais computacionalmente é o *método das diferenças finitas*, em que aproxima-se as derivadas parciais por diferenças finitas (para mais informações veja [2]). Por exemplo, a derivada parcial de segunda ordem em  $x$  será aproximadamente:

$$\frac{\partial^2 V}{\partial x^2} \approx \frac{V(i+1, j, k) - 2V(i, j, k) + V(i-1, j, k)}{(\Delta x)^2}$$

com as as coordenadas  $y$  e  $z$  tendo expressões similares. Além disso, simplificaremos a notação, em que vamos considerar que  $\Delta x = \Delta y = \Delta z = h$  e que  $\epsilon_0 = 1$ . Logo, podemos reescrever a equação

(2) como

$$V(i, j, z) = \frac{1}{6}[V(i+1, j, k) + V(i-1, j, k) + V(i, j+1, k) + V(i, j-1, k) + V(i, j, k+1) + V(i, j, k-1) + \rho(i, j, k)h^2]. \quad (4)$$

Além disso, o campo elétrico calculado por diferenças finitas será dado por

$$E_x(i, j, k) \approx -\frac{V(i+1, j, k) - V(i-1, j, k)}{2h}, \quad (5)$$

com aproximações análogas para  $y$  e  $z$ .

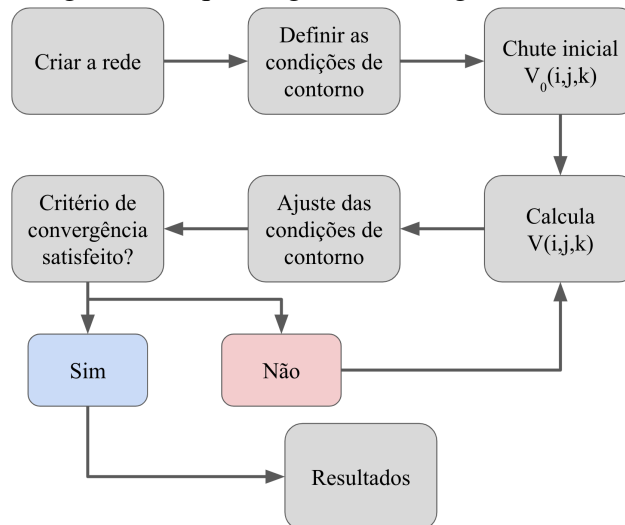
### 1.3 Algoritmo de relaxação de Jacobi

Para implementar o método de diferenças finitas descrito anteriormente é necessário a utilização de processos iterativos. Nós começamos com um chute inicial  $V_0(i, j, k)$  e utilizamos a equação (4) para calcular um valor melhor do potencial  $V_1(i, j, k)$ . A partir de  $V_1(i, j, k)$  calculamos um valor ainda melhor  $V_2(i, j, k)$ . Continuamos esse processo até que nossa solução atinga um critério de convergência. Esse é o conhecido *algoritmo de relaxação de Jacobi* e definimos como critério de convergência a seguinte expressão

$$\frac{\sqrt{\sum_{i=0, j=0, k=0}^n |V_{n+1}(i, j, k) - V_n(i, j, k)|^2}}{\sqrt{\sum_{i=0, j=0, k=0}^n |V_n(i, j, k)|^2}} < \epsilon, \quad (6)$$

em que  $\epsilon$  representa o limite de convergência estabelecido. Um esquema geral desse algoritmo pode ser visto na figura 1.

Figura 1: Esquema geral do código utilizado.



Escolhemos implementar os algoritmos em Python, visto que é uma linguagem simples de trabalhar e muito utilizada no meio científico, além de possuir uma extensa comunidade de usuários que podem ajudar na solução de problemas nos códigos.

## 2 Potencial de uma carga pontual

### 2.1 Esquemático do problema

Vamos resolver a equação de Poisson para o caso de uma carga pontual em três dimensões. De forma idealizada, a densidade de carga de uma carga pontual é representada por uma delta de Dirac

$$\rho(\mathbf{r}) = q\delta(\mathbf{r} - \mathbf{r}_0), \quad (7)$$

em que  $\mathbf{r} = x\hat{\mathbf{i}} + y\hat{\mathbf{j}} + z\hat{\mathbf{k}}$  e  $\mathbf{r}_0$  é a posição da carga. Essa densidade de carga substituída na equação de Poisson apresenta como solução exata o potencial de Coulomb

$$V(r) = \frac{q}{4\pi\epsilon_0 r}, \quad (8)$$

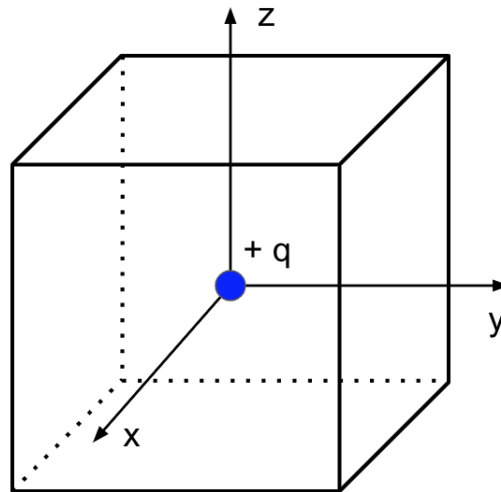
em que  $r = \sqrt{x^2 + y^2 + z^2}$ .

Em métodos numéricos de resolução da equação de Poisson, a expressão (7) torna-se inviável de ser utilizada. Além disso, precisamos de condições de contorno para resolver o problema em questão. A solução para esses dois problemas é apresentada a seguir.

Primeiro, diremos que a densidade de carga será zero em todos os pontos exceto no ponto  $(0, 0, 0)$ , em que teremos  $\rho(0, 0, 0) = q/h^3$ , com  $h$  sendo o incremento espacial introduzido na seção 1.2. Ademais, a carga pontual é colocada no centro de uma caixa cúbica metálica aterrada, i.e.,  $V = 0$ , centrada em  $(0, 0, 0)$ , a qual nos dará condições de contorno relativamente fáceis de trabalhar. A figura 2 apresenta o esquemático do problema.

É importante notar que as condições de contorno escolhidas irão distorcer as linhas equipotenciais esperadas. Isso ocorre porque a caixa metálica irá impor uma simetria cúbica para um problema que é, na verdade, esfericamente simétrico. Mesmo quando fazemos a caixa suficientemente grande, ainda iremos utilizar um incremento espacial cúbico, distorcendo o potencial esperado. Exploramos esse fato na seção 2.2.

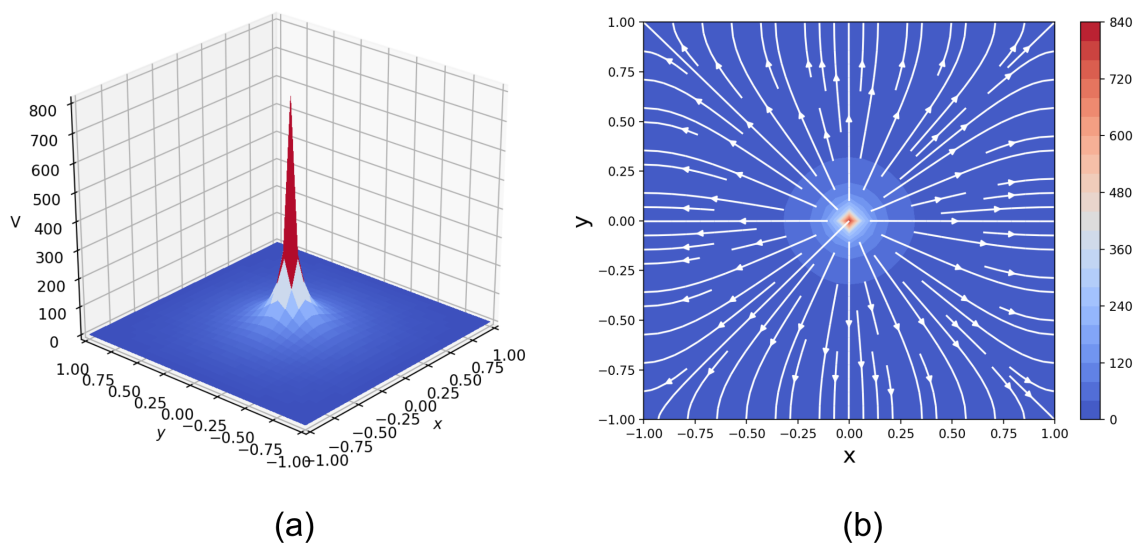
Figura 2: Esquemático de uma carga pontual  $+q$  localizada na origem dos eixos cartesianos. A carga está cercada por uma caixa cúbica metálica, em que as faces do cubo estão com  $V = 0$ .



## 2.2 Solução numérica

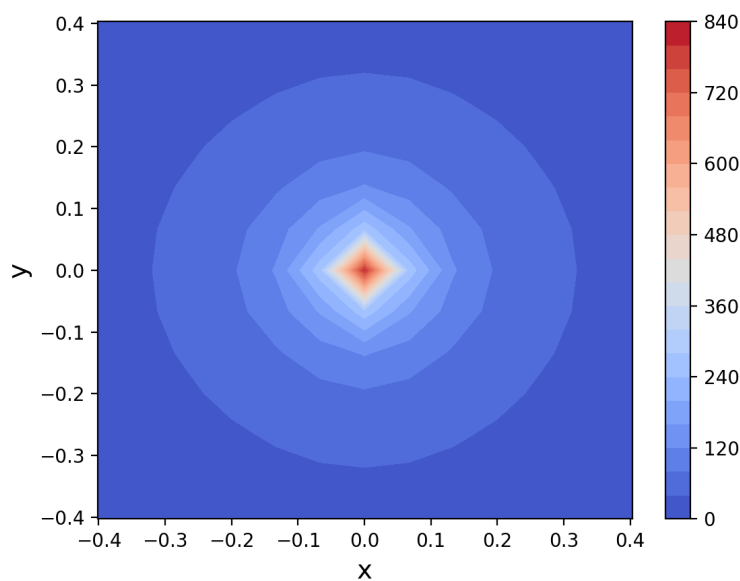
Nesta seção apresentamos os resultados encontrados o problema da carga pontual localizada no centro de uma caixa metálica aterrada. Para uma visualização mais fácil, fizemos o plot do potencial no plano  $z = 0$ . A figura 3 apresenta o plot 3D do potencial  $V(x, y, z = 0)$  assim como as linhas equipotenciais e de campo elétrico.

Figura 3: Resultados para uma carga pontual. (a) Plot 3D do potencial para o plano  $z=0$ . (b) Linhas equipotenciais e de campo elétrico.



Nota-se que os resultados refletem em parte o que era esperado de uma carga pontual: um potencial muito grande na origem e com simetria nas três direções. Contudo, é importante destacar que as linhas equipotenciais estão bem distorcidas perto do centro, apresentando um caráter quadrado e não circular, enquanto que as linhas mais distantes apresentam o caráter esperado. Isso fica mais evidente quando nos aproximamos da origem, como pode ser visto na figura 4. Esse fato simplesmente nos mostra que é necessário termos cuidado com nossas aproximações numéricas, sendo importante sempre analisar o regime de validade delas.

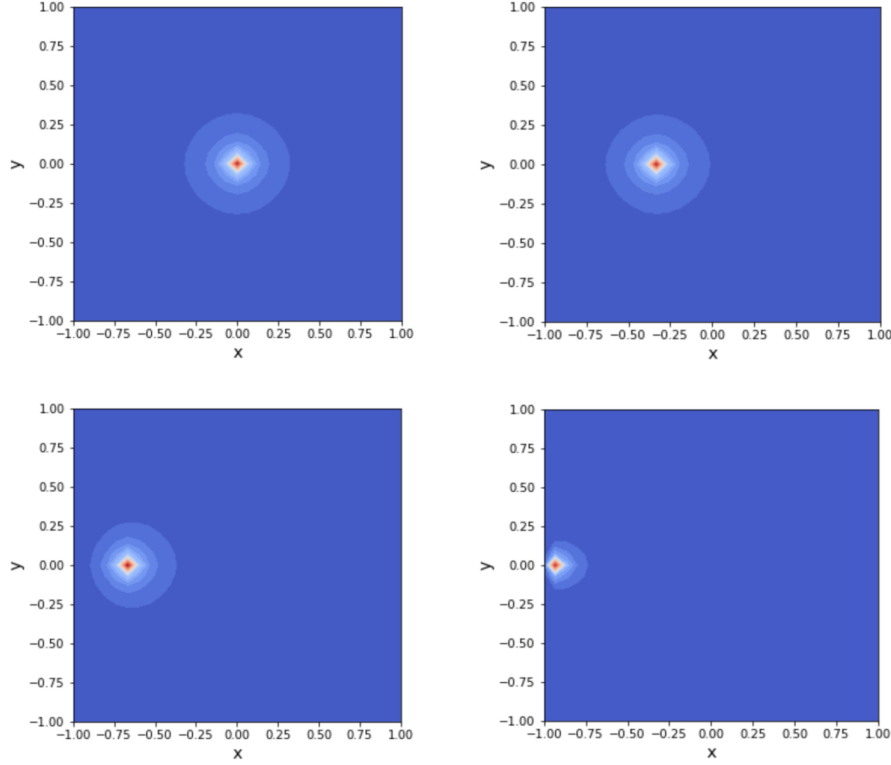
Figura 4: Distorção das linhas equipotenciais perto da origem.



### 2.3 Variando a localização da carga pontual

Para finalizar a seção 2, analisamos como as linhas equipotenciais se comportam quando a carga pontual está próxima de uma das faces do cubo. Nesse caso não esperamos que elas apresentem uma simetria esférica, visto que foi quebrada a isotropia espacial. Na figura 5 vemos como as linhas equipotenciais mudam à medida que aproximamos a carga da face  $x = -1$ . Perceba que a "amplitude" das equipotenciais diminui, ou seja, elas vão se tornando cada vez menores. Outro fato interessante é que a distorção no potencial ao redor da carga, encontrada na seção anterior, se mantém mesmo quando a carga se aproxima das faces aterradas. Isso comprova mais uma vez que essa distorção está associada, muito provavelmente, ao fato de que a rede discretizada possui um incremento espacial cúbico.

Figura 5: Variação das linhas equipotenciais à medida que a posição da carga pontual varia em direção a face  $x = -1$ .



### 3 Carga pontual em coordenadas esféricas

Nessa seção trataremos o caso de uma carga pontual com a geometria característica do problema. Em coordenadas esféricas, queremos encontrar  $V(r, \theta, \phi)$  que satisfaz a equação de Poisson. Como o sistema é invariante nas direções  $\theta$  e  $\phi$ , temos  $V(r, \theta, \phi) = V(r)$ . Logo, a equação de Poisson terá a forma

$$\nabla^2 V = \frac{\partial^2 V}{\partial r^2} + \frac{2}{r} \frac{\partial V}{\partial r} = -\frac{\rho}{\epsilon_0}. \quad (9)$$

Perceba que essa equação é bem diferente do caso cartesiano e uma nova discretização é necessária.

#### 3.1 Discretização em coordenadas esféricas

Para realizar a discretização da equação (9), discretizamos o raio como  $r = r_{min} + ih$ , com  $h$  sendo o incremento espacial. Aplicando a aproximação de diferenças finitas para as derivadas parciais da equação acima, temos que a equação de Poisson discretizada em coordenadas esféricas será dada por:

$$V(i) = \frac{1}{2} \left( V(i+1) + V(i-1) + \frac{h}{r(i)} (V(i+1) - V(i-1)) + \rho(i)h^2 \right) \quad (10)$$

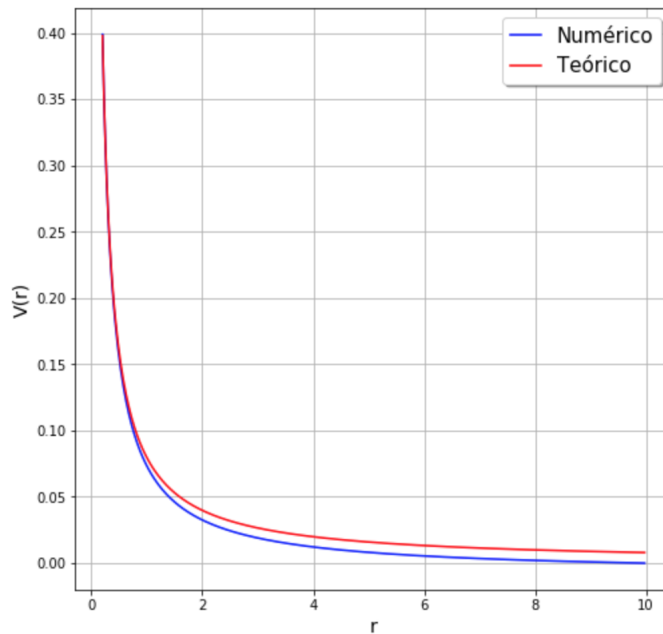


De acordo com o livro, boas escolhas para os parâmetros raio mínimo e incremento espacial são:  $r_{min} = 0.2$  e  $h = 0.025$ . Contudo, o livro sugere que usemos  $V = 0$  para  $r = 5$  como nossa condição de contorno. Percebemos que seria mais vantajoso utilizar  $V = 0$  para  $r = 10$ , visto que se aproxima um pouco mais do caso realista, já que o potencial só iria a zero realmente no infinito.

### 3.2 Solução numérica

Aqui apresentamos os resultados encontrados para o problema da carga pontual em coordenadas esféricas e comparamos com o resultado analítico da lei de Coulomb. Eles podem ser visto na figura 6.

Figura 6: Comparação entre o potencial calculado numericamente e pela lei de Coulomb.



É importante notar que para  $r$  pequeno as curvas se sobrepõem, como era esperado. Porém, conforme  $r$  cresce, o efeito da condição de contorno começa a se tornar importante e, por isso, as curvas acabam divergindo.

## 4 Performance do algoritmo SOR para uma carga pontual em duas e três dimensões

Para finalizar esse trabalho iremos analisar como a performance do algoritmo SOR varia de acordo com a variação do parâmetro de sobre-relaxação. Trataremos o caso de uma carga pontual em duas e três dimensões. Antes de partirmos para os resultados é importante lembrar como funciona o algoritmo.

Em suma, introduzimos um fator  $\alpha$  no algoritmo de Jacobi para acelerar a convergência. Mais precisamente, calculamos o novo potencial como

$$V_{novo} = \alpha \Delta V + V_{velho}, \quad (11)$$

em que  $\alpha$  é o parâmetro de sobre-relaxação,

$$\Delta V = V^* - V_{velho} \quad (12)$$

e  $V^*$  é o potencial calculado pelo método de Jacobi.

De acordo com [3], o valor ótimo de  $\alpha$  é dado por

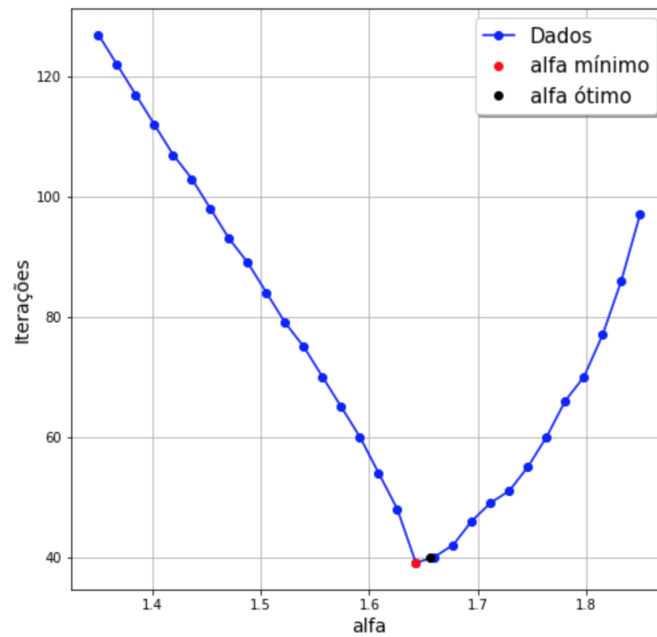
$$\alpha_{opt} = \frac{2}{1 + \sin\left(\frac{\pi}{L}\right)}, \quad (13)$$

em que  $L$  é o número de pontos da rede. Perceba que esse resultado é independente da dimensão do problema, sendo necessário apenas a quantidade de pontos na discretização. Testaremos agora esse resultado analítico. Nessa seção escolhemos  $L = 15$ , de forma que esperaríamos que  $\alpha_{opt} \approx 1.6557$ .

## 4.1 SOR em duas dimensões

Para testar a previsão analítica do valor ótimo para o parâmetro de sobre-relaxação, começamos com o caso bidimensional. Analisamos ele para uma quantidade de pontos de  $L = 15$  e uma tolerância de  $10^{-7}$ . A figura 7 apresenta o número de iterações para diferentes valores de  $\alpha$ .

Figura 7: Comparação de diferentes valores de  $\alpha$  no número de iterações.

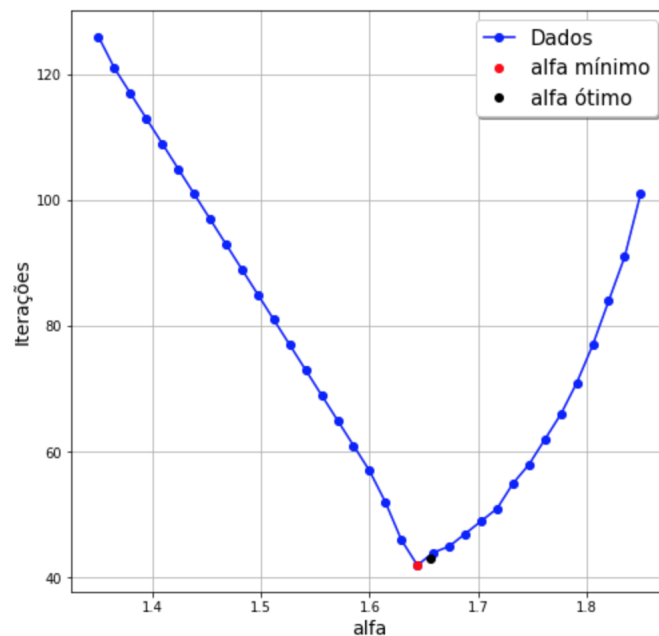


A partir da figura é possível perceber que o valor de  $\alpha$  que produz o menor número de iterações está muito próximo daquele previsto pela equação (13). De fato, o  $\alpha$  mínimo encontrado pelo gráfico foi de 1.6431, apresentando um erro de apenas 0.8%. A pequena diferença pode ser atribuída a diversos fatores como a linguagem que estamos utilizando, o próprio hardware que está realizando as operações, a forma de contagem das iterações, etc. O importante desse resultado é que ele apresenta um mínimo de  $\alpha$  que reside muito perto do valor esperado analiticamente.

## 4.2 SOR em três dimensões

Vamos provar agora que o resultado analítico para  $\alpha$  ótimo realmente não depende da dimensão do problema. Novamente tomamos  $L = 15$  e a tolerância de  $10^{-7}$ . A figura 8 apresenta os resultados.

Figura 8: Comparação de diferentes valores de  $\alpha$  no número de iterações.



Perceba que o gráfico possui um formato muito parecido com o caso 2D. Mais importante ainda, ele possui um  $\alpha$  mínimo de 1.6441, apresentando uma diferença com o valor teórico de 0.7%. Dessa forma, conseguimos provar numericamente que o resultado analítico para  $\alpha$  ótimo, derivado em [3], de fato se concretiza computacionalmente.

## 5 Conclusão

Nesse trabalho foi possível perceber como a escolha da geometria adequada para as condições de contorno possui um papel importante na resolução numérica da equação de Poisson. Além disso, trabalhamos com a equação de Poisson em coordenadas esféricas, a qual possui uma discretização mais complicada e que foi muito útil para nossa bagagem na física computacional. Por fim, analisamos a performance do algoritmo SOR para diferentes valores do parâmetro de sobre-relaxação em duas e três dimensões. Observamos que a melhor escolha para esse parâmetro é para valores muito perto de  $\alpha = 2/(1 + \sin(\pi/L))$ , em que  $L$  é o número de pontos da rede na discretização, concordando com o valor teórico encontrado analiticamente.

## Referências

- [1] N.J. Giordano. *Computational Physics: 2nd edition*. Dorling Kindersley, 2012.
- [2] Wikipedia. *Finite difference method*. [https://en.wikipedia.org/wiki/Finite\\_difference\\_method](https://en.wikipedia.org/wiki/Finite_difference_method).
- [3] Shiming Yang and Matthias K Gobbert. The optimal relaxation parameter for the sor method applied to the poisson equation in any space dimensions. *Applied Mathematics Letters*, 22(3):325–331, 2009.
- [4] Douglas N Arnold. Stability, consistency, and convergence of numerical discretizations. *Encyclopedia of Applied and Computational Mathematics*, pages 1358–1364, 2015.
- [5] Shiming Yang and K Gobbert Matthias. The optimal relaxation parameter for the sor method applied to a classical model problem. Technical report, Technical Report TR2007–6, University of Maryland, Baltimore County, 2007.