

Summary

Numerical Solution of Linear Systems of Equations

Consider the linear systems

$$Ax = b$$

where $A = (a_{ij})_{n \times n}$, x and b are n -vectors. If A is invertible, a unique solution x exists and given by $x = A^{-1}b$.

We are concerned with algorithms for obtaining numerical solutions of the linear systems (1) efficiently, and (2) as accurately as desired, on computers using finite-digit floating point arithmetic.

Two classes of methods: (1) Direct methods, (2) Iterative methods.

1 Direct methods – a revision

Two basic methods: (1) Cramer's rule, (2) Gauss elimination method covered in the first course of numerical methods. Important concept recalled is *LU decomposition*.

Suppose

$$A = LU$$

where L is lower triangular in the form

$$L = \begin{pmatrix} l_{11} & & & \mathbf{0} \\ l_{21} & l_{22} & & \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{pmatrix}$$

and

$$U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & \mathbf{O} & & u_{nn} \end{pmatrix}$$

LU decomposition is useful as now solving $Ax = b$ can be replaced by the more efficient two-step procedure of solving $Ly = b$ for y first, and then $Ux = y$ for x .

Forward substitution for solving $Ly = b$.

$$y_1 := b_1/l_{11};$$

for $k = 2, 3, \dots, n$

$$y_k := \frac{1}{l_{kk}}(b_k - \sum_{j=1}^{k-1} l_{kj}y_j);$$

Backward substitution for solving $Ux = y$.

$$x_n := y_n/u_{nn};$$

for $k = n - 1, n - 2, \dots, 1$

$$x_k := \frac{1}{u_{kk}}(y_k - \sum_{j=k+1}^n u_{kj}x_j);$$

2 Direct methods — special matrices

We consider some matrices of special forms which allows for LU factors of special form and make possible more efficient algorithms for solving the corresponding linear systems.

1. **Tridiagonal matrices** Consider the tridiagonal matrix

$$A = \begin{pmatrix} a_1 & c_1 & & \mathbf{O} \\ b_2 & a_2 & c_2 & \\ & \ddots & \ddots & \ddots \\ & & b_{n-1} & a_{n-1} & c_{n-1} \\ \mathbf{O} & & & b_n & a_n \end{pmatrix}$$

A possible LU decomposition of A has

$$L = \begin{pmatrix} 1 & & & & \mathbf{O} \\ \beta_2 & 1 & & & \\ & \beta_3 & 1 & & \\ & & \ddots & \ddots & \\ \mathbf{O} & & & \beta_n & 1 \end{pmatrix}$$

and

$$U = \begin{pmatrix} \alpha_1 & c_1 & & & \\ & \alpha_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & \alpha_{n-1} & c_{n-1} \\ \mathbf{O} & & & & \alpha_n \end{pmatrix}$$

which are both bi-diagonal matrices. The following algorithm computes the $2n - 1$ unknowns in the LU factors.

$$\begin{aligned} \alpha_1 &:= a_1; \\ \text{for } k &= 2, 3, \dots, n \\ \beta_k &:= b_k / \alpha_{k-1}; \\ \alpha_k &:= a_k - \beta_k c_{k-1}; \end{aligned}$$

2. Symmetric positive definite matrices

- A square matrix A is *symmetric* if $A^T = A$.
- A symmetric matrix A which satisfies

$$x^T A x > 0$$

for all n -vectors $x \neq 0$, is called a *positive definite* matrix.

There are other ways to show that a symmetric matrix is positive definite. For example:

- A matrix A is positive definite if and only if all its eigenvalues are positive.

- A symmetric $n \times n$ matrix A is positive definite if and only if

$$\det(A^{(k)}) > 0, \quad k = 1, 2, \dots, n,$$

where $A^{(k)} = (a_{ij})_{i,j=1}^k$ is the k th principal minors of A .

Now a result on the LU decomposition of a symmetric positive definite matrix.

Theorem If A is a positive definite $n \times n$ matrix, there is a unique lower triangular matrix $L = (l_{ij})_{n \times n}$ with $l_{ii} > 0$, $i = 1, \dots, n$, such that $A = LL^T$.

Such a special decomposition is called *Cholesky's Decomposition*.

Algorithm for Cholesky's decomposition

for $i = 1, 2, \dots, n$

$$l_{ii} := (a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2)^{1/2}$$

for $j = i + 1, i + 2, \dots, n$

$$l_{ji} := (a_{ji} - \sum_{k=1}^{i-1} l_{jk}l_{ik})/l_{ii};$$

3 Error analysis – an introduction

Let x^t be the true solution of

$$Ax = b$$

and x^c one of the many possible computed solutions.

Define the *residual* vector of x^c as

$$r = b - Ax^c.$$

Define the *error* vector of x^c as

$$e = x^t - x^c.$$

We have the relationship

$$r = Ae, \quad \text{or} \quad e = A^{-1}r.$$

From above, “small” r seems to imply “small” e but this is not true as the “size” of the matrix A has to be taken into account.

To measure the size of vectors and matrices, we need the concepts of vector and matrix norms.

4 Vector norms

Consider the vector $x = (x_1, x_2, \dots, x_n)$.

Notation: $\|x\|$ denotes a vector norm of x .

Commonly used vector norms are as follows.

- vector 1-norm: $\|x\|_1 = \sum_{i=1}^n |x_i|$.
- vector 2-norm: $\|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{1/2}$.
- vector ∞ -norm: $\|x\|_\infty = \max_{i=1}^n |x_i|$.

Properties of vector norms

Let x and y be n -vectors.

- $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = 0$.
- $\|\alpha x\| = |\alpha| \|x\|$, $\alpha \in \mathbf{R}$.
- $\|x + y\| \leq \|x\| + \|y\|$.

4.1 Matrix norms

Consider the matrix $A = (a_{ij})_{n \times n}$.

Notation: $\|A\|$ denotes a matrix norm of A .

Commonly used matrix norms are as follows.

- matrix 1-norm: $\|A\|_1 = \max_{j=1}^n (\sum_{i=1}^n |a_{ij}|)$ (the maximum of column sums).
- matrix F -norm: $\|A\|_F = (\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2)^{1/2}$.

- matrix ∞ -norm: $\|A\|_\infty = \max_{i=1}^n (\sum_{j=1}^n |a_{ij}|)$ (the maximum of row sums).
- matrix 2-norm: $\|A\|_2 = (\rho(A^T A))^{1/2}$.
Here $\rho(A)$ stands for the spectral radius of a matrix A which is defined as $\rho(A) = \max |\lambda|$, where λ represents the eigenvalues of A .

Properties of matrix norms

Let $A = (a_{ij})_{n \times n}$, x and y be n -vectors.

- $\|A\| \geq 0$. $\|A\| = 0$ if and only if $A = 0$.
- $\|\alpha A\| = |\alpha| \|A\|$, $\alpha \in \mathbf{R}$.
- $\|A + B\| \leq \|A\| + \|B\|$.
- $\|Ax\| \leq \|A\| \|x\|$.
- $\|AB\| \leq \|A\| \|B\|$.

5 Condition numbers

Let A be an invertible $n \times n$ matrix, x^t be the true solution of

$$Ax = b, \quad b \neq 0,$$

and x^c be an approximate solution. Then

$$\frac{1}{\|A\| \|A^{-1}\|} \frac{\|r\|}{\|b\|} \leq \frac{\|x^t - x^c\|}{\|x^t\|} \leq \|A\| \|A^{-1}\| \frac{\|r\|}{\|b\|},$$

where $r = b - Ax^c$.

Brief proof: Make use of $A(x^t - x^c) = r$ and $Ax^t = b$ and their inverted forms as starting points, and apply the third matrix norm property to obtain upper and lower bounds for $\|x^t - x^c\|$ and $\|x^t\|$.

Define the *condition number* of an invertible matrix A to be

$$\text{Cond}(A) = \|A\| \|A^{-1}\|.$$

The numerical value of $\text{Cond}(A)$ depends on the matrix norm used. If $\text{Cond}(A)$ is large, then the upper bound on the relative error can be large even though $\|r\|$ is small. Hence when $\text{Cond}(A)$ is large, the linear system is said to be *ill-conditioned*. The matrix A is also said to be *ill-conditioned*.

It can be shown that $\text{Cond}(A) \geq 1$.

If $\text{Cond}(A)$ is of moderate size, the linear system is said to be *well-conditioned*.

The matrix A is also said to be *well-conditioned*.

6 Residual Correction Method

Let x^t be the true solution of

$$Ax = b$$

and $x^{(0)}$ be a computed solution.

Let

$$r^{(0)} = b - Ax^{(0)},$$

and

$$e^t = x^t - x^{(0)}.$$

It can be shown that

$$Ae^t = r^{(0)}$$

and e^t can be viewed as the true solution of $Ax = r^{(0)}$. If e^t can be obtained exactly, then we could obtain the true solution of $Ax = b$ through

$$x^t = x^{(0)} + e^t.$$

In practice, one can only obtain an approximation to e^t and hence an improved computed solution. The residual correction method is based on the above idea and iterates the process until a good enough computed solution is obtained.

Algorithm for Residual Correction Method:

Step 1 Compute $x^{(0)}$ an approximate solution of $Ax = b$ by a LU -decomposition method.

Step 2 for $i = 0, 1, \dots$,

 compute $r^{(i)} := b - Ax^{(i)}$;

 solve $Ae = r^{(i)}$ for $e^{(i)}$ using the matrix factors LU from step 1;

 compute $x^{(i+1)} := x^{(i)} + e^{(i)}$;

 stop the iteration if $\frac{\|x^{(i)} - x^{(i+1)}\|}{\|x^{(i+1)}\|} < \epsilon$ for some prescribed ϵ .

For $x^{(i+1)}$ to be closer to x^t than $x^{(i)}$, we must have $r^{(i)}$ computed in floating point arithmetic with doubled precision compared to other steps in the algorithm.

7 Iterative methods – an introduction

Iterative methods are usually used for solving large and sparse linear systems. consider the linear system $Ax = b$ and $A = N - P$. We rewrite the linear system as

$$Nx = Px + b.$$

The iterative methods start with an initial guess vector $x^{(0)}$ and compute a sequence of vectors $x^{(k)}$, $k = 1, 2, \dots$, using

$$Nx^{(k)} = Px^{(k-1)} + b, \quad k = 1, 2, \dots$$

Note here that we are solving a sequence of linear systems with the same coefficient matrix A . The matrix N should be chosen such that it is invertible and its associated linear systems easier to solve.

We want to know the condition required for $x^{(k)} \rightarrow x^t$.

Definition: The **spectral radius** of a $n \times n$ matrix A is defined to be

$$\rho(A) = \max_{i=1}^n |\lambda_i|$$

where $\lambda_1, \dots, \lambda_n$ are the n eigenvalues of A .

Theorem Let A be a square matrix. Then

$$\rho(A) \leq \|A\|$$

for any matrix norm $\|\cdot\|$.

Theorem: Let A be a square matrix. Then A^m converges to the zero matrix as $m \rightarrow \infty$ if and only if $\rho(A) < 1$.

Theorem: Let A be an invertible matrix, $A = N - P$, where N is invertible, and $M = N^{-1}P$. Let x^t be the true solution of $Ax = b$, $x^{(0)}$ a given starting vector, and

$$Nx^{(k)} = Px^{(k-1)} + b.$$

Then $\|x^{(k)} - x^t\| \rightarrow 0$ as $k \rightarrow \infty$ if and only if $\rho(M) < 1$.

8 Gauss-Jacobi method

Choose $N = \text{diag}(a_{11}, a_{22}, \dots, a_{nn}) = D$.

Then $P = -(A - D) = -(L + U)$, where L is the lower triangular part of A minus the diagonal and U is the upper triangular part of A minus the diagonal.

Therefore the Gauss-Jacobi method is

$$x^{(k)} = M_J x^{(k-1)} + D^{-1}b, \quad k = 1, 2, \dots,$$

where

$$M_J = -D^{-1}(L + U)$$

is called the Gauss-Jacobi matrix.

The more computationally efficient form is:

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right),$$

for $i = 1, 2, \dots, n$, and $k = 1, 2, \dots$.

Note that in solving the sequence of linear systems, there is no need for pivoting, inverting of matrices, LU factorizations, etc.

9 Gauss-Seidel Method

If the Gauss-Jacobi method converges, it is likely that $x_i^{(k)}$ is a closer approximation to the true value x_i^t than $x_i^{(k-1)}$. Thus writing the equation for the Gauss-Jacobi method in the form

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k-1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k-1)} \right),$$

and replacing $x_j^{(k-1)}$ in the right hand side by $x_j^{(k)}$, for $j = 1, \dots, i-1$, gives

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^n a_{ij} x_j^{(k-1)} \right),$$

which is the Gauss-Seidel method.

The matrix form of the Gauss-Seidel method is given by

$$x^{(k)} = M_S x^{(k-1)} + (D + L)^{-1} b,$$

where $M_S := -(D + L)^{-1} U$ is called the Gauss-Seidel matrix.

10 Convergence of Gauss-Jacobi and Gauss-Seidel methods

To ensure that any application of the Gauss-Jacobi or the Gauss-Seidel method will result in a convergence to the true solution, one has to have $\rho(M_J) < 1$ or $\rho(M_S) < 1$ respectively (by the theorem in the section 1.8).

For two special classes of matrices, one can show that these two methods will converge.

Definition: An $n \times n$ matrix $A = (a_{ij})$ is strictly diagonally dominant if

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|, \quad i = 1, 2, \dots, n.$$

Theorem Let A be a $n \times n$ strictly diagonally dominant matrix. The vector sequence $x^{(k)}$, $k = 1, 2, \dots$, generated by the Gauss-Jacobi method converges to the true solution of $Ax = b$ for any choice of starting vector $x^{(0)}$.

Theorem Let A be a positive definite matrix. The vector sequence $x^{(k)}$, $k = 1, 2, \dots$, generated by the Gauss-Seidel method converges to the true solution of $Ax = b$ for any starting vector $x^{(0)}$.

The proofs of both theorems above rely on a theorem in the section 1.8, i.e. showing $\rho(M_J) < 1$ and $\rho(M_S) < 1$ respectively.

The following is a result on the rate of convergence, c , of an iterative method.

Theorem Let $A = N - P$, N invertible, x^t be the true solution of $Ax = b$ and

$$Nx^{(k)} = Px^{(k-1)} + b, \quad k = 1, 2, \dots, \quad x^{(0)} \text{ given.}$$

Then the rate of convergence c is given by $c = \|N^{-1}P\|$ and

$$\|x^{(k)} - x^t\| \leq c^k \|x^{(0)} - x^t\|.$$

11 Successive Over-relaxation (SOR)

The SOR method is modified from the Gauss-Seidel method through introducing a relaxation parameter ω in the manner shown:

$$\begin{cases} z_i^{(k)} = \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^n a_{ij}x_j^{(k-1)}), \\ x_i^{(k)} = \omega z_i^{(k)} + (1 - \omega)x_i^{(k-1)}, \end{cases}$$

for $i = 1, 2, \dots, n$ and $k = 1, 2, \dots$.

The equivalent matrix form is

$$x^{(k)} = M(\omega)x^{(k-1)} + \omega(I + \omega D^{-1}L)^{-1}D^{-1}b,$$

where $M(\omega) := (I + \omega D^{-1}L)^{-1}((1 - \omega)I - \omega D^{-1}U)$ is the SOR matrix.

The SOR method is usually used in finite difference methods for solving partial differential equations with $1 < \omega < 2$ and it requires less number of iterations than the Gauss-Jacobi and Gauss-Seidel method for convergence for the same error tolerance.

It can be shown that $\rho(M(\omega))$ is minimized at the optimal ω^* with

$$\omega^* = \frac{2}{1 + (1 - \rho(M_J)^2)^{1/2}}.$$