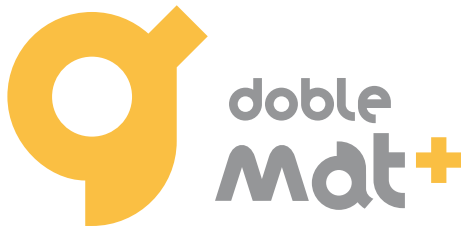


Tema 7: Resolución numérica de EDPs elípticas

Alicia Cordero, Juan R. Torregrosa



- ➊ Introducción
- ➋ Métodos iterativos estacionarios
 - Método de Jacobi
 - Método de Gauss-Seidel
 - Métodos SOR
- ➌ Problemas de contorno elípticos
 - Error de truncamiento y consistencia
 - Métodos iterativos en problemas elípticos
- ➍ Referencias

Sistema de ecuaciones lineales de n ecuaciones y n incógnitas

$$Ax = b,$$

donde A es una matriz real $n \times n$ y b un vector de \mathbb{R}^n .

Método estacionario

Se dice que un método iterativo es **estacionario** si la matriz de iteración H es constante en todo el proceso.

- **Métodos iterativos estacionarios** $x = Hx + d$, H matriz $n \times n$, $d \in \mathbb{R}^n$
 - Jacobi,
 - Gauss-Seidel,
 - Relajación.

Sea $A = (a_{ij})$ la matriz del sistema lineal tal que $a_{ii} \neq 0$, $i = 1, 2, \dots, n$, y consideremos la partición (splitting) de A de la forma

$$A = L + D + U,$$

donde

- L es la parte estrictamente triangular inferior de A
- D es la diagonal principal de A
- U es la parte estrictamente triangular superior de A

El **Método de Jacobi** es un método estacionario en el que $M = D$ y $N = -(L + U)$, por lo que su expresión iterativa es

$$x^{(k+1)} = -D^{-1}(L + U)x^{(k)} + D^{-1}b, \quad k = 0, 1, \dots$$

Expresión escalar del proceso iterativo:

$$\left. \begin{array}{rcl} x_1^{(k+1)} & = & -\frac{a_{12}}{a_{11}}x_2^{(k)} - \frac{a_{13}}{a_{11}}x_3^{(k)} - \dots - \frac{a_{1n}}{a_{11}}x_n^{(k)} + \frac{b_1}{a_{11}} \\ x_2^{(k+1)} & = & -\frac{a_{21}}{a_{22}}x_1^{(k)} - \frac{a_{23}}{a_{22}}x_3^{(k)} - \dots - \frac{a_{2n}}{a_{22}}x_n^{(k)} + \frac{b_2}{a_{22}} \\ \vdots & \vdots & \vdots \\ x_n^{(k+1)} & = & -\frac{a_{n1}}{a_{nn}}x_1^{(k)} - \frac{a_{n2}}{a_{nn}}x_2^{(k)} - \dots - \frac{a_{nn-1}}{a_{nn}}x_{n-1}^{(k)} + \frac{b_n}{a_{nn}} \end{array} \right\}$$

La matriz de iteración es, en este caso,

$$H_J = -D^{-1}(L + U) = I - D^{-1}A.$$

Teorema

Supongamos que los elementos diagonales de la matriz $A = (a_{ij})$ son no nulos y que A es estrictamente diagonal dominante, es decir,

$$|a_{ii}| > \sum_{k=1, k \neq i}^n |a_{ik}|, \quad i = 1, 2, \dots, n,$$

entonces el método de Jacobi es convergente para cualquier aproximación inicial $x^{(0)}$.

Ejemplo

Consideremos el sistema de tamaño 4×4

$$\left. \begin{array}{rrcr} 10x_1 & -x_2 & +2x_3 & & = & 6 \\ -x_1 & +11x_2 & -x_3 & +3x_4 & = & 25 \\ 2x_1 & -x_2 & +10x_3 & -x_4 & = & -11 \\ & 3x_2 & -x_3 & +8x_4 & = & 15 \end{array} \right\}$$

Aplicando el método de Jacobi y el criterio de parada

$$\frac{\|x^{(k+1)} - x^{(k)}\|_{\infty}}{\|x^{(k+1)}\|_{\infty}} < 10^{-3}$$

se obtienen los resultados

k	0	1	2	3	4	5	...	9	10
$x_1^{(k)}$	0.0000	0.0600	1.0473	0.9326	1.0152	0.9890		0.9997	1.0001
$x_2^{(k)}$	0.0000	2.2727	1.7159	2.0533	1.9537	2.0114		2.0004	1.9998
$x_3^{(k)}$	0.0000	-1.1000	-0.8052	-1.0493	-0.9681	-1.0103		-1.0004	-0.9999
$x_4^{(k)}$	0.0000	1.8750	0.8852	1.1309	0.9739	1.0214		1.0006	0.9999

Teniendo en cuenta que la solución exacta es $\bar{x} = (1, 2, -1, 1)^T$, obtenemos

$$\|x^{(10)} - \bar{x}\|_{\infty} = 0.0002.$$

Sustituimos a la derecha de la expresión de punto fijo las componentes del iterado k o del $k + 1$ si ya se tienen

$$\left. \begin{aligned} x_1^{(k+1)} &= -\frac{a_{12}}{a_{11}}x_2^{(k)} - \frac{a_{13}}{a_{11}}x_3^{(k)} - \dots - \frac{a_{1n}}{a_{11}}x_n^{(k)} + \frac{b_1}{a_{11}} \\ x_2^{(k+1)} &= -\frac{a_{21}}{a_{22}}x_1^{(k+1)} - \frac{a_{23}}{a_{22}}x_3^{(k)} - \dots - \frac{a_{2n}}{a_{22}}x_n^{(k)} + \frac{b_2}{a_{22}} \\ &\vdots \\ x_n^{(k+1)} &= -\frac{a_{n1}}{a_{nn}}x_1^{(k+1)} - \frac{a_{n2}}{a_{nn}}x_2^{(k+1)} - \dots - \frac{a_{nn-1}}{a_{nn}}x_{n-1}^{(k+1)} + \frac{b_n}{a_{nn}} \end{aligned} \right\}$$

Volviendo a utilizar la partición de la matriz A del sistema

$$A = L + D + U$$

y tomando $M = D + L$ y $N = -U$, obtenemos el **método de Gauss-Seidel** cuya expresión iterativa es

$$x^{(k+1)} = -(D + L)^{-1}Ux^{(k)} + (D + L)^{-1}b, \quad k = 0, 1, \dots$$

- En cada iteración debemos resolver el sistema $(D + L)x^{(k+1)} = b - Ux^{(k)}$
- En el método de Gauss-Seidel las componentes de $x^{(k+1)}$ que ya conocemos se utilizan en la propia iteración $k + 1$

Convergencia del método de Gauss-Seidel

La matriz de iteración en este caso es $H_{GS} = -(D + L)^{-1}U$.

Teorema

Supongamos que los elementos diagonales de la matriz $A = (a_{ij})$ son no nulos y que A es estrictamente diagonal dominante, entonces el método de Gauss-Seidel es convergente para cualquier aproximación inicial $x^{(0)}$.

Teorema

Si $\|H\| < 1$ para cualquier norma matricial, entonces la sucesión $\{x^{(k)}\}$ obtenida por un método iterativo estacionario converge a la solución del sistema, para cualquier $x^{(0)}$ inicial, y se satisfacen las **cotas de error**:

$$\|\bar{x} - x^{(k)}\| \leq \|H\|^k \|\bar{x} - x^{(0)}\|,$$

$$\|\bar{x} - x^{(k)}\| \leq \frac{\|H\|^k}{1 - \|H\|} \|x^{(1)} - x^{(0)}\|.$$

Ejemplo

$$\left. \begin{array}{rrrrr} 10x_1 & -x_2 & +2x_3 & & = & 6 \\ -x_1 & +11x_2 & -x_3 & +3x_4 & = & 25 \\ 2x_1 & -x_2 & +10x_3 & -x_4 & = & -11 \\ & 3x_2 & -x_3 & +8x_4 & = & 15 \end{array} \right\}$$

Aplicando el método de Gauss-Seidel y el criterio de parada

$$\frac{\|x^{(k+1)} - x^{(k)}\|_{\infty}}{\|x^{(k+1)}\|_{\infty}} < 10^{-3}$$

se obtienen los resultados

k	0	1	2	3	4	5
$x_1^{(k)}$	0.0000	0.0600	1.0300	1.0065	1.0009	1.0001
$x_2^{(k)}$	0.0000	2.3272	2.0370	2.0036	2.0003	2.0000
$x_3^{(k)}$	0.0000	-0.9873	-1.0140	-1.0025	-1.0003	-1.0000
$x_4^{(k)}$	0.0000	0.8789	0.9844	0.9983	0.9999	1.0000

Teniendo en cuenta que la solución exacta es $\bar{x} = (1, 2, -1, 1)^T$, obtenemos

$$\|x^{(5)} - \bar{x}\|_{\infty} = 0.00002.$$

En los métodos de relajación interviene un parámetro w del que depende, en gran medida, la convergencia del método. Se basan en la descomposición de A de la forma

$$A = \frac{1}{w}D - \frac{1-w}{w}D + L + U = \frac{1}{w}(D + wL) - \left(\frac{1-w}{w}D - U\right).$$

De aquí,

$$\begin{aligned} Ax = b \quad \Leftrightarrow \quad \frac{1}{w}(D + wL)x &= \left(\frac{1-w}{w}D - U\right)x + b \quad \Leftrightarrow \\ (D + wL)x &= ((1-w)D - wU)x + wb. \end{aligned}$$

De esta expresión podemos obtener los métodos iterativos

- Método **SOR1** (successive over relaxation)

$$(D + wL)x^{(k+1)} = ((1-w)D - wU)x^{(k)} + wb$$

- Método **SOR2**

$$(D + wU)x^{(k+1)} = ((1-w)D - wL)x^{(k)} + wb$$

$$\begin{aligned}
 x_1^{(k+1)} &= (1-w)x_1^{(k)} + w \left(-\frac{a_{12}}{a_{11}}x_2^{(k)} - \frac{a_{13}}{a_{11}}x_3^{(k)} - \dots - \frac{a_{1n}}{a_{11}}x_n^{(k)} + \frac{b_1}{a_{11}} \right) \\
 x_2^{(k+1)} &= (1-w)x_2^{(k)} + w \left(-\frac{a_{21}}{a_{22}}x_1^{(k+1)} - \frac{a_{23}}{a_{22}}x_3^{(k)} - \dots - \frac{a_{2n}}{a_{22}}x_n^{(k)} + \frac{b_2}{a_{22}} \right) \\
 &\vdots \\
 x_n^{(k+1)} &= (1-w)x_n^{(k)} + w \left(-\frac{a_{n1}}{a_{nn}}x_1^{(k+1)} - \frac{a_{n2}}{a_{nn}}x_2^{(k+1)} - \dots - \frac{a_{nn-1}}{a_{nn}}x_{n-1}^{(k+1)} + \frac{b_n}{a_{nn}} \right)
 \end{aligned}$$

Si $\bar{x}^{(k+1)}$ denota el iterado $k+1$ del método de Gauss-Seidel, la expresión iterativa vectorial del método SOR1 se puede escribir de la forma:

Expresión vectorial

$$x^{(k+1)} = (1-w)x^{(k)} + w\bar{x}^{(k+1)}, \quad k = 0, 1, 2, \dots$$

Matriz de iteración

$$H_w = (D + wL)^{-1}((1-w)D - wU).$$

Teorema

La matriz de iteración H_w del método SOR1 verifica que $\rho(H_w) \geq |w - 1|$.

Corolario

Una condición necesaria para que el método SOR1 converja es que $w \in]0, 2[$.

Convergencia del método SOR1

Comentarios

- ✓ La condición $w \in]0, 2[$ es sólo necesaria. Hay tipos de matrices para las cuales es condición necesaria y suficiente.
- ✓ ¿Cuál es el valor óptimo de w ?
- ✓ Teniendo en cuenta que el método SOR1 puede escribirse de la forma

$$x^{(k+1)} = x^{(k)} - wD^{-1}(Lx^{(k+1)} - Dx^{(k)} + Ux^{(k)} + b,$$

el **factor de proporcionalidad** que multiplica el error es w/a_{ii} mientras que en método de Gauss-Seidel era $1/a_{ii}$.

- ✓ Si A es una matriz tridiagonal y $\rho(H_J)$ el radio espectral de la matriz de Jacobi, el parámetro óptimo de relajación viene dado por

$$w_{op} = \frac{2}{1 + (1 - (\rho(H_J))^2)^{1/2}}.$$

Teorema

Si A es una matriz diagonal dominante, entonces el método SOR converge para cualquier $x^{(0)}$ y cualquier valor de w tal que $0 < w \leq 1$.

Teorema

Si A es una matriz simétrica, definida positiva, el método SOR1 converge para cualquier elección de la aproximación inicial $x^{(0)}$ si y sólo si $0 < w < 2$.

Teorema

Si $A = D + L + U$ es una matriz simétrica, definida positiva, entonces el método de Jacobi converge para cualquier elección de la aproximación inicial $x^{(0)}$ si $D - L - U$ es definida positiva.

Ejemplo

Consideremos el sistema

$$\left. \begin{array}{rrcr} 4x_1 & +3x_2 & & = & 24 \\ 3x_1 & +4x_2 & -x_3 & = & 30 \\ & -x_2 & +4x_3 & = & -24 \end{array} \right\}$$

que tiene como solución exacta $\bar{x} = (3, 4, -5)^T$. Aplicando el método de Gauss-Seidel y el criterio de parada

$$\|x^{(k+1)} - x^{(k)}\|_{\infty} < 10^{-7},$$

se obtienen los resultados

k	0	1	2	3	4	5	6	7
$x_1^{(k)}$	1.0000	5.2500	3.1406	3.0878	3.0549	3.0343	3.0214	3.0134
$x_2^{(k)}$	1.0000	3.8125	3.8828	3.9267	3.9542	3.9713	3.9821	3.9888
$x_3^{(k)}$	1.0000	-5.0468	-5.0292	-5.0183	-5.0114	-5.0071	-5.0044	-5.0027

Para que se cumpla el criterio de parada necesitamos 34 iteraciones.

Ejemplo

Aplicamos el método SOR1 con $w = 1,25$ y el criterio de parada

$$\|x^{(k+1)} - x^{(k)}\|_{\infty} < 10^{-7}.$$

Los resultados obtenidos son:

k	0	1	2	3	4	5	6	7
$x_1^{(k)}$	1.0000	6.3125	2.6223	3.1330	2.9570	3.0037	2.9963	3.0000
$x_2^{(k)}$	1.0000	3.5195	3.9585	4.0102	4.0074	4.0029	4.0009	4.0002
$x_3^{(k)}$	1.0000	-6.6501	-4.6004	-5.0966	-4.9734	-5.0037	-4.9982	-5.0003

Para que se cumpla el criterio de parada necesitamos 14 iteraciones.

Las ecuaciones de este tipo surgen de manera natural en el estudio de problemas físicos independientes del tiempo, como la distribución de calor en una región plana, la energía potencial de un punto en el plano bajo la acción de fuerzas gravitacionales y problemas estacionarios acerca de fluidos incompresibles.

La descripción más sencilla de un problema de este tipo es

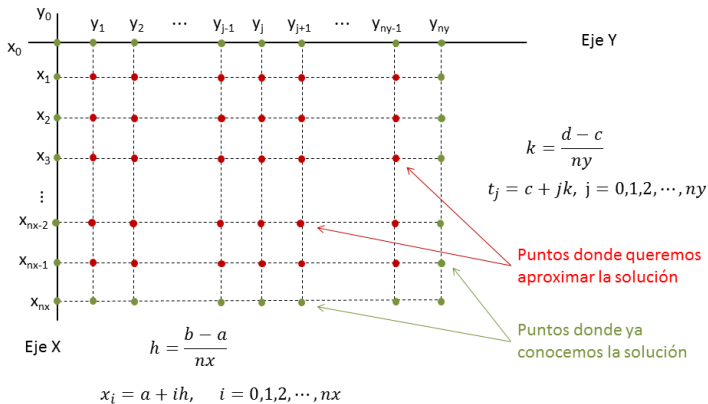
$$\nabla^2 u(x, y) \equiv u_{xx}(x, y) + u_{yy}(x, y) = f(x, y), \quad (x, y) \in R,$$

$$u(x, y) = g(x, y) \text{ para } (x, y) \in S$$

donde $R = \{(x, y) : a < x < b, c < y < d\}$ y S es la frontera de R

Mediante **diferencias finitas** transformamos el problema en un **sistema de ecuaciones lineales** (ya que la edp es lineal) cuya solución nos dará valores aproximados del problema de contorno en los puntos elegidos.

Ecuaciones elípticas



$$h = \frac{b-a}{nx}, \quad k = \frac{d-c}{ny}, \quad x_i = a + ih, i = 0, 1, \dots, nx; \quad y_j = c + jk, j = 0, 1, \dots, ny$$

$$u_{xx}(x_i, y_j) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}, \quad u_{yy}(x_i, y_j) \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2}$$

Sustituyendo en la ecuación

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

y llamando $\lambda = \frac{h}{k}$, resulta

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

con $i = 1, 2, \dots, nx - 1$ y $j = 1, 2, \dots, ny - 1$.

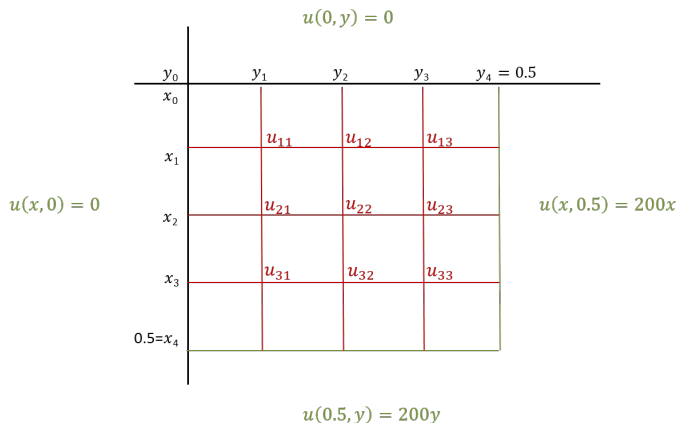
Con las condiciones de contorno

$$\begin{aligned} u_{0,j} &= g(a, y_j), & u_{nx,j} &= g(b, y_j), & j &= 0, 1, \dots, ny \\ u_{i,0} &= g(x_i, c), & u_{i,ny} &= g(x_i, d), & i &= 0, 1, \dots, nx \end{aligned}$$

Ejemplo

$$\begin{aligned}u_{xx}(x, y) + u_{yy}(x, y) &= 0, \quad (x, y) \in R = [0, 0.5] \times [0, 0.5], \\u(x, 0) &= 0, \quad u(x, 0.5) = 200x \quad u(0, y) = 0, \quad u(0.5, y) = 200y,\end{aligned}$$

Tomamos $nx = 4$, $ny = 4$



$$4u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j-1} - u_{i,j+1} = 0, \quad i, j = 1, 2, 3$$

Matricialmente:

$$\begin{pmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{pmatrix} \begin{pmatrix} u_{1,1} \\ u_{1,2} \\ u_{1,3} \\ u_{2,1} \\ u_{2,2} \\ u_{2,3} \\ u_{3,1} \\ u_{3,2} \\ u_{3,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 25 \\ 0 \\ 0 \\ 50 \\ 25 \\ 50 \\ 125 \end{pmatrix}.$$

Se puede:

- Buscar una **estructura en la matriz** a la hora de introducirla en el ordenador
- **Redefinir las variables** para que éstas sean w_1, w_2, \dots, w_9
- Resolver el sistema por el **método de Gauss** (Matriz dispersa)
- Aplicar un **método iterativo** para resolver sistemas lineales.

Si llamamos

$$A = \begin{pmatrix} 4 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 4 \end{pmatrix}, \quad I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad O_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

U al vector de incógnitas (9 componentes) y d a los términos independientes (9 componentes), entonces el sistema anterior se puede escribir en la forma

$$\begin{pmatrix} A & -I_3 & O_3 \\ -I_3 & A & -I_3 \\ O_3 & -I_3 & A \end{pmatrix} \begin{pmatrix} U \end{pmatrix} = \begin{pmatrix} d \end{pmatrix}.$$

También podríamos introducir la matriz del sistema mediante diagonales.

Algoritmo para la ecuación de Poisson

$$u_{xx}(x, y) + u_{yy}(x, y) = f(x, y), \quad (x, y) \in R =]a, b[\times]c, d[, \\ u(x, y) = g(x, y), \quad \text{para } (x, y) \in S$$

- **Parámetros de entrada** Funciones f, g ; valores a, b, c, d ; enteros nx, ny ; tolerancia tol ; número máximo de iteraciones $maxiter$.
- **Salida** Aproximaciones $u_{i,j}$ de $u(x_i, y_j)$, $i = 1, 2, \dots, nx, j = 1, 2, \dots, ny - 1$, o un mensaje de fracaso.
- Paso 1. Tomar $h = (b - a)/nx, k = (d - c)/ny$.
- Paso 2. Elegir los nodos del método: $x = a : h : b, y = c : k : d$
- Paso 3. Inicializar con ceros la matriz U de tamaño $(nx + 1) \times (ny + 1)$
- Paso 4. Rellenar con las condiciones de contorno las filas y columnas correspondientes de U
- Paso 5. Tomar $\lambda = h/k, iter = 1$ y $incre = tol + 1, U^{(0)} = zeros$
- Paso 6. Mientras $iter \leq maxiter$ e $incre > tol$
 - Método iterativo para resolver el sistema lineal $\rightarrow U$
 - $incre = norm(U - U^{(0)})$
 - $U^{(0)} = U, iter = iter + 1$
- Paso 7. Analizar porqué el ordenador se ha salido del bucle.

$$\nabla^2 u(x, y) \equiv u_{xx}(x, y) + u_{yy}(x, y) = f(x, y), \quad (x, y) \in R,$$

$$u(x, y) = g(x, y) \text{ para } (x, y) \in S$$

donde $R = \{(x, y) : a < x < b, c < y < d\}$ y S es la frontera de R .

La expresión en diferencias se define mediante el operador lineal L ,

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

y la relación entre error de truncamiento y error exacto es

$$Le_{i,j} = LU_{i,j} - Lu_{i,j} = LU_{i,j} - f_{i,j} = T_{i,j}.$$

Dado que

$$T_{i,j} = \frac{U_{i+1,j} + U_{i-1,j}}{h^2} + \frac{U_{i,j+1} + U_{i,j-1}}{k^2} - 2 \left(\frac{1}{h^2} + \frac{1}{k^2} \right) U_{i,j} - (U_{xx}(x, y) + U_{yy}(x, y))_{i,j},$$

Desarrollando por Taylor en torno a $U_{i,j}$,

$$U_{i+1,j} = U_{i,j} + h(U_x)_{i,j} + \frac{h^2}{2}(U_{xx})_{i,j} + \frac{h^3}{6}(U_{xxx})_{i,j} + \frac{h^4}{24}(U_{xxxx})_{i,j} + \cdots,$$

$$U_{i-1,j} = U_{i,j} - h(U_x)_{i,j} + \frac{h^2}{2}(U_{xx})_{i,j} - \frac{h^3}{6}(U_{xxx})_{i,j} + \frac{h^4}{24}(U_{xxxx})_{i,j} + \cdots,$$

$$U_{i,j+1} = U_{i,j} + k(U_y)_{i,j} + \frac{k^2}{2}(U_{yy})_{i,j} + \frac{k^3}{6}(U_{yyy})_{i,j} + \frac{k^4}{24}(U_{yyyy})_{i,j} + \cdots,$$

$$U_{i,j-1} = U_{i,j} - k(U_y)_{i,j} + \frac{k^2}{2}(U_{yy})_{i,j} - \frac{k^3}{6}(U_{yyy})_{i,j} + \frac{k^4}{24}(U_{yyyy})_{i,j} + \cdots,$$

se obtiene

$$T_{i,j} = \frac{1}{12} \left(h^2 U_{xxxx}(x_i, y_j) + k^2 U_{yyyy}(x_i, y_j) \right) + \cdots.$$

Definiendo $M_4 = \max \left(\max_{R \cup S} \left| (U_{xxxx})_{i,j} \right|, \max_{R \cup S} \left| (U_{yyyy})_{i,j} \right| \right)$, entonces

$$\max |Le_{i,j}| = \max |T_{i,j}| \leq \frac{h^2 + k^2}{12} M_4.$$

$$\max |Le_{i,j}| = \max |T_{i,j}| \leq \frac{h^2 + k^2}{12} M_4.$$

Se puede probar que

$$\max_{R \cup S} |e_{i,j}| \leq \max_S |e_{i,j}| + \frac{b^2 + d^2}{4} \max_R |Le_{i,j}|,$$

y teniendo en cuenta las condiciones de contorno,

$$\max_{R \cup S} |e_{i,j}| \leq \frac{b^2 + d^2}{4} \frac{h^2 + k^2}{12} M_4,$$

por lo que $e_{i,j} \rightarrow 0$ cuando $h, k \rightarrow 0$, por lo que la consistencia queda probada.

Dado que el error de truncamiento $T_{i,j}$ es de orden $O(h^2 + k^2)$, la **extrapolación de Richardson** es una buena opción para mejorar la precisión de los resultados.

Escribimos el sistema a resolver de la forma:

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

$$i = 1, 2, \dots, nx - 1, j = 1, 2, \dots, ny - 1.$$

Método de Jacobi

Para $i = 2 : nx$

Para $j = 2 : ny$

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

Fin para j

Fin para i

Criterio de parada

$$\max_i \left(\max_j \left(\left| u_{i,j}^{(k+1)} - u_{i,j}^{(k)} \right| \right) \right) < tol.$$

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

$$i = 1, 2, \dots, nx - 1, j = 1, 2, \dots, ny - 1.$$

Método de Gauss-Seidel

Para $i = 2 : nx$

Para $j = 2 : ny$

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

Fin para j

Fin para i

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

$i = 1, 2, \dots, nx - 1, j = 1, 2, \dots, ny - 1.$

Método de Sobrerelajación (SOR)

Para $i = 2 : nx$

Para $j = 2 : ny$

$$\bar{u}_{i,j}^{(k+1)} = \frac{1}{2(\lambda^2 + 1)} [\lambda^2(u_{i,j+1}^{(k)} + u_{i,j-1}^{(k+1)}) + u_{i+1,j}^{(k)} + u_{i-1,j}^{(k+1)}] - \frac{h^2 f(x_i, y_j)}{2(\lambda^2 + 1)},$$

$$u_{i,j}^{(k+1)} = (1 - w)u_{i,j}^{(k)} + w\bar{u}_{i,j}^{(k+1)}$$

Fin para j

Fin para i

El parámetro w se llama **parámetro de relajación** y su valor va a determinar si el método es o no convergente.

Ejemplo

$$u_{xx}(x, y) + u_{yy}(x, y) = xe^y, \quad (x, y) \in R =]0, 2[\times]0, 1[,$$

$$u(x, 0) = x, u(x, 1) = xe, \quad u(0, y) = 0, u(2, y) = 2e^y$$

Solución exacta: $u(x, y) = xe^y$

Criterio de parada: $\left| u_{i,j}^{(l)} - u_{i,j}^{(l-1)} \right| \leq 10^{-10}$

i	j	x_i	y_j	$u_{i,j}$	$u(x_i, y_j)$	$ u_{i,j} - u(x_i, y_j) $
1	1	0.3333	0.2	0.40726	0.40713	1.3e-4
1	2	0.3333	0.4	0.49748	0.49727	2.08e-4
1	3	0.3333	0.6	0.60760	0.60737	2.23e-4
1	4	0.3333	0.8	0.74201	0.74185	1.6e-4
2	1	0.6667	0.2	0.81452	0.81427	2.5e-4
2	2	0.6667	0.4	0.99496	0.99455	4.08e-4
2	3	0.6667	0.6	1.21520	1.21470	4.37e-4
2	4	0.6667	0.8	1.48400	1.48370	3.15e-4
3	1	1.0	0.2	1.22180	1.22140	3.64e-4
3	2	1.0	0.4	1.49240	1.49180	5.8e-4
3	3	1.0	0.6	1.82270	1.82210	6.24e-4
⋮	⋮					



S. LARSSON, V. THOMÉE, *Partial differential equations with numerical methods*, Springer, Berlin, 2016.



T. MYINT-U, L. DEBNATH, *Partial Differential Equations for Scientist and Engineers*, Ed. North-Holland, New York, 1987.



R. BURDEN, J. FAIRES, *Análisis Numérico*, Ed. Thompson, 2002.



S.C. CHAPRA, R.P. CANALE, *Métodos numméricos para ingenieros*, Ed. McGraw-Hill, México D.F., 2006.



L. LAPIDUS, G. PINDER, *Numerical solution of partial differential equations in science and engineering*, Ed. Wiley Interscience Publication, New York, 1999.



A. CORDERO, J.L. HUESO, E. MARTÍNEZ, J.R. TORREGROSA, *Problemas resueltos de métodos numéricos*, Ed. Thompson, 2006.



J. MATHEWS, K. FINK, *Métodos Numéricos con Matlab*, Ed. Prentice-Hall, 1999.