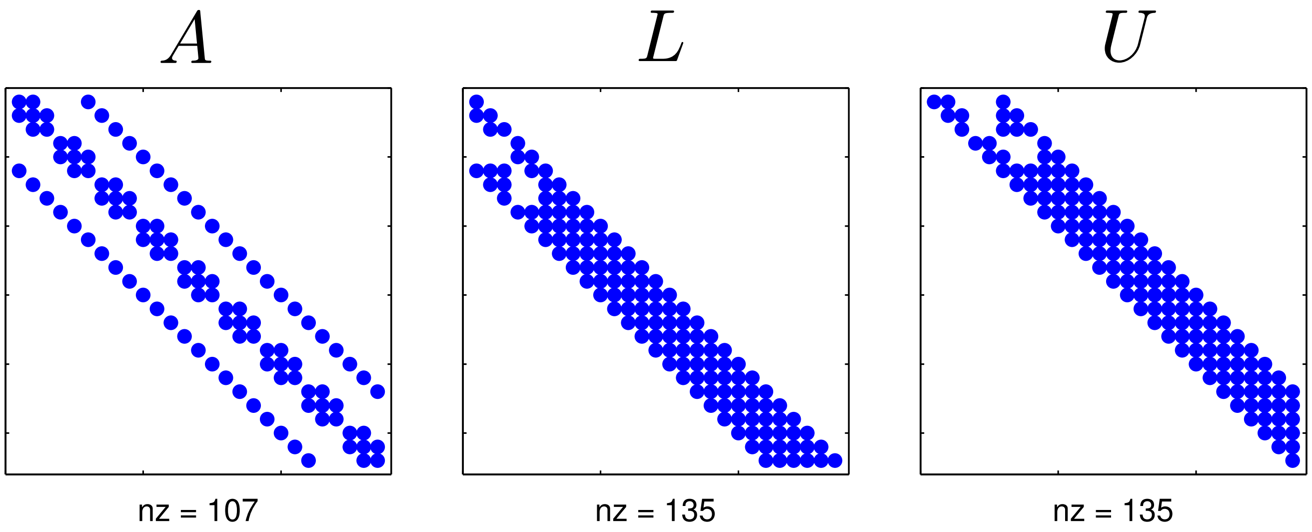
9. Iterative Methods for Large Linear Systems

- introduction
- splitting method
- Jacobi method
- Gauss-Seidel method
- successive overrelaxation (SOR)

Large sparse linear systems

consider solving $Ax = b$ when A is **sparse** and the dimension of A is **huge**



factorization methods are sometimes not a good technique because

- the number of non-zero entries in the factors is increased due to fill-in
- storing the factors L and U will require much more storage

Application on solving PDE

large sparse matrices arise in the numerical solution of PDE/ODE

ODE

$$-u''(x) = f(x), \qquad 0 < x < 1, \qquad \text{where } u(0) \text{ and } u(1) \text{ are given}$$

if we discretize the spatial variable by

$$x_0 = 0, \quad x_1 = h, \quad x_2 = 2h, \ldots, \quad x_i = ih, \ldots, \quad x_n = nh = 1$$

and apply a *good approximation* of $u''(x)$ as

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

then by denoting $u_i = u(x_i)$ we can try to solve the above approximation exactly

$$-u_{i-1} + 2u_i - u_{i+1} = h^2 f(x_i), \quad i = 1, \ldots, n-1$$

this is actually a system of $n-1$ equations with $n-1$ unknowns $u_1, \ldots, u_{n-1}$

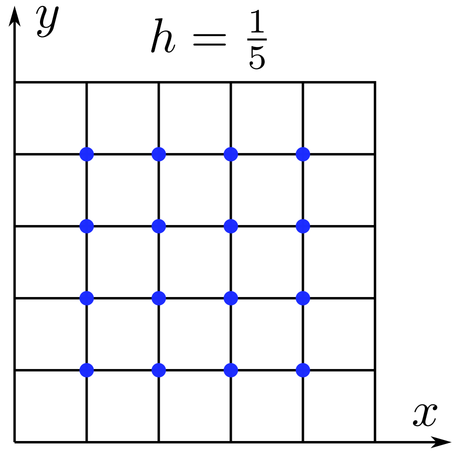
$$Au = b$$

$$A = \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & \ddots & & \\ & & \ddots & \ddots & -1 & \\ & & & 1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} h^2 f(x_1) + u(0) \\ h^2 f(x_2) \\ h^2 f(x_3) \\ \vdots \\ h^2 f(x_{n-2}) \\ h^2 f(x_{n-1}) + u(1) \end{bmatrix}$$

- we obtain an *approximate* ODE solution to by solving linear equations
- by making h small, the solution is more accurate, but # of variables increases
- we can show that A is nonsingular (and pdf), hence the solution is unique
- A is tri-diagonal (extremely sparse)
- in fact, we can solve by Cholesky's method; no need to use iterative methods

PDE: Poisson's equation with variables $(x, y) \in [0,1] \times [0,1]$

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = f, \quad \text{with some boundary condition}$$



$(x_i, y_j) = (ih, jh)$
$i, j = 0, 1, 2, \ldots, m$

use the approximations

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_i) \approx \frac{u(x_{i-1}, y_i) - 2u(x_i, y_j) + u(x_{i+1}, y_j)}{h^2}$$

$$\frac{\partial^2 u}{\partial y^2}(x_i, y_i) \approx \frac{u(x_i, y_{j-1}) - 2u(x_i, y_j) + u(x_{i+1}, y_{j+1})}{h^2}$$

let $u_{i,j} = u(x_i, y_j)$; we treat the above approximations as equations

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

with some arrangment

$$-u_{i,j-1} - u_{i-1,j} + 4u_{i,j} - u_{i+1,j} - u_{i,j+1} = h^2 f_{i,j}, \quad i,j = 1,2,\ldots,m-1$$

we can order $u_{i,j}$ by sweeping by rows and define the vector

$$u = [\ u_{1,1},\ldots,u_{m-1,1},u_{1,2},\ldots,u_{m-1,2},\ldots,u_{m-1,m-1}\ ]^T$$

we can now write the equations as $Au = b$

$$A = \begin{bmatrix} T & -I & & & & \\ -I & T & -I & & & \\ & -I & T & \ddots & & \\ & & \ddots & \ddots & -I & \\ & & & 1 & 2 & -I \\ & & & & -I & T \end{bmatrix}, \quad T = \begin{bmatrix} 4 & -1 & & & & \\ -1 & 4 & -1 & & & \\ & -1 & 4 & \ddots & & \\ & & \ddots & \ddots & -1 & \\ & & & 1 & 4 & -1 \\ & & & & -1 & 4 \end{bmatrix}$$

A has dimension $n \times n$ where $n = (m-1)^2$

- for PDE, the problem dimension is much higher ($n = (m-1)^2$)
- A has block-tridiagonal structure and the semi-band width is m
- A is symmetric, nonsingular and even positive definite
- if solved by Cholesky, it's known that the cost of solving a banded system is

$$ns^2/2 \qquad \text{where } s \text{ is the semi-band width}$$

  so the cost is about $\approx (1/2)m^4$
- the Cholesky factor is not nearly so sparse, and requires storage of $ns \approx m^3$
- every we halve h, m is doubled, the cost and storage are increased by a factor of 16 and 8, respectively

when n is fairly large, the iterative methods can require less cost and storage

Splitting methods

we solve the system $Ax = b$ where $A \in \mathbf{R}^{n \times n}$ and A is nonsingular

we *split* A into a difference

$$A = M - N$$

where M is such that solving $Mz = f$ is *easy*; then we have

$$(M - N)x = b \quad \Longrightarrow \quad Mx = Nx + b \quad \Longrightarrow \quad x = M^{-1}Nx + M^{-1}b$$

this suggests the following iteration

$$x^{(k+1)} = M^{-1}Nx^{(k)} + M^{-1}b$$

until the sequence converges

Convergence of splitting methods

we can write the iteration matrix as

$$T = M^{-1}N = M^{-1}(M - A) = I - M^{-1}A$$

idea: T should be less than one in some sense (even better if $M \approx A$)

Theorem: the iteration

$$x^{(k+1)} = Tx^{(k)} + M^{-1}b$$

converges for all $x^{(0)}$ if and only if

$$\text{spectral radius of } T \triangleq \rho(T) \triangleq \max |\lambda(T)| < 1$$

$i.e.$, the largest magnitude of all eigenvalues of T is less than 1

Proof sketch.

- we can show that for any T, $\rho(T) = \inf \|T\|$ (not obvious)

  where the infimum is taken over all *induced matrix norm* $\| \cdot \|$

- if $\rho(T) < 1$ then there exists an induced norm such that $\|T\| < 1$

- the iteration $x^{(k+1)} = Tx^{(k)} + c$ has the closed-form formula:

$$x^{(k)} = T^k x^{(0)} + \sum_{j=0}^{k-1} T^j c$$

- the term $T^k x^{(0)}$ must go to 0 as $k \to \infty$ if $\rho(T) < 1$

$$\|T^k x^{(0)}\| \leq \|T^k\| \|x^{(0)}\| \leq \|T\|^k \|x^{(0)}\| \to 0$$

- $\sum_{j=0}^{\infty} T^j c = (I - T)^{-1} c$ (Neumann series)

Jacobi iteration

split the **diagonal** part of A, denoted by D

$$A = D - (D - A)$$

the Jacobi's iteration is

$$x^{(k+1)} = (I - D^{-1}A)x^{(k)} + D^{-1}b$$

or equivalently

$$x^{(k+1)} = x^{(k)} - D^{-1}(Ax^{(k)}) + D^{-1}b = x^{(k)} + D^{-1}r^{(k)}$$

where $r^{(k)} = b - Ax^{(k)}$ is the *residual* after k iterations

note: we should exploit the sparsity structure of A in the implementation

Gauss-Seidel iteration

split the **lower triangular** part of A, denoted by L

$$A = L - (L - A)$$

the Gauss-Seidel iteration is

$$x^{(k+1)} = (I - L^{-1}A)x^{(k)} + L^{-1}b$$

or equivalently

$$x^{(k+1)} = x^{(k)} - L^{-1}(Ax^{(k)}) + L^{-1}b = x^{(k)} + L^{-1}r^{(k)}$$

where $r^{(k)} = b - Ax^{(k)}$ is the *residual* after k iterations

Convergence

Jacobi adn Gauss-Seidel convergence theorem:

1. if A is diagonally dominant, $i.e.$,

   $$|a_{ii}| > \sum_{j \neq i} |a_{ij}|, \quad (1 \leq i \leq n)$$

   then both Jacobi and Gauss-Seidel iteration converge

   moreover, Gauss-Seidel converges faster in the sense that

   $$\rho(T_{GS}) < \rho(T_J)$$

   where T_{GS} and T_J are the iteration matrices for Gauss-Seidel and Jacobi

2. if A is positive semidefinite then both Jacobi and Gauss-Seidel will converge

Successive Over-Relaxation (SOR)

let D, L and U be diagonal, strictly lower and strictly upper triangular parts of A

we split A as follows

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

the SOR iteration is

$$x^{(k+1)} = x^{(k)} - Q^{-1}Ax^{(k)} + Q^{-1}b$$

where Q is *lower triangular* and depends on $\omega \in \mathbf{R}$:

$$Q = \left(\frac{1}{\omega}D + L\right) = \omega^{-1}(D + \omega L)$$

remarks:

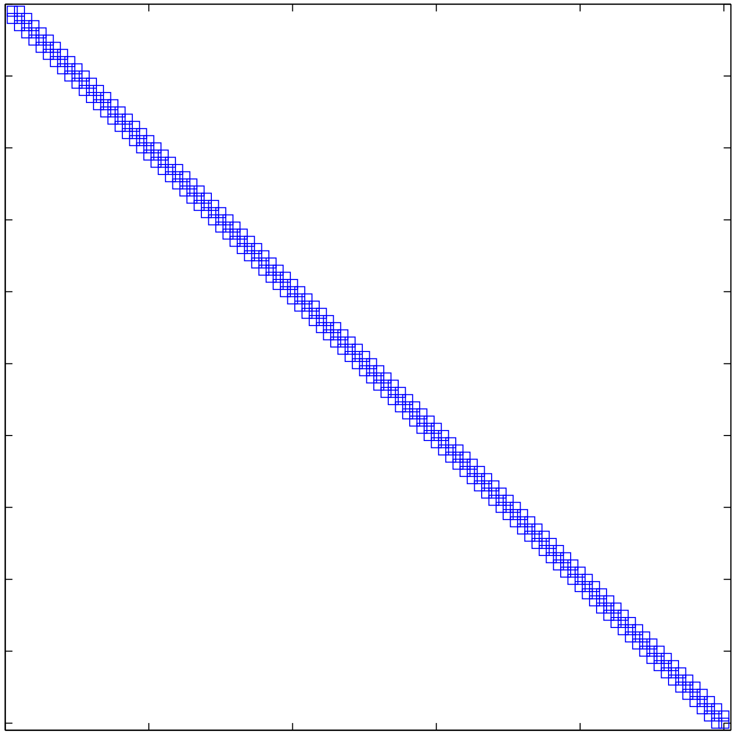
- we are averaging the result of a Gauss-Seidel step with the previous iterate
- ω is the weighting parameter in the average
- when $\omega = 1$, SOR reduces to Gauss-Seidel iteration
- for most problems, the optimal ω is not known
- the best performance of SOR often occurs when $\omega \in [1, 2]$

convergence theorem:

1. if A is positive semidefinite then for any value $\omega \in (0, 2)$, the SOR iteration will converge to the exact solution of $Ax = b$
2. if $\omega < 0$ or $\omega > 2$ then the iteration will not converge

Numerical example

we solve the system $Ax = b$ where A has a triband structure



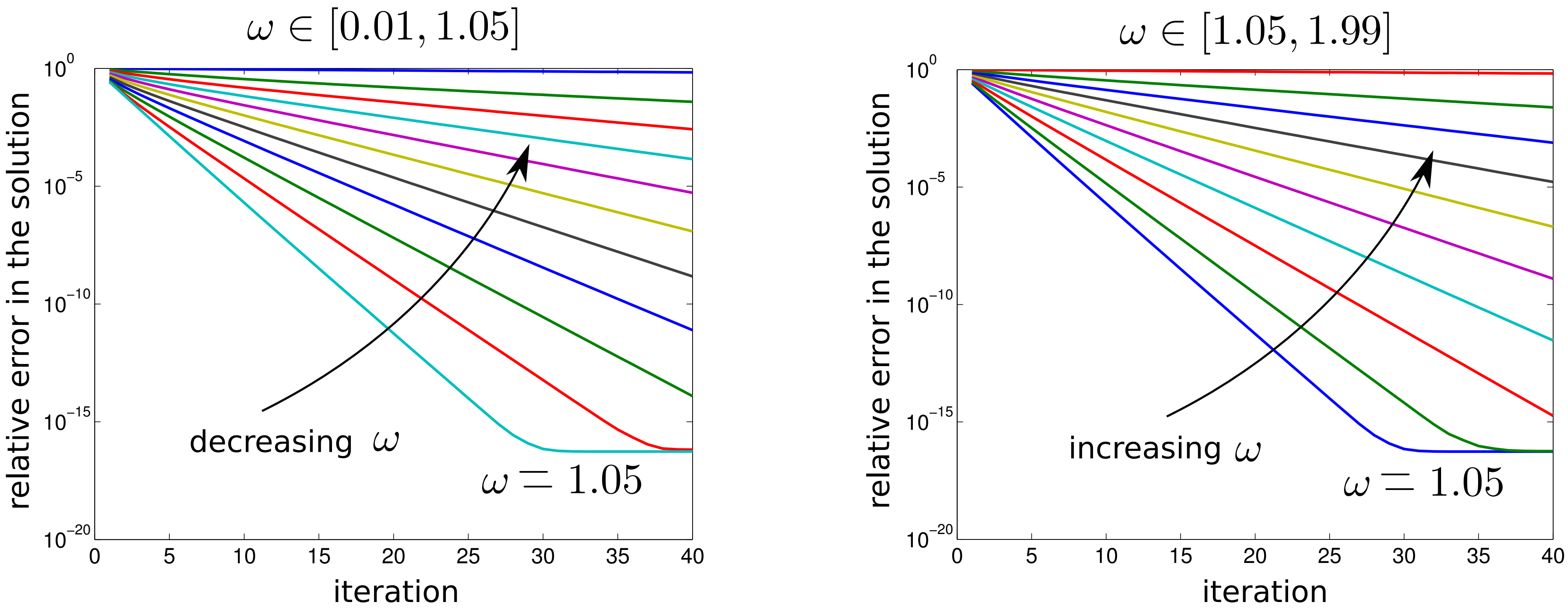
$$A \in \mathbf{R}^{10000 \times 10000}$$

$$a_{ij} = \begin{cases} 4, & i = j \\ -1, & |i-j| = 1 \\ 0, & \text{otherwise} \end{cases}$$

- $x_i = \pm 1$, generated randomly
- b is obtained by multiplying A with x

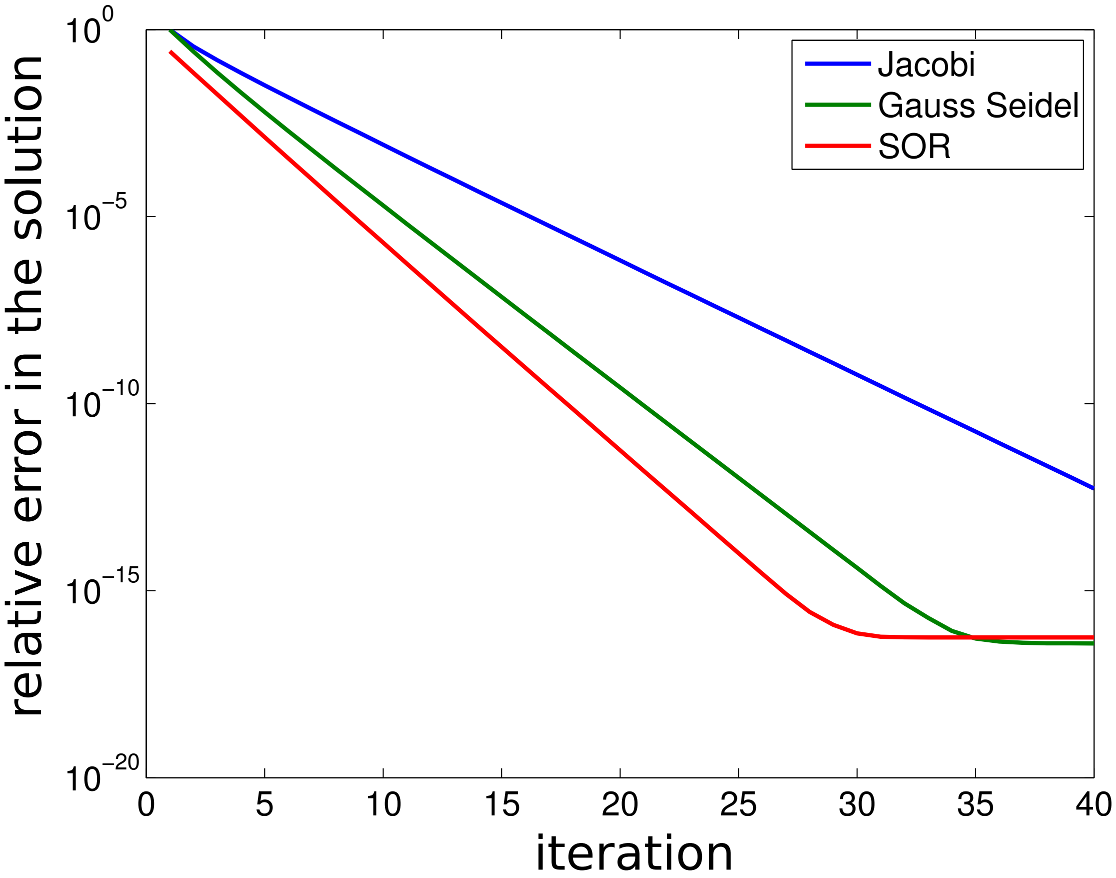
solve the system by Jacobi, Gauss-Seidel, and SOR methods

SOR method



- vary $\omega \in (0, 2)$; the convergence depends heavily on the choice of ω
- as $\omega \to 2$ or $\omega \to 0$, SOR is slow to converge
- we found that in this example, using $\omega = 1.05$ gives the best performance

comparison of the three methods



- Gauss-Seidel is converging faster than Jacobi method
- SOR with $\omega = 1.05$ is converging slightly faster than Gauss-Seidel

Summary

method	**Splitting matrix (M)**
Jacobi	D
Gauss-Seidel	$D + L$
SOR	$\frac{1}{\omega} D + L$

References

Chapter 7 in

J. F. Epperson, *An Introduction to Numerical Methods and Analysis*, John Wiley & Sons, 2007

Chapter 8 in

D. S. Watkins, *Fundamentals of Matrix Computations*, John Wiley & Sons, 2010

Chapter 4 in

D. Kincaid and W. Cheney, *Numerical Analysis: Mathematics of Scientific Computing*, 3rd edition, Brooks & Cole, 2002