

# Chapter Two: Numerical Methods for Elliptic PDEs

## 1 Finite Difference Methods for Elliptic PDEs

**1.1. Finite difference scheme.** We consider a simple example

$$-\Delta u := -\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = f(x, y) \quad (x, y) \in \Omega = [0, 1] \times [0, 1] \quad (1.1)$$

subject to Dirichlet boundary conditions

$$u|_{\partial\Omega} = g$$

Basic concept: replacing the derivatives by finite differences.

Let  $\pi = \{x_i, y_j\}_{i,j=0}^{M+1,N+1}$  define a mesh on  $\Omega$ . For simplicity, we consider a uniform mesh with the meshsize  $h_x = x_i - x_{i-1}$  and  $h_y = y_j - y_{j-1}$ , and  $u_{ij}$  be a mesh function on  $\pi$ .

The finite difference approximation is defined by

$$u_{xx} \approx \frac{u(x_{i-1}, y_j) - 2u(x_i, y_j) + u(x_{i+1}, y_j))}{h_x^2}$$

$$u_{yy} \approx \frac{u(x_i, y_{j-1}) - 2u(x_i, y_j) + u(x_i, y_{j+1}))}{h_y^2}$$

Then the finite difference solution  $u_{ij}$  is determined by the following system

$$L_h u_{ij} := \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h_x^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h_y^2} = f(x_i, y_j) \quad (1.2)$$

$$i = 1, 2, \dots, M; \quad j = 1, 2, \dots, N$$

$$u_{ij} = g(x_i, y_j) \quad i = 0, i = M + 1, j = 0, j = N + 1$$

In matrix form,

$$Au = f$$

where for  $h_x = h_y = h$

$$A = \frac{1}{h^2} \begin{bmatrix} B & -I & & & \\ -I & B & -I & & \\ & & \ddots & \ddots & \ddots \\ & & & -I & B \end{bmatrix} \quad B = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 4 \end{bmatrix}$$

and

$$u = (u_{11}, u_{12}, \dots, u_{1N}, u_{21}, \dots, u_{2N}, \dots, u_{M1}, \dots, u_{MN})^T.$$

In tensor product form,

$$(A_1 \otimes I_N + I_M \otimes A_2)u = f$$

where

$$A_1 = \frac{1}{h_x^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 2 \end{bmatrix}_{M \times M} \quad A_2 = \frac{1}{h_y^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 2 \end{bmatrix}_{N \times N}$$

Here the tensor product is defined by

$$B \otimes C = (b_{ij}C)$$

Some properties:

- (1)  $(A_1 \otimes A_2)(B_1 \otimes B_2) = (A_1 B_1) \otimes (A_2 B_2)$
- (2)  $(A_1 \otimes A_2)^{-1} = A_1^{-1} \otimes A_2^{-1}$
- (3) *if*  $Ax = \lambda_A x, \quad By = \lambda_B y$   
 $(A \otimes B)(x \otimes y) = \lambda_A \lambda_B (x \otimes y)$

## 1.2. Linear Solvers.

A major problem for elliptic PDEs is the efficient algorithms for solving the linear systems since the dimension of matrices for elliptic PDEs is much large than for two-point boundary value problems.

- Direct algorithm: Gaussian elimination.

Complexity:  $(NM)^3/3$  for two-dimensional elliptic PDEs. Memory:  $O(NM^2)$ .

- Iterative algorithms:

Jacobi, Gauss-Seidel, SOR, CG, Krylov subspace, ...

- Alternating direction implicit (ADI) algorithm

$$\begin{aligned} [(A_1 + \omega I) \otimes I] u^{n+1/2} &= -[I \otimes (A_2 - \omega I)] u^n + f \\ [I \otimes (A_2 + \omega I)] u^{n+1} &= -[(A_1 - \omega I) \otimes I] u^{n+1/2} + f \end{aligned}$$

Convergence: (i) ADI is convergence for any  $\omega > 0$ ; (ii) If  $\omega_n = \lambda_n$ , ADI is a finite method (direct) where  $\lambda_n$  is the eigenvalue of  $A_1$ .

Complexity: at each iteration, ADI algorithm needs to solve  $NM$  one-dimensional problems, *i.e.*, tridiagonal linear systems. The complexity at each step is  $(N^3)$  for  $N = M$ .

Remarks: ADI algorithm is simple, efficient and restricted on separable problems.

Convergence analysis:

First we consider the following basic iteration

$$u^{n+1} = T u^n + b$$

The iterative scheme is convergent if

$$\rho(T) < 1.$$

( $u$ :  $u = Tu + b$ , proof is easy).

ADI algorithm is a special case with

$$T = [I \otimes (A_2 + \omega I)]^{-1} [(A_1 - \omega I) \otimes I] [(A_1 + \omega I) \otimes I]^{-1} [I \otimes (A_2 - \omega I)]$$

It is easy to get

$$T = [(A_1 - \omega I)(A_1 + \omega I)^{-1}] \otimes [(A_1 + \omega I)^{-1}(A_2 - \omega I)]$$

and the eigenvalues of  $T$  are

$$\lambda(T) = \frac{(\lambda(A_1) - \omega)(\lambda(A_2) - \omega)}{(\lambda(A_1) + \omega)(\lambda(A_2) + \omega)} < 1$$

- Fast Poisson's solver.

We have proved that

$$A_1 F_s = \Lambda_1 F_s$$

where

$$F_s = \sqrt{\frac{2}{M+1}} \left( \sin \frac{ij\pi}{M+1} \right) \quad \lambda_i = 4(M+1)^2 \sin^2 \frac{i\pi}{2(M+1)}$$

and  $F_s^2 = I$ . Then the finite difference system can be rewritten by

$$(F_s \otimes I)(\Lambda_1 \otimes I + I \otimes A_2)(F_s \otimes I)u = f$$

and moreover,

$$(\Lambda_1 \otimes I + I \otimes A_2)\bar{u} = (F_s \otimes I)f$$

where

$$\bar{u} = (F_s \otimes I)u$$

Fast Poisson solver:

- Calculate  $\bar{f} = (F_s \otimes I)f$ ;
- Solve the system

$$(\Lambda_1 \otimes I + I \otimes A_2)\bar{u} = \bar{f}$$

- calculate  $u = (F_s \otimes I)\bar{u}$ .

Complexity:  $O(N^2 \log N)$  for  $N = M$ .

### 1.3. General second-order Elliptic PDEs.

- General domain
- General boundary conditions
- Problems in three-dimensional spaces
- More general elliptic PDEs

$$a_1 u_{xx} + a_2 u_{xy} + a_3 u_{yy} + b_1 u_x + b_2 u_y + cu = f$$

- Linear elasticity equations

$$\begin{aligned} (\lambda + 2\mu)u_{xx} + \mu u_{yy} + (\lambda + \mu)v_{xy} &= f_1 \\ (\lambda + 2\mu)v_{yy} + \mu v_{xx} + (\lambda + \mu)u_{xy} &= f_2 \end{aligned}$$

where  $(u, v)$  define the displacement.

## 2 FEM for two-dimensional elliptic PDEs

### 2.1 Basic idea

We consider a simple model problem, Poisson equation

$$\begin{aligned} -\Delta u &= f(x, y), & (x, y) \in \Omega \\ u|_{\partial\Omega} &= 0 \end{aligned} \tag{2.1}$$

where  $\partial\Omega$  denotes the boundary of  $\Omega$  and

$$\Delta u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}.$$

Similarly to one-dimensional problem, we need a variational model which will be obtained by using Green formula instead of integration by part in the last section. Let  $u = u(x, y)$  and  $w = w(x, y)$  be smooth functions in  $\Omega$ . Green formula is given by

$$-\int_{\Omega} v \Delta w \, d\Omega = -\int_{\partial\Omega} v w_n \, ds + \int_{\Omega} \nabla w \cdot \nabla v \, d\Omega$$

*i.e.*,

$$-\int_{\Omega} v \left( \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \right) \, dx dy = -\int_{\partial\Omega} v \frac{\partial w}{\partial n} \, ds + \int_{\Omega} \left( \frac{\partial v}{\partial x} \frac{\partial w}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial w}{\partial y} \right) \, dx dy$$

where

$$\begin{aligned} \frac{\partial w}{\partial n} &= \frac{\partial w}{\partial x} n_x + \frac{\partial w}{\partial y} n_y \\ n_x &= \cos \langle \vec{n}, \vec{x} \rangle, & n_y &= \cos \langle \vec{n}, \vec{y} \rangle, \\ \nabla v &= \begin{pmatrix} \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{pmatrix} & \nabla w &= \begin{pmatrix} \frac{\partial w}{\partial x} \\ \frac{\partial w}{\partial y} \end{pmatrix}. \end{aligned}$$

By using the Green formula for the model problem (2.1) and letting  $v \in H_0^1(\Omega)$ , we have

$$-\int_{\Omega} v \Delta w \, d\Omega = \int_{\Omega} \nabla u \cdot \nabla v \, d\Omega = \int_{\partial\Omega} f v \, d\Omega.$$

Let

$$\begin{aligned} a(u, v) &= \int_{\Omega} \nabla u \cdot \nabla v \, d\Omega \\ (f, v) &= \int_{\partial\Omega} f v \, d\Omega \\ F(v) &= \frac{1}{2} a(v, v) - (f, v). \end{aligned}$$

We can obtain the equivalent (V) model and (M) model

$$\begin{aligned} \text{Find } u \in H_0^1(\Omega) \text{ such that for all } v \in H_0^1(\Omega), \\ a(u, v) &= (f, v) \end{aligned} \tag{V}$$

and

$$\min_{v \in H_0^1(\Omega)} F(v) \tag{M}$$

Similarly, the above models are to find a solution in a infinite dimensional space. In FEM, we need to approximate the infinite dimensional space by finite dimensional spaces  $V_h$ . Let

$\phi_j(x, y)$ ,  $j = 1, 2, \dots, N$ , be the basis functions of  $V_h$ . A function  $u_h \in V_h$  can be expressed by

$$u_h(x, y) = \sum_{j=1}^N \alpha_j \phi_j(x, y).$$

Letting  $v = \phi_i$ ,  $i = 1, 2, \dots, N$ , respectively, in the  $(V_h)$ -model, we have

$$a(u_h, \phi_j) = (f, \phi_j)$$

and

$$\sum_{j=1}^N \alpha_j \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, d\Omega = \int_{\Omega} f \phi_i \, d\Omega, \quad i = 1, 2, \dots, N.$$

The linear system is given by

$$A\boldsymbol{\alpha} = \mathbf{b}$$

where

$$a_{ij} = \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, d\Omega, \quad b_i = \int_{\Omega} f \phi_i \, d\Omega.$$

The importance is the choice of finite dimensional space. Piecewise polynomial space is one of most popular choices. In this case, one has to construct a mesh, *i.e.*, divided the domain  $\Omega$  into small pieces (element). Mesh generation in multi-dimensional space is one of important parts in study of FEM.

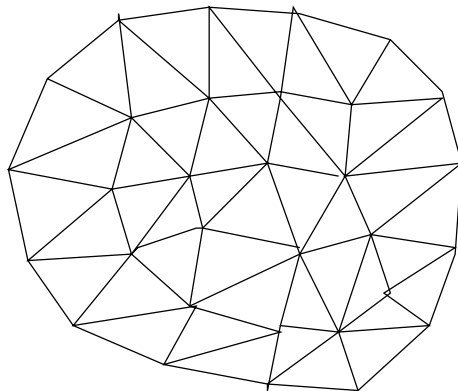


Figure 4: Finite element mesh in two-dimensional space.

## 2.2 Basis functions

Here we consider piecewise polynomial basis functions. Let  $\Omega$  be a bounded domain which consists of the elements  $\{K_i\}_{i=1}^M$  and nodal points  $\{(x_j, y_j)\}_{j=1}^N$ . We first consider the piecewise linear space where

$$V_h = \{v : \text{continuous on } \Omega, \text{ piecewise linear, } v = 0 \text{ on } \partial\Omega\}.$$

The corresponding basis functions  $\phi_j(\mathbf{x}) \in V_h$ ,  $j = 1, 2, \dots, N$ , satisfy

\*  $\phi_j(\mathbf{x})$  is piecewise linear; \*  $\phi_j(\mathbf{x}_i) = \delta_{ij}$  where  $\mathbf{x} = (x, y)$ .

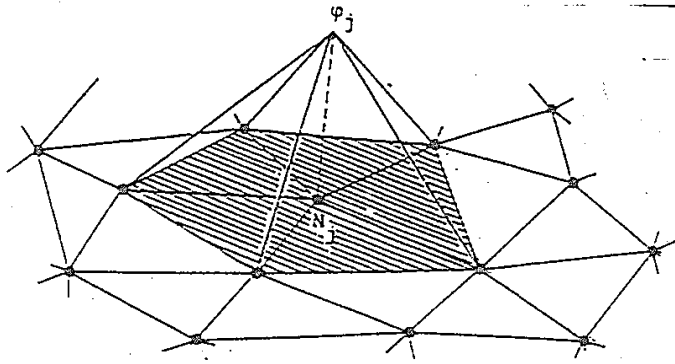


Figure 5: The basis function of linear element in two-dimensional space.

We see that the support of  $\phi_j$  (the set of points  $\mathbf{x}$  for which  $\phi_j \neq 0$ ) consists of the triangles with the common node  $\mathbf{x}_j$  (shaded area).

Now we need to formulate the piecewise linear function.

*Example.* Find a linear function  $L(x, y)$  on  $\Omega_1 = \{(x, y) | 0 \leq x \leq a, 0 \leq y \leq b, x/a + y/b \leq 1\}$  satisfies

- \* linear;
- \*  $L(a, 0) = 0, L(0, b) = 0$ ;
- \*  $L(0, 0) = 1$ .

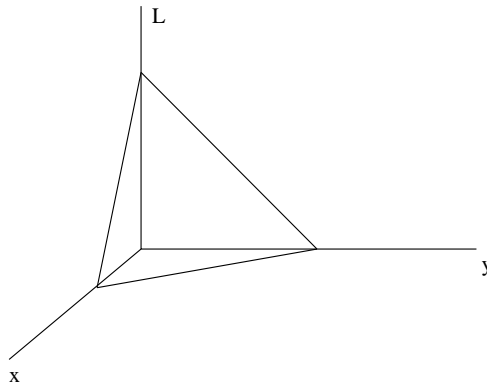


Figure 6: The function  $L(x, y)$ .

Since  $L(x, y)$  is a linear function, it can be expressed by  $L(x, y) = b_0 + b_1x + b_2y$ . Then the solution of problem is unique. The equation of line passed through the points  $(a, 0)$  and  $(0, b)$  is

$$\frac{x}{a} + \frac{y}{b} = 1.$$

Let

$$l(x, y) = \frac{x}{a} + \frac{y}{b} - 1.$$

Then the solution is

$$L(x, y) = \frac{l(x, y)}{l(0, 0)} = 1 - \frac{x}{a} - \frac{y}{b}$$

which satisfies the all above conditions.

For a given triangle  $\Omega_1$  with the three vertices  $\mathbf{x}_1$ ,  $\mathbf{x}_2$  and  $\mathbf{x}_3$ . We need to find three linear functions  $L_j(\mathbf{x})$ ,  $j = 1, 2, 3$ , satisfying  $L_j(\mathbf{x}_i) = \delta_{ij}$ . Let  $l_{ij}(\mathbf{x})$  denote the equation of line passed through the vertices  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . Then

$$L_1(\mathbf{x}) = \frac{l_{23}(\mathbf{x})}{l_{23}(\mathbf{x}_1)} \quad L_2(\mathbf{x}) = \frac{l_{31}(\mathbf{x})}{l_{31}(\mathbf{x}_2)} \quad L_3(\mathbf{x}) = \frac{l_{12}(\mathbf{x})}{l_{12}(\mathbf{x}_3)}.$$

Now we can formulate the linear basis functions, each of them corresponds to each interior node.  $\phi_j(\mathbf{x})$  is nonzero only in the elements around the vertex  $\mathbf{x}_j$ . The formula in such each element can be obtained as above.

*Example (Quadratic element).* Find a function  $Q_j(x, y)$  on a triangle  $\Omega_1$  with its vertices  $\mathbf{x}_i$ ,  $i = 1, 2, 3$ , satisfies

\* quadratic;

\*  $Q_j(\mathbf{x}_i) = \delta_{ij}$ ,  $i, j = 1, 2, 3$ .

It is obvious that the solution is not unique. We need to add some more conditions. Here we consider a triangular element below.

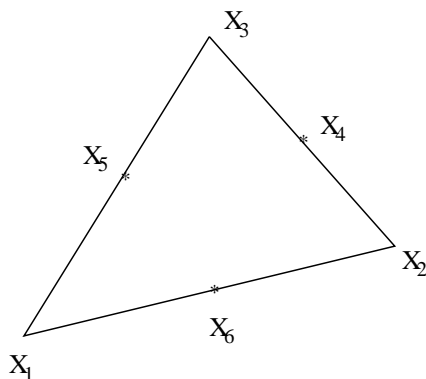


Figure 7: The basis function of quadratic element.

We require the second condition to be satisfied for  $j = 1, 2, 3, 4, 5, 6$ . Let  $l_{ij}(\mathbf{x})$  be the equation of line passed through  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . Then the solution is

$$\begin{aligned} Q_1(\mathbf{x}) &= \frac{l_{56}(\mathbf{x}) \cdot l_{23}(\mathbf{x})}{l_{56}(\mathbf{x}_1) \cdot l_{23}(\mathbf{x}_1)} & Q_2(\mathbf{x}) &= \frac{l_{46}(\mathbf{x}) \cdot l_{13}(\mathbf{x})}{l_{46}(\mathbf{x}_2) \cdot l_{13}(\mathbf{x}_2)} \\ Q_3(\mathbf{x}) &= \frac{l_{45}(\mathbf{x}) \cdot l_{12}(\mathbf{x})}{l_{45}(\mathbf{x}_3) \cdot l_{12}(\mathbf{x}_3)} & Q_4(\mathbf{x}) &= \frac{l_{12}(\mathbf{x}) \cdot l_{13}(\mathbf{x})}{l_{12}(\mathbf{x}_4) \cdot l_{13}(\mathbf{x}_4)} \\ Q_5(\mathbf{x}) &= \frac{l_{21}(\mathbf{x}) \cdot l_{23}(\mathbf{x})}{l_{21}(\mathbf{x}_5) \cdot l_{23}(\mathbf{x}_5)} & Q_6(\mathbf{x}) &= \frac{l_{31}(\mathbf{x}) \cdot l_{32}(\mathbf{x})}{l_{31}(\mathbf{x}_6) \cdot l_{32}(\mathbf{x}_6)}. \end{aligned}$$

It is noted that  $Q_j(x, y)$  defined above is quadratic for each component.

*Example (Quadrilateral element.* Four-point quadrilateral element)

The linear basis functions  $L_j(\mathbf{x})$  can be obtained analogously.

$$\begin{aligned} L_1(\mathbf{x}) &= \frac{l_{23}(\mathbf{x}) \cdot l_{34}(\mathbf{x})}{l_{23}(\mathbf{x}_1) \cdot l_{34}(\mathbf{x}_1)} & L_2(\mathbf{x}) &= \frac{l_{14}(\mathbf{x}) \cdot l_{34}(\mathbf{x})}{l_{14}(\mathbf{x}_2) \cdot l_{34}(\mathbf{x}_2)} \\ L_3(\mathbf{x}) &= \frac{l_{12}(\mathbf{x}) \cdot l_{14}(\mathbf{x})}{l_{12}(\mathbf{x}_3) \cdot l_{14}(\mathbf{x}_3)} & L_4(\mathbf{x}) &= \frac{l_{12}(\mathbf{x}) \cdot l_{23}(\mathbf{x})}{l_{12}(\mathbf{x}_4) \cdot l_{23}(\mathbf{x}_4)} \end{aligned}$$

which is not exact "linear" function in general.

Some other high-order elements are as follows.

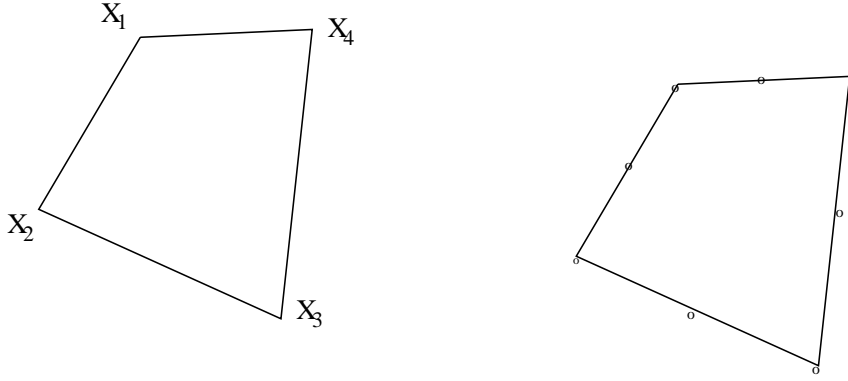


Figure 8: Some other elements.

### Stiffness matrix

For the simple model problem (2.1), we have

$$a_{ij} = \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, d\Omega.$$

Here we only consider the linear element.

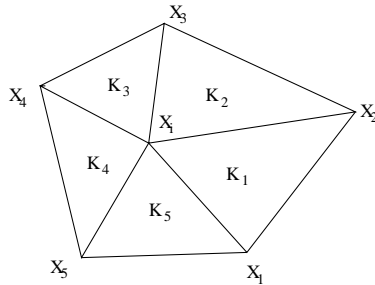


Figure 9: The elements around the node  $X_i$

Let  $K_i$  denote the five elements around the node  $\mathbf{x}_j$ . Then

$$\phi_j(x) = \frac{l_{12}(x)}{l_{12}(\mathbf{x}_j)}.$$

The equation of line passed through  $\mathbf{x}_1$  and  $\mathbf{x}_2$  is

$$\frac{x - x_1}{x_2 - x_1} - \frac{y - y_1}{y_2 - y_1} = 0$$

i.e.,

$$l_{12}(x, y) := (x_1 y_2 - x_2 y_1) - x(y_2 - y_1) + y(x_2 - x_1) = 0.$$

We see that

$$l_{12}(x, y) = \begin{vmatrix} 1 & 1 & 1 \\ x & x_1 & x_2 \\ y & y_1 & y_2 \end{vmatrix}$$

and

$$l_{12}(x_j, y_j) = \begin{vmatrix} 1 & 1 & 1 \\ x_j & x_1 & x_2 \\ y_j & y_1 & y_2 \end{vmatrix} = 2A(K_1) = (2\Delta_1)$$

where  $A(K_1) = \Delta_1$  is the area of the element  $K_1$ . Since

$$\frac{\partial \phi_j}{\partial x} = -\frac{y_2 - y_1}{2\Delta_1} \quad \frac{\partial \phi_j}{\partial y} = \frac{x_2 - x_1}{2\Delta_1},$$

we have

$$a_{ij} = \sum_l \int_{K_l} \nabla \phi_i \cdot \nabla \phi_j d\Omega = \sum_l \int_{K_l} \left( \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} \right) + \left( \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) d\Omega.$$

Obviously, if no segment between  $\mathbf{x}_i$  and  $\mathbf{x}_j$ ,  $a_{ij} = 0$ .

*Example.* We consider the following Poisson's equation

$$\begin{aligned} -\Delta u &= 1 & (x, y) \in \Omega &= [0, 1] \times [0, 1] \\ u|_{\partial\Omega} &= 0 \end{aligned}$$

By using linear triangular element and  $h = 1/3$ .

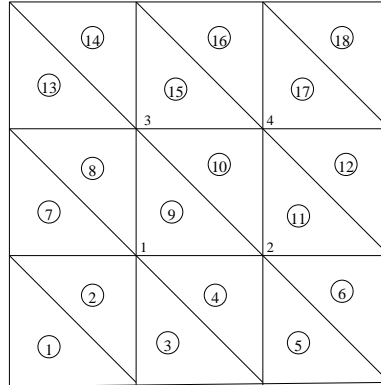


Figure 10: An example.

Node  $i = 1$ ,

$$\begin{aligned} a_{11} &= \int_{K_2+K_3+K_4+K_7+K_8+K_9} \nabla \phi_1 \cdot \nabla \phi_1 dx dy \\ &= \int_{K_2} + \int_{K_3} + \int_{K_4} + \int_{K_7} + \int_{K_8} + \int_{K_9} . \end{aligned}$$

We have calculate the integrals one by one. Since on  $K_2$ ,  $\phi_1 = (x + y - h)/h$ ,

$$\int_{K_2} \nabla \phi_1 \cdot \nabla \phi_1 \, dxdy = \int_{K_2} 2 \left(\frac{1}{h}\right)^2 \, dxdy = 1.$$

Also we have

$$\int_{K_3} \nabla \phi_1 \cdot \nabla \phi_1 \, dxdy = \int_{K_3} \left(\frac{1}{h}\right)^2 \, dxdy = 1/2$$

and

$$\int_{K_4} = \int_{K_7} = \int_{K_8} = 1/2 \quad \int_{K_9} = 1.$$

Then

$$a_{11} = 4.$$

For  $a_{12}$ , we have

$$a_{12} = \int_{K_4+K_9} \nabla \phi_1 \cdot \nabla \phi_1 \, dxdy.$$

We can obtain

$$\int_{K_4} \nabla \phi_1 \cdot \nabla \phi_1 \, dxdy = \int_{K_9} \nabla \phi_1 \cdot \nabla \phi_1 \, dxdy = -1/2$$

and therefore,

$$a_{12} = -1.$$

Similarly,

$$a_{13} = -1 \quad a_{14} = 0 \quad (\text{no edge between } \mathbf{x}_1 \text{ and } \mathbf{x}_4)$$

and

$$\begin{aligned} a_{22} &= a_{33} = a_{44} = 4 \\ a_{23} &= 0 \quad a_{24} = a_{34} = 1. \end{aligned}$$

Then the stiffness matrix is

$$A = \frac{1}{h} \begin{bmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{bmatrix}$$

The calculation of the load vector  $b$  is more complicated.

$$\begin{aligned} b_1 &= \int_{K_2+K_3+K_4+K_7+K_8+K_9} \phi_1 \, dxdy \\ b_2 &= \int_{K_4+K_5+K_6+K_9+K_{10}+K_{11}} \phi_2 \, dxdy \\ b_3 &= \int_{K_8+K_9+K_{10}+K_{13}+K_{14}+K_{15}} \phi_3 \, dxdy \\ b_4 &= \int_{K_{10}+K_{11}+K_{12}+K_{15}+K_{16}+K_{17}} \phi_4 \, dxdy. \end{aligned}$$

The finite element linear system is

$$A\boldsymbol{\alpha} = \mathbf{b}.$$

If we consider the case with  $N \times N$  mesh. Then the stiffness matrix is block tridiagonal

$$A = \begin{bmatrix} B & -I & & & \\ -I & B & -I & & \\ & \ddots & \ddots & \ddots & \\ & & & -I & B \end{bmatrix}$$

where

$$B = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 4 \end{bmatrix}.$$

If we have used different partition, or different index, we will obtain different stiffness matrices. It will be very complicated for general geometry, equations and boundary conditions.

### 2.3 Element stiffness matrix.

The above approach is very complicated to implement in computer. As we discussed in the section one, we consider an alternative approach, element stiffness matrix.

We assume that we have had a triangular mesh on the domain  $\Omega$ , which consists of the elements  $K_k$ ,  $k = 1, 2, \dots, M$  and  $\phi_j$ ,  $j = 1, 2, \dots, N$ , be the linear basis functions.

For the element  $K_k$ , we denote its element stiffness matrix by

$$A_k = (a_{ij}^{(k)})_{3 \times 3}$$

where for the simple model (2.1),

$$a_{ij}^{(k)} = \bar{a}(\phi_i, \phi_j) = \int_{K_k} \nabla \phi_i \cdot \nabla \phi_j \, dx dy.$$

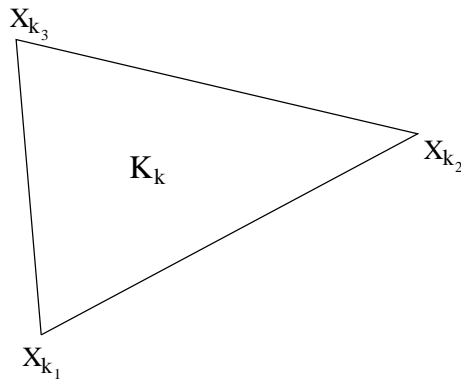


Figure 11: A typical element.

Let  $(\bar{x}_i, \bar{y}_i)$ ,  $i = 1, 2, 3$ , be the coordinates of three vertices of  $K_k$ . Then

$$a_{ij}^{(k)} = \int_{K_k} \left( \frac{\partial \phi_{k_i}}{\partial x} \frac{\partial \phi_{k_j}}{\partial x} + \frac{\partial \phi_{k_i}}{\partial y} \frac{\partial \phi_{k_j}}{\partial y} \right) dx dy \quad i, j = 1, 2, 3$$

and

$$\phi_{k_1} = \frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 & 1 \\ x & \bar{x}_2 & \bar{x}_3 \\ y & \bar{y}_2 & \bar{y}_3 \end{vmatrix} \quad \phi_{k_2} = \frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 & 1 \\ x & \bar{x}_3 & \bar{x}_1 \\ y & \bar{y}_3 & \bar{y}_1 \end{vmatrix} \quad \phi_{k_3} = \frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 & 1 \\ x & \bar{x}_1 & \bar{x}_2 \\ y & \bar{y}_1 & \bar{y}_2 \end{vmatrix}.$$

In general,

$$\phi_{k_i} = \frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 & 1 \\ x & \bar{x}_{i+1} & \bar{x}_{i+2} \\ y & \bar{y}_{i+1} & \bar{y}_{i+2} \end{vmatrix}$$

where  $\mathbf{x}_4 = \mathbf{x}_1$  and  $\mathbf{x}_5 = \mathbf{x}_2$ . Hence,

$$\frac{\partial \phi_{k_i}}{\partial x} = -\frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 \\ \bar{y}_{i+1} & \bar{y}_{i+2} \end{vmatrix} \quad \frac{\partial \phi_{k_i}}{\partial y} = \frac{1}{2\Delta_k} \begin{vmatrix} 1 & 1 \\ \bar{x}_{i+1} & \bar{x}_{i+2} \end{vmatrix}.$$

We have the following formulas:

$$a_{ij}^{(k)} = \frac{1}{4\Delta_k} [(\bar{y}_{i+2} - \bar{y}_{i+1})(\bar{y}_{j+2} - \bar{y}_{j+1}) + (\bar{x}_{i+2} - \bar{x}_{i+1})(\bar{x}_{j+2} - \bar{x}_{j+1})] \quad (2.2)$$

and

$$b_i^{(k)} = \int_{K_k} f(x, y) \phi_i dx dy.$$

Since  $f(x, y)$  could be an arbitrary function, we need to use Gauss-Quadrature technique to calculate the above integrals

$$b_i^{(k)} = \sum_{l=1}^p w_l f(x_l^*, y_l^*) \phi_i(x_l^*, y_l^*)$$

where  $(x_l^*, y_l^*)$  and  $w_l$  define the Gaussian points and the corresponding weight.

*Remarks.* For some complicated elliptic PDEs,  $a(u, v)$  could be complicated and we cannot obtain the above general formula. We have to use Gauss-Quadrature to calculate both  $a_{ij}^{(k)}$  and  $b_i^{(k)}$ .

In most computer program, the stiffness matrix is obtained by using the following approach. Let  $A = (a_{ij})$  and  $A_k = a_{ij}^{(k)}$  be the (global) stiffness matrix and element stiffness matrix, respectively. Then for each element  $K_k$ , we need the following loop

$$a_{j_p, j_q} = a_{j_p, j_q} + a_{pq}^{(k)} \quad p, q = 1, 2, 3 \quad (2.3)$$

and

$$b_{j_p} = b_{j_p} + b_p^{(k)} \quad p, q = 1, 2, 3.$$

The relationship between  $(j_p, j_q)$  and  $(p, q)$  will be discussed later. Finally we use the boundary conditions to reduce the size of the matrix.

*Example.* We consider the same problem as in the last example. Here  $\Delta_k = 1/18$ .

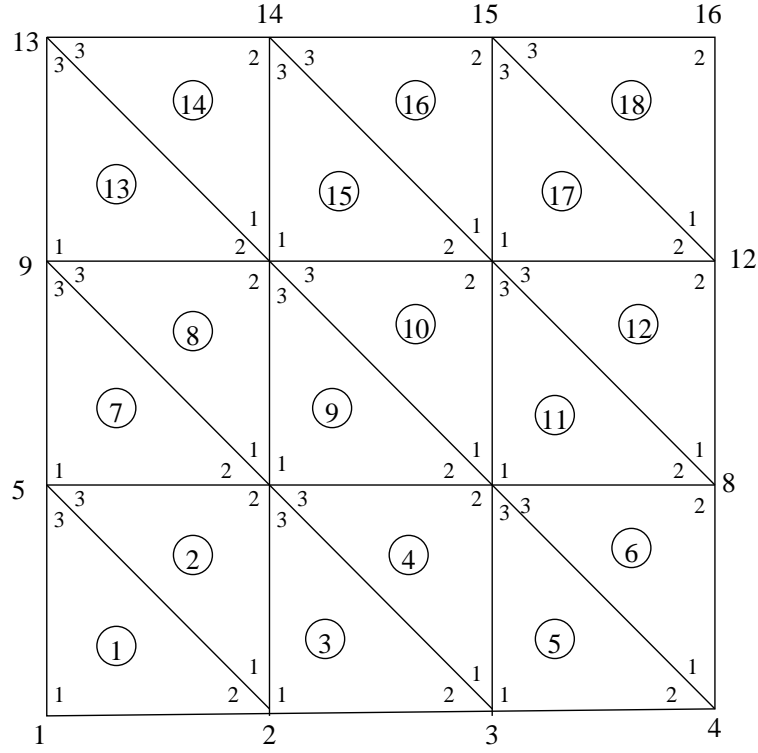


Figure 12: An example.

(i) Element stiffness matrices. By using the above formulas,

$$\begin{aligned}
 a_{11}^{(1)} &= \frac{1}{4\Delta_1} \left[ \left(\frac{1}{3}\right) \left(\frac{1}{3}\right) + \left(\frac{-1}{3}\right) \left(\frac{-1}{3}\right) \right] = 1 \\
 a_{12}^{(1)} &= \frac{1}{4\Delta_1} \left[ \left(\frac{1}{3}\right) \left(\frac{-1}{3}\right) + \left(\frac{-1}{3}\right) \cdot 0 \right] = -1/2 \\
 a_{13}^{(1)} &= \frac{1}{4\Delta_1} \left[ \left(\frac{1}{3}\right) \cdot 0 + \left(\frac{-1}{3}\right) \left(\frac{1}{3}\right) \right] = -1/2 \\
 a_{22}^{(1)} &= \frac{1}{4\Delta_1} \left[ \left(\frac{-1}{3}\right) \left(\frac{-1}{3}\right) + 0 \right] = 1/2 \\
 a_{11}^{(2)} &= \frac{1}{4\Delta_1} \left[ \left(\frac{-1}{3}\right) \cdot 0 + 0 \cdot \left(\frac{1}{3}\right) \right] = 0 \\
 a_{33}^{(1)} &= \frac{1}{4\Delta_1} \left[ 0 + \left(\frac{1}{3}\right) \left(\frac{1}{3}\right) \right] = 1/2.
 \end{aligned}$$

Then we have

$$A_1 = \begin{bmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1/2 & 0 \\ -1/2 & 0 & 1/2 \end{bmatrix}$$

and similarly

$$A_2 = \begin{bmatrix} 1/2 & -1/2 & 0 \\ -1/2 & 1 & -1/2 \\ 0 & -1/2 & 1/2 \end{bmatrix}.$$

Moreover, we have

$$A_1 = A_3 = A_5 \cdots = A_{15} = A_{17} \quad A_2 = A_4 = \cdots = A_{16} = A_{18}.$$



The finite element solution can be obtained by solving the above system.

#### **2.4 Implementation Computer program structure.**

(i) Data input

\* functions (such as  $f(x, y)$ ) and parameters in the PDE;

\* boundary conditions;

\* Geometry  $\Omega$ .

(ii) Mesh generation

\* generate the coordinates of all nodal points;

\* generate the connection among the local indexes (of points) and global indexes (of points).

(iii) Stiffness matrix and load vector

\* generate element stiffness matrices;

\* generate element load vector;

\* Gaussian quadrature;

\* assembling.

(iv) Linear solver (iterative methods or direct methods)

(v) Post-process.

#### **Gauss quadrature**

We consider the integral

$$\int_0^1 f(x) dx \quad \text{or} \quad \int_{-1}^1 f(x) dx$$

In general,

$$\int \int_{\Delta} f(x, y) dx dy$$

Usually we use the same approximation in FEM to approximate the function  $f(x, y)$ . For example, if we use linear FEM method, then

$$\int \int_{\Delta} f(x, y) dx dy \approx \int \int_{\Delta} f_L(x, y) dx dy$$

where  $f_L(x, y)$  is a linear interpolation polynomial on the element  $\Delta$ .

#### **Linear solver**

# Chapter Three Numerical Solution for Parabolic PDEs

## 1 FD for parabolic PDEs.

Here we consider numerical solutions of a simple model problem

$$\begin{aligned}u_t &= u_{xx}, & x \in (0, 1), t > 0 \\u(0, x) &= u_0(x), & x \in (0, 1) \\u(t, 0) &= u(t, 1) = 0, & t > 0\end{aligned}\tag{1.1}$$

where  $u_0(x)$  is given.

Let  $\{t_n, x_j\}$  define a partition, where

$$\begin{aligned}0 &= t_0 < t_1 < t_2 \dots < t_n < t_{n+1} \\0 &= x_0 < x_1 < x_2 \dots < x_N < x_{N+1} = 1.\end{aligned}$$

For simplicity, we assume the mesh is uniform, *i.e.*,

$$h = x_{j+1} - x_j \quad \tau = t_{n+1} - t_n.$$

Let  $u_j^n$  be a mesh function defined on the mesh  $\{t_n, x_j\}$ . Similarly, we use the following approximations

$$\begin{aligned}u_{xx}(t_n, x_j) &\approx \frac{u(t_n, x_{j-1}) - 2u(t_n, x_j) + u(t_n, x_{j+1}))}{h^2} \\u_t(t_n, x_j) &\approx \frac{u(t_{n+1}, x_j) - u(t_n, x_j)}{\tau}\end{aligned}\tag{1.2}$$

for the parabolic equation. The FD solution satisfies the following system

$$\begin{aligned}\frac{u_j^{n+1} - u_j^n}{\tau} &= \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{h^2} & j = 1, 2, \dots, N; n = 0, 1, \dots \\u_j^0 &= u_0(x_j) \\u_0^n &= u_{N+1}^n = 0.\end{aligned}\tag{1.3}$$

In matrix form, we have

$$\begin{aligned}U^{n+1} &= \left(I - \frac{\tau}{h^2} A_1\right) U^n \\U^0 &= U_0\end{aligned}\tag{1.4}$$

where

$$A_1 = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \quad U^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_N^n \end{pmatrix} \quad U_0 = \begin{pmatrix} u_0(x_1) \\ u_0(x_2) \\ \vdots \\ u_0(x_N) \end{pmatrix}$$

which is called the explicit (forward) Euler FD scheme.

Similarly, if we use the approximation

$$u_t(t_n, x_j) \approx \frac{u(t_n, x_j) - u(t_{n-1}, x_j)}{\tau}.$$

Then we obtain the implicit (backward) Euler FD scheme

$$\begin{aligned} \frac{u_j^n - u_j^{n-1}}{\tau} &= \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{h^2} \quad j = 1, 2, \dots, N; n = 1, 2, \dots \\ u_j^0 &= u_0(x_j) \\ u_0^n &= u_{N+1}^n = 0 \end{aligned} \quad (1.5)$$

and in matrix form,

$$\begin{aligned} (I + \frac{\tau}{h^2} A_1) U^n &= U^{n-1} \quad n = 1, 2, \dots \\ U^0 &= U_0 \end{aligned} \quad (1.6)$$

In terms of Taylor's expansion, we can prove that the truncation error for both explicit and implicit Euler scheme is  $O(\tau + h^2)$ .

In order to get a scheme with second-order accuracy in  $t$  direction, we consider the approximation

$$\begin{aligned} u_t(t_n + \tau/2, x_j) &\approx \frac{u(t_{n+1}, x_j) - u(t_n, x_j)}{\tau} \\ u_{xx}(t_n + \tau/2, x_j) &\approx \frac{1}{2} \frac{u(t_{n+1}, x_{j-1}) - 2u(t_{n+1}, x_j) + u(t_{n+1}, x_{j+1}))}{h^2} \\ &\quad + \frac{1}{2} \frac{u(t_n, x_{j-1}) - 2u(t_n, x_j) + u(t_n, x_{j+1}))}{h^2}. \end{aligned} \quad (1.7)$$

The corresponding scheme can be written by

$$\begin{aligned} (I + \frac{1}{2} \frac{\tau}{h^2} A_1) U^{n+1} &= (I - \frac{1}{2} \frac{\tau}{h^2} A_1) U^n \quad n = 0, 1, 2, \dots \\ U^0 &= U_0 \end{aligned} \quad (1.8)$$

This is so-called the Crank-Nicolson FD scheme. The truncation error is  $O(\tau^2 + h^2)$ .

## 2 FE for parabolic PDEs

To formulate a finite element method, we need a variational model for the parabolic PDE.

Let  $v \in H_0^1$  and

$$\begin{aligned} (u_t, v) &= (u_{xx}, v) \\ (u(0, x), v) &= (u_0, v) \end{aligned}$$

Using integration by part,

$$\begin{aligned} (u_t, v) + a(u, v) &= 0 \\ (u(0, x), v) &= (u_0, v) \end{aligned} \quad (2.1)$$

where  $a(u, v) = (u_x, v_x)$ .

The corresponding variational model is: Find  $u \in H_0^1$  such that for all  $v \in H_0^1$ , the above equation holds. Theoretically, we can prove the equivalence of the parabolic PDE and the variational model.

The variational model for FEM is to find  $u \in V_N$  such that for all  $v \in V_N$ , the equation (2.1) holds.

Now we consider the linear FEM. Let  $\{\phi_j\}_{j=0}^N$  be the basis functions of linear FEM. Then

$$u(t, x) = \sum \alpha_j(t) \phi_j(x), \quad u_t(t, x) = \sum \alpha_j'(t) \phi_j(x).$$

Let  $v = \phi_i(x)$ ,  $i = 1, 2, \dots, N$ ,

$$\begin{aligned} (\sum \alpha_j'(t) \phi_j, \phi_i) + a(\sum \alpha_j(t) \phi_j, \phi_i) &= 0 \\ (\sum \alpha_j(0) \phi_j, \phi_i) &= (u_0, \phi_i) \end{aligned}$$

equivalently

$$\begin{aligned} \sum \alpha_j'(t) (\phi_j, \phi_i) + \sum \alpha_j(t) a(\phi_j, \phi_i) &= 0 \\ \sum \alpha_j(0) (\phi_j, \phi_i) &= (u_0, \phi_i). \end{aligned}$$

In matrix form,

$$\begin{aligned} C\boldsymbol{\alpha}' + A\boldsymbol{\alpha} &= 0 \\ C\boldsymbol{\alpha}(0) &= b \end{aligned} \quad (2.2)$$

where

$$a_{ij} = a(\phi_j, \phi_i), \quad c_{ij} = (\phi_j, \phi_i) \quad b_i = (u_0, \phi_i)$$

or

$$A = \frac{1}{h} A_1, \quad C = \text{tridiag}(1/6, 2/3, 1/6),$$

$$\boldsymbol{\alpha} = (\alpha_1(t), \alpha_2(t), \dots, \alpha_N(t))^T \quad \boldsymbol{\alpha}' = (\alpha_1'(t), \alpha_2'(t), \dots, \alpha_N'(t))^T.$$

(2.2) is a system of ODEs. In fact, in terms of FEM approximation in  $x$  direction, we reduce the parabolic PDE to a system of ODEs. One can use classical ODE solvers for solving the resulting ODE system. Thus such a scheme is called the semi-discrete method.

There are many different fully-discrete schemes. Here we only consider some FD schemes for  $t$  direction. If we use the approximation in (1.3), the variational model is to find  $u \in V_N$  such that for all  $v \in V_N$ ,

$$\begin{aligned} \left( \frac{u^{n+1}(x) - u^n(x)}{\tau}, v \right) + a(u^n(x), v) &= 0 \\ (u^0(x), v) &= (u_0, v) \end{aligned} \quad (2.3)$$

The variational model for FEM is to find  $u \in V_n$  such that for all  $v \in V_N$ , the equation (2.3) holds.

Similarly, we consider the linear FEM. Let  $\{\phi_j\}_{j=0}^N$  be the basis functions of linear FEM. Then

$$u^n(x) = \sum \alpha_j^n \phi_j(x), .$$

Let  $v = \phi_i(x)$ ,  $i = 1, 2, \dots, N$ ,

$$\begin{aligned} \sum (\alpha_j^{n+1} - \alpha_j^n) (\phi_j, \phi_i) + \tau \sum \alpha_j^n a(\phi_j, \phi_i) &= 0 \\ \sum \alpha_j^0 (\phi_j, \phi_i) &= (u_0, \phi_i) \end{aligned}$$

In matrix form

$$\begin{aligned} C\boldsymbol{\alpha}^{n+1} &= (C - \tau A)\boldsymbol{\alpha}^n \\ C\boldsymbol{\alpha}^0 &= b \end{aligned} \tag{2.4}$$

where

$$\boldsymbol{\alpha}^n = (\alpha_1^n(t), \alpha_2^n(t), \dots, \alpha_N^n(t))^T .$$

This is so-called the explicit Euler with linear FEM. Similarly, there are the implicit Euler with linear FEM and the Crank-Nicolson with linear FEM.

### 3 Stability

- Stability of parabolic PDEs

We consider model problem (1.1). A perturbed model problem is given by

$$\begin{aligned} v_t &= v_{xx}, & x \in (0, 1), t > 0 \\ v(0, x) &= u_0(x) + \delta(x), & x \in (0, 1) \\ v(t, 0) &= v(t, 1) = 0, & t > 0 \end{aligned} \tag{3.1}$$

where  $u_0(x)$  is given.

Let  $e(t, x) = u(t, x) - v(t, x)$ . We have

$$\begin{aligned} e_t &= e_{xx}, & x \in (0, 1), t > 0 \\ e(0, x) &= \delta(x), & x \in (0, 1) \\ e(t, 0) &= e(t, 1) = 0, & t > 0 \end{aligned} \tag{3.2}$$

The model problem (1.1) is said to be stable with respect to the norm  $\|\cdot\|$ , if there exist positive constants  $K$  such that

$$\|e\| \leq K\|\delta\| .$$

- Stability of discrete methods

First we consider a general scheme

$$\begin{aligned} u^{n+1} &= Mu^n + f \\ u^0 &= b(\text{given}) \end{aligned} \tag{3.3}$$

where  $M$  is an  $N \times N$  matrix. A perturbed system is defined by

$$\begin{aligned}v^{n+1} &= Mv^n + f \\v^0 &= b + \delta\end{aligned}$$

Let  $\epsilon^n = v^n - u^n$ . Then

$$\begin{aligned}\epsilon^{n+1} &= M\epsilon^n \\ \epsilon^0 &= \delta\end{aligned}$$

and therefore,

$$\epsilon^{n+1} = M\epsilon^n = M^2\epsilon^{n-1} = \dots = M^{n+1}\epsilon^0 = M^{n+1}\delta.$$

It follows that

$$\|\epsilon^{n+1}\| \leq \|M^{n+1}\|\|\delta\|.$$

By theorem in Linear Algebra, if

$$\rho(M) := \max_j |\lambda_j(M)| < 1, \tag{3.4}$$

$$\|M^{n+1}\| \rightarrow 0 \quad \text{as } n \rightarrow \infty$$

and  $\|\epsilon^{n+1}\| \rightarrow 0$ . In this case, we say that the discrete method is (absolutely) stable.

For the explicit (forward) Euler FD scheme

$$M = \left(I - \frac{\tau}{h^2}A_1\right).$$

Then

$$\lambda_j\left(I - \frac{\tau}{h^2}A_1\right) = 1 - \tau \frac{4}{h^2} \sin^2\left(\frac{j\pi}{2(N+1)}\right).$$

The explicit Euler FD scheme is absolutely stable.

$$\tau \frac{4}{h^2} \sin^2\left(\frac{j\pi}{2(N+1)}\right) < 2$$

and approximately

$$\frac{\tau}{h^2} < \frac{1}{2}.$$

For the implicit (backward) Euler FD scheme

$$M = \left(I + \frac{\tau}{h^2}A_1\right)^{-1}.$$

We have

$$\lambda_j(M) = \lambda_j\left(\left(I + \frac{\tau}{h^2}A_1\right)^{-1}\right) = \frac{1}{1 + \tau \frac{4}{h^2} \sin^2\left(\frac{j\pi}{2(N+1)}\right)} < 1 \quad j = 1, 2, \dots, N$$

The implicit (backward) Euler FD scheme is unconditionally stable.

For the Crank-Nicolson FD scheme,

$$M = \left(I + \frac{1}{2} \frac{\tau}{h^2} A_1\right)^{-1} \left(I - \frac{1}{2} \frac{\tau}{h^2} A_1\right).$$

It is easy to verify that this method is also unconditionally stable.

For FEM, we have

$$A = \frac{1}{h} A_1, \quad C = h \left(I - \frac{1}{6} A_1\right)$$

For linear FEM with explicit Euler scheme,

$$M = C^{-1}(C - \tau A) = I - \frac{\tau}{h^2} \left(I - \frac{1}{6} A_1\right)^{-1} A_1.$$

Then

$$\lambda_j(M) = 1 - \frac{\tau}{h^2} \frac{4 \sin^2 \frac{j\pi}{2(N+1)}}{1 - \frac{2}{3} \frac{j\pi}{2(N+1)}}$$

The scheme is absolutely stable if

$$\frac{\tau}{h^2} < \frac{1}{6}.$$

We can study the stability of linear FEM methods with implicit and Crank-Nicolson schemes similarly.