thm (Young 1950)

1. If $\rho(B_\omega) < 1$, then $0 < \omega < 2$.

2. Assume A is symmetric, block tridiagonal, and positive definite (defined later).

Then $\omega_* = \dfrac{2}{1+\sqrt{1-\rho(B_J)^2}}$ is the optimal SOR parameter in the sense that

$\rho(B_{\omega_*}) = \min\limits_{0<\omega<2} \rho(B_\omega) = \omega_* - 1 < \rho(B_{GS}) < \rho(B_J) < 1$.

pf : Math 571 (sometimes)

return to example : $\omega_* = \dfrac{2}{1+\sqrt{1-\rho(B_J)^2}} = \dfrac{2}{1+\sqrt{1-(\frac{1}{2})^2}} = \dfrac{4}{2+\sqrt{3}} = 1.0718$

k	$x_1^{(k)}$	$x_2^{(k)}$	$\|\|e_k\|\|$	$\|\|e_k\|\|/\|\|e_{k-1}\|\|$
0	0.0000	0.0000	1.0000	$\cdots$
1	0.5359	0.8231	0.4641	0.4641
2	0.9385	0.9798	0.0615	0.1325
3	0.9936	0.9980	0.0064	0.1047
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
∞	1	1	0	$\rho(B_{\omega_*}) = \omega_* - 1 = 0.0718$

Hence optimal SOR converges faster than GS. 13 Tue 3/2

def : A is positive definite if $x^TAx > 0$ for all $x \neq 0$

ex 1 : $A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$ is positive definite

pf : $x^TAx = (x_1, x_2)\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = (x_1, x_2)\begin{pmatrix} 2x_1 - x_2 \\ -x_1 + 2x_2 \end{pmatrix}$

$$= 2(x_1^2 + x_2^2) - 2x_1x_2 = x_1^2 + x_2^2 + (x_1 - x_2)^2 \geq 0$$

If $x \neq 0$, then either $x_1 \neq 0$ or $x_2 \neq 0$, but in any case we have $x^TAx > 0$. ok

ex 2 : $A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ is positive definite : hw

ex 3 : $A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ is not positive definite

pf : $x^TAx = (x_1, x_2)\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1^2 + x_2^2 + 4x_1x_2$: indefinite

for example : $x = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \Rightarrow x^TAx = 1$, $x = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \Rightarrow x^TAx = -2$ ok ex 4

$$A_h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{pmatrix} \ : \ \text{dimension } n \times n \ , \ h = \frac{1}{n+1}$$

The matrix A_h represents the finite difference operator $-D_+D_-$; A_h is symmetric, tridiagonal, and positive definite, and hence Young's theorem applies.

note : The real advantage of iterative methods, in comparison with direct methods, is for BVPs in more than one dimension.

3.9 two-dimensional BVP

problem : A metal plate has a square shape. The plate is heated by internal sources and the edges are held at a given temperature. Find the temperature at points inside the plate.

$D = \{(x, y) : 0 \le x, y \le 1\}$: plate domain

$\phi(x, y)$: temperature

$f(x, y)$: heat sources , $g(x, y)$: boundary temperature

Then $\phi(x, y)$ satisfies the following two equations.

1. $\underbrace{-\Delta}_{\text{Laplace operator}}\phi = -\nabla^2\phi = -\left(\dfrac{\partial^2\phi}{\partial x^2} + \dfrac{\partial^2\phi}{\partial y^2}\right) = f$ for (x, y) in D : Poisson equation

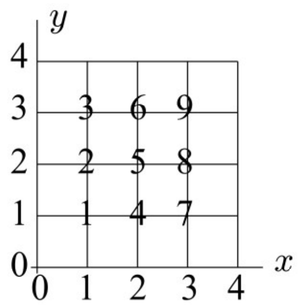
(note : This equation arises in many areas, e.g. if f is a charge/mass distribution, then ϕ is the electrostatic/gravitational potential.)

2. $\phi = g$ for (x, y) on ∂D : Dirichlet boundary condition

finite-difference scheme

$h = \dfrac{1}{n+1}$: mesh size , $(x_i, y_j) = (ih, jh)$, $i, j = 0, \ldots, n+1$: mesh points

ex : $n = 3$, $h = \frac{1}{4}$



$\phi(x_i, y_j)$: exact solution

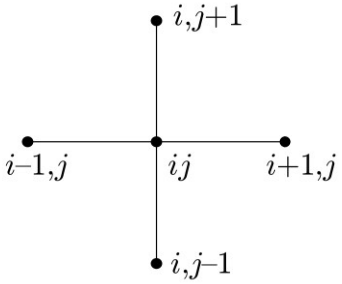
w_{ij} : numerical solution

ordering of mesh points : $w_{11}, w_{12}, \ldots$

$$-\left(D_+^x D_-^x w_{ij} + D_+^y D_-^y w_{ij}\right) = f_{ij} \quad : \quad \text{finite-difference equations}$$

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

$$\frac{1}{h^2}\left(4w_{ij} - w_{i+1,j} - w_{i-1,j} - w_{i,j+1} - w_{i,j-1}\right) = f_{ij}$$



: <u>5-point stencil</u>

Consider what happens near the boundary.

$$(i,j) = (1,1) \quad \Rightarrow \quad \frac{1}{h^2}\left(4w_{11} - w_{21} - w_{01} - w_{12} - w_{10}\right) = f_{11}$$

$$\Rightarrow \quad \frac{1}{h^2}\left(4w_{11} - w_{21} - w_{12}\right) = f_{11} + \frac{1}{h^2}\left(g_{01} + g_{10}\right)$$

Write the equations for w_{ij} in matrix form.

$$\begin{array}{ccc|ccc|ccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
w_{11} & w_{12} & w_{13} & w_{21} & w_{22} & w_{23} & w_{31} & w_{32} & w_{33} \\
\hline
4 & -1 & & -1 & & & & & \\
-1 & 4 & -1 & & -1 & & & & \\
 & -1 & 4 & & & -1 & & & \\
\hline
-1 & & & 4 & -1 & & -1 & & \\
 & -1 & & -1 & 4 & -1 & & -1 & \\
 & & -1 & & -1 & 4 & & & -1 \\
\hline
 & & & -1 & & & 4 & -1 & \\
 & & & & -1 & & -1 & 4 & -1 \\
 & & & & & -1 & & -1 & 4
\end{array}$$

$$A_h w_h = f_h \ , \ A_h = \begin{pmatrix} T & -I & & & \\ -I & T & -I & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -I \\ & & & -I & T \end{pmatrix}$$

$T \,:\, n \times n$, symmetric , tridiagonal

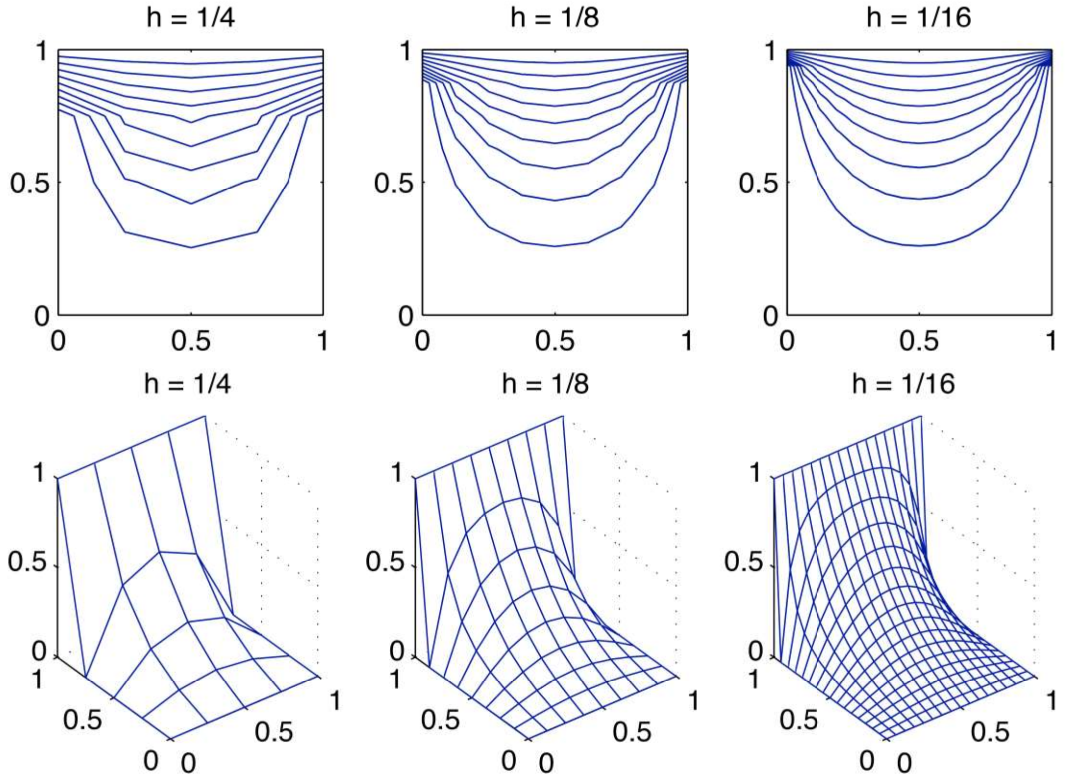
$A_h \,:\, n^2 \times n^2$, symmetric , block tridiagonal , positive definite (pf : omit)

<u>temperature distribution on a metal plate</u> : no heat sources, one side heated

differential equation : $\phi_{xx} + \phi_{yy} = 0$

boundary conditions : $\phi(x, 1) = 1$, $\phi(x, 0) = \phi(0, y) = \phi(1, y) = 0$

finite-difference scheme : $D_+^x D_-^x w_{ij} + D_+^y D_-^y w_{ij} = 0$



<u>above</u> : solution of linear system $A_h w_h = f_h$ for given mesh size h

<u>below</u> : number of iterations k required for each method

initial guess = zero vector, stopping criterion : $||r_k||/||r_0|| \leq 10^{-4}$

Jacobi

h	k	$\rho(B)$
1/4	26	0.7071
1/8	96	0.9239
1/16	334	0.9808

Gauss-Seidel

h	k	$\rho(B)$
1/4	15	0.5000
1/8	51	0.8536
1/16	172	0.9619

optimal SOR

h	k	$\rho(B)$
1/4	9	0.1716
1/8	18	0.4465
1/16	34	0.6735

14 Tue 3/9

<u>note</u>

1. For each method, more iterations are needed as the mesh size $h \to 0$. Hence refining the mesh yields a more accurate solution of the BVP, but the computational cost increases.

2. For a given mesh size h, SOR converges the fastest, then GS, and then J.

3. Explicit formulas for $\rho(B)$ can be derived in this example. (Math 571)

$$\rho(B_J) \;=\; \cos \pi h \;\sim\; 1 - \tfrac{1}{2}\pi^2 h^2$$

$$\rho(B_{GS}) \;=\; \cos^2 \pi h \;\sim\; 1 - \pi^2 h^2$$

$$\rho(B_{\omega_*}) \;=\; \frac{2}{1 + \sqrt{1 - \rho(B_J)^2}} - 1 \;=\; \frac{1 - \sin \pi h}{1 + \sin \pi h} \;\sim\; \frac{1 - \pi h}{1 + \pi h} \;\sim\; 1 - 2\pi h$$

This shows that $\rho(B) \to 1$ as $h \to 0$ (confirming that the iteration slows down as the mesh is refined). The formulas also show that $\rho(B_{\omega_*}) < \rho(B_{GS}) < \rho(B_J) < 1$ (confirming that SOR converges the fastest, then GS, and then J).

4. Consider what happens if Gaussian elimination is used instead of J/GS/SOR.

$$\begin{pmatrix}
4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & \rlap{0}{\,*} & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & \rlap{0}{\,*} & \rlap{0}{\,*} & -1 & 0 & 0 & 0 \\
-1 & \rlap{0}{\,*} & \rlap{0}{\,*} & 4 & -1 & \rlap{0}{\,*} & -1 & 0 & 0 \\
0 & -1 & \rlap{0}{\,*} & -1 & 4 & -1 & \rlap{0}{\,*} & -1 & 0 \\
0 & 0 & -1 & \rlap{0}{\,*} & -1 & 4 & \rlap{0}{\,*} & \rlap{0}{\,*} & -1 \\
0 & 0 & 0 & -1 & \rlap{0}{\,*} & \rlap{0}{\,*} & 4 & -1 & \rlap{0}{\,*} \\
0 & 0 & 0 & 0 & -1 & \rlap{0}{\,*} & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & \rlap{0}{\,*} & -1 & 4
\end{pmatrix}$$

a) A_h is a <u>band matrix</u>, i.e. $a_{ij} = 0$ for $|i - j| > m$, where m is the <u>bandwidth</u> (in this example we have $m = 3$).

b) As the elimination proceeds, zeros inside the band can become non-zero (this is called <u>fill-in</u>), but zeros outside the band are preserved. Hence we can adjust the limits on the loops to reduce the operation count for Gaussian elimination from $O(n^3)$ to $O(nm^2)$.

c) Due to fill-in, more memory needs to be allocated than is required for the original matrix A_h. This is a disadvantage in comparison with iterative methods like J/GS/SOR which preserve the <u>sparsity</u> of A_h.

<u>final comments on linear systems</u>

1. <u>comparison of operation counts</u> : two-dimensional BVP

mesh size : $h = \frac{1}{n+1}$

typical equation : $\frac{1}{h^2}(4w_{ij} - w_{i+1,j} - w_{i-1,j} - w_{i,j+1} - w_{i,j-1}) = f_{ij}$

vector w_{ij} has length n^2

matrix A_h has dimension $n^2 \times n^2$ and bandwidth $m = n$

a) Gaussian elimination : $O((n^2)^3) = O(n^6)$ ops

banded Gaussian elimination : $O(n^2 m^2) = O(n^4)$ ops

b) iterative methods

cost per iteration : $O(n^2)$ ops (roughly the same for J/GS/SOR)

stopping criterion : $\dfrac{||r_k||}{||r_0||} = \epsilon \;\Rightarrow\; \rho(B)^k = \epsilon \;\Rightarrow\; k = \dfrac{\log \epsilon}{\log \rho(B)}$

J , GS $\Rightarrow\; \rho(B) \sim 1 - ch^2 \;\Rightarrow\; \log \rho(B) \sim \log(1 - ch^2) \sim -ch^2$

$$\Rightarrow\; k \sim \frac{\log \epsilon}{-ch^2} = O(n^2) \text{ iterations}$$

$$\Rightarrow\; \text{total cost} = O(n^2) \times O(n^2) = O(n^4) \text{ ops}$$

15
Thu
3/11

SOR $\Rightarrow\; \rho(B) \sim 1 - ch$

$$\Rightarrow\; k \sim \frac{\log \epsilon}{-ch} = O(n) \text{ iterations}$$

$$\Rightarrow\; \text{total cost} = O(n^2) \times O(n) = O(n^3) \text{ ops}$$

2. <u>developments after SOR</u>
   conjugate gradient method
   FFT = fast Fourier transform
   multigrid
   GMRES
   preconditioning : $Ax = b \rightarrow PAx = Pb$
   software
   parallel