

**MAP 2320 – MÉTODOS NUMÉRICOS EM EQUAÇÕES
DIFERENCIAIS II**

2º Semestre - 2019

Prof. Dr. Luis Carlos de Castro Santos

lsantos@ime.usp.br

3. Approximate the solutions to the following elliptic partial differential equations, using Algorithm 12.1:

a. $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0, \quad 0 < x < 1, \quad 0 < y < 1;$

$$u(x, 0) = 0, \quad u(x, 1) = x, \quad 0 \leq x \leq 1;$$

$$u(0, y) = 0, \quad u(1, y) = y, \quad 0 \leq y \leq 1.$$

Use $h = k = 0.2$, and compare the results to the actual solution $u(x, y) = xy$.

Vamos fazer passo a passo com $h = k = 1/3$

Centered finite difference formulas

First Derivative

Error

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1}))}{2h}$$

$$O(h^2)$$

$$f'(x_i) = \frac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2}))}{12h}$$

$$O(h^4)$$

Second Derivative

$$f''(x_i) = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{h^2}$$

$$O(h^2)$$

$$f''(x_i) = \frac{-f(x_{i+2}) + 16f(x_{i+1}) - 30f(x_i) + 16f(x_{i-1}) - f(x_{i-2}))}{12h^2}$$

$$O(h^4)$$

Third Derivative

$$f'''(x_i) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_{i-1}) - f(x_{i-2}))}{2h^3}$$

$$O(h^2)$$

$$f'''(x_i) = \frac{-f(x_{i+3}) + 8f(x_{i+2}) - 13f(x_{i+1}) + 13f(x_{i-1}) - 8f(x_{i-2}) + f(x_{i-3}))}{8h^3}$$

$$O(h^4)$$

Fourth Derivative

$$f^{(4)}(x_i) = \frac{f(x_{i+2}) - 4f(x_{i+1}) + 6f(x_i) - 4f(x_{i-1}) + f(x_{i-2}))}{h^4}$$

$$O(h^2)$$

$$f^{(4)}(x_i) = \frac{-f(x_{i+3}) + 12f(x_{i+2}) + 39f(x_{i+1}) + 56f(x_i) - 39f(x_{i-1}) + 12f(x_{i-2}) + f(x_{i-3}))}{6h^4}$$

$$O(h^4)$$

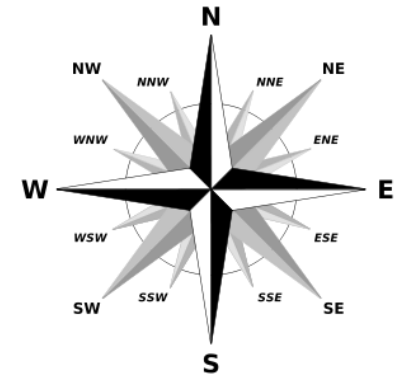


	iterações								
n	0	1	2	3	4	5		sol	erro_2
u_1_1	0	0	0,041667	0,09375	0,106771	0,110026		0,111111	1,17738E-06
u_2_1	0	0,083333	0,1875	0,213542	0,220052	0,22168		0,222222	2,94344E-07
u_1_2	0	0,083333	0,1875	0,213542	0,220052	0,22168		0,222222	2,94344E-07
u_2_2	0	0,375	0,427083	0,440104	0,443359	0,444173		0,444444	7,3586E-08
								norma_2	1,83965E-06

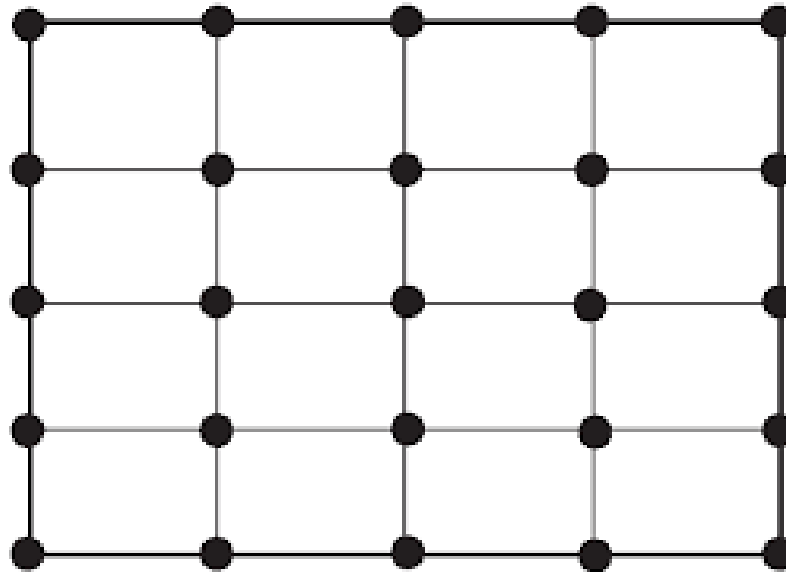
$$u_{xx} + u_{yy} = f(x, y) \quad a \leq x \leq b$$

$$c \leq y \leq d$$

$$u(x, d) = g_N(x)$$



$$u(a, y) = g_W(y)$$



$$u(b, y) = g_E(x)$$

$$u(x, c) = g_S(x)$$

Definindo os respectivos espaçamentos:

$$h = \frac{b-a}{m}, \quad k = \frac{d-c}{n}, \text{ onde } m \text{ e } n \text{ são os respectivos nos. de intervalos}$$

Aproximando $u_{xx} + u_{yy} = f(x, y)$ por diferenças finitas de 2ª ordem:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2} = f_{i,j} = f(x_i, y_j)$$

$$-2\left(\frac{1}{h^2} + \frac{1}{k^2}\right)u_{i,j} + \left(\frac{1}{h^2}\right)(u_{i+1,j} + u_{i-1,j}) + \left(\frac{1}{k^2}\right)(u_{i,j+1} + u_{i,j-1}) = f_{i,j}$$

Definindo $\lambda = \frac{h^2}{k^2}$ e multiplicando por h^2 :

$$-2(1 + \lambda)u_{i,j} + u_{i+1,j} + u_{i-1,j} + \lambda(u_{i,j+1} + u_{i,j-1}) = \overset{h^2}{\downarrow} f_{i,j}$$

Definindo $\mu = 2(1 + \lambda)$ e isolando $u_{i,j}$ temos:

$$u_{i,j} = \frac{1}{\mu} \left\{ u_{i+1,j} + u_{i-1,j} + \lambda(u_{i,j+1} + u_{i,j-1}) - h^2 f_{i,j} \right\}$$

Para o caso particular de domínios quadrados temos: $h = k, \lambda = 1, \mu = 4$

$$u_{i,j} = \frac{1}{4} \left\{ u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - h^2 f_{i,j} \right\}$$

E no caso ainda mais particular para a eq. de Laplace:

$$u_{i,j} = \frac{1}{4} \left\{ u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} \right\}$$

Aplicando uma varredura indicial dos menores valores de i e j para os maiores pode-se implementar o método iterativo de Gauss-Seidel para o problema de Dirichlet da equação de Poisson usando o algoritmo a seguir (**Não é o melhor pseudo-código do mundo, mas é um começo**).

INPUT a,b,c,d -> extremos do domínio
m,n -> no. de intervalos nas direções x e y respectivamente
tol -> tolerância do critério de parada do método iterativo
itmax -> no. máximo de iterações (para evitar loop infinito)

$h=(b-a)/m$, $k=(d-c)/n$, $\lambda = (h*h)/(k*k)$, $\mu = 2*(1+\lambda)$

for i = 0,m -> definição as coordenadas da malha
x(i) = a + i*h
endfor

for j = 0,n
y(j) = c + j*k
endfor

for i = 0,m
for j = 0,n
u(i,j) = 0. -> inicialização da solução
f(i,j) = f(x(i),y(j)) -> definição do termo forçante
endfor
endfor

for i = 0,m -> aplicação das condições de contorno de Dirichlet
u(i,0) = gs(x(i))
u(i,n) = gn(x(i))
endfor

for j = 1,n-1
u(0,j) = gw(y(j))
u(m,j) = ge(y(j))
endfor

```
norm = 1e+02          -> valor inicial arbitrário da norma
l = 1                  -> início do loop do método de Gauss-Seidel

"TAG"

for i = 1,m-1          -> laço para os pontos interiores
for j = 1,n-1
  u(i,j) = (u(i+1,j)+u(i-1,j)+lambda*(u(i,j+1)+u(i,j-1))-h*h*f(i,j))/(mu)
endfor
endfor

z = max(u)             -> norma do máximo da linguagem escolhida

if (abs(z-norm).le. tol) then
  write(u)              -> comando de impressão/plotagem/saída da linguagem escolhida
  write("erro=", abs(z-norm))
  STOP
else
  norm = z
endif

if (l.ge.itmax) then
  write(u)              -> comando de impressão/plotagem/saída da linguagem escolhida
  write("erro=", abs(z-norm))
  write ("WARNING - no. max. Iterações atingido")
  STOP
endif

l = l + 1

Goto "TAG"
```

3. Approximate the solutions to the following elliptic partial differential equations, using Algorithm 12.1:

a. $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0, \quad 0 < x < 1, \quad 0 < y < 1;$

$$u(x, 0) = 0, \quad u(x, 1) = x, \quad 0 \leq x \leq 1;$$

$$\boxed{} \quad u(1, y) = y, \quad 0 \leq y \leq 1.$$

Use $h = k = 0.2$, and compare the results to the actual solution $u(x, y) = xy$.

Vamos repetir o problema com $h = k = 1/3$

Vamos mudar uma condição de contorno para $du/dx=y$ em $x=0$

O que deve acontecer ?

MAP 2320 – MÉTODOS NUMÉRICOS EM EQUAÇÕES DIFERENCIAIS II

2º Semestre - 2019

Roteiro do curso

- Introdução
- Séries de Fourier
- **Método de Diferenças Finitas**
- Equação do calor transiente (parabólica)
- **Equação de Poisson (elíptica)**
- Equação da onda (hiperbólica)