



澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU



不惑新航
揚帆追夢
SET SAIL ANEW ON
THE RUBY JUBILEE

Finite Difference Methods for Parabolic PDEs

MATH 3014

Monday & Thursday 14:30-15:45

Instructor: **Dr. Luo Li**

<https://www.fst.um.edu.mo/personal/liluo/math3014/>

Department of Mathematics
Faculty of Science and Technology

A linear PDE of the form

$$u_t = Lu, \tag{4.1}$$

where t usually denotes the time and L is a linear elliptic differential operator in one or more spatial variables, is called *parabolic*.

Furthermore, the second-order canonical form

$$a(x, t)u_{tt} + 2b(x, t)u_{xt} + c(x, t)u_{xx} + \text{lower-order terms} = f(x, t)$$

is parabolic if $b^2 - ac \equiv 0$ in the entire x and t domain.

- 1D heat equation with a source

$$u_t = u_{xx} + f(x, t).$$

The dimension refers to the space variable (x direction).

- General heat equation

$$u_t = \nabla \cdot (\beta \nabla u) + \boxed{f(\mathbf{x}, t)}, \quad (4.2)$$

or the reaction term

where β is the diffusion coefficient and $f(\mathbf{x}, t)$ is the source (or sink) term.

- Diffusion–advection equation

$$u_t = \nabla \cdot (\beta \nabla u) + \mathbf{w} \cdot \nabla u + f(\mathbf{x}, t),$$

where $\nabla \cdot (\beta \nabla u)$ is the diffusion term and $\mathbf{w} \cdot \nabla u$ the advection term.

Initial and Boundary Conditions

$$\left\{ \begin{array}{ll} u_t = u_{xx} & \text{on } a < x < b \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) & t = 0 \text{ on } a < x < b \\ u(a, t) = g_1(t) & \\ u(b, t) = g_2(t) & \end{array} \right.$$

$u_0(a) = g_1(0) \text{ and } u_0(b) = g_2(0)$
for consistency

Finite difference methods applicable to elliptic PDEs can be used to treat the spatial discretization and boundary conditions, so **let us focus on the time discretization and initial condition(s)**.

4.1 The Euler Methods

$$u_t = \beta u_{xx} + f(x, t), \quad a < x < b, \quad t > 0,$$

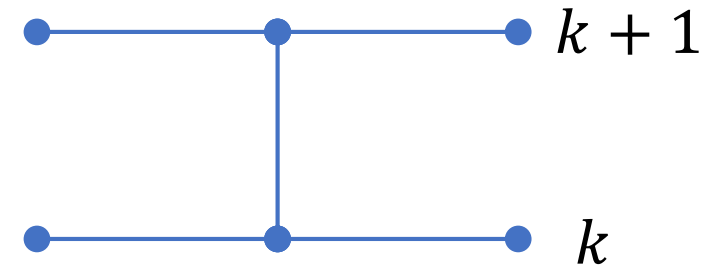
$$u(a, t) = g_1(t), \quad u(b, t) = g_2(t), \quad u(x, 0) = u_0(x),$$

Steps to find a numerical solution:

1. Generate a grid.

$$x_i = a + ih, \quad i = 0, 1, \dots, m, \quad h = \frac{b - a}{m},$$
$$t^k = k\Delta t, \quad k = 0, 1, \dots, n, \quad \Delta t = \frac{T}{n}.$$

2. Approximate the derivatives with finite difference approximations.



4.1.1 Forward Euler Method (FW-CT)

Forward FDM for u_t

Central FDM for u_{xx}

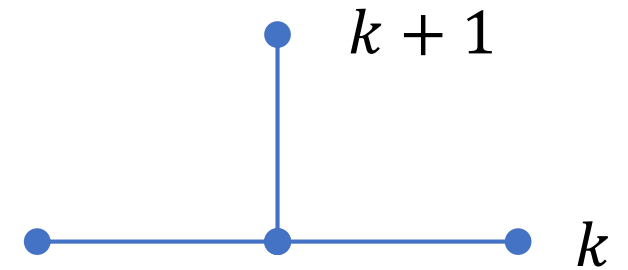
$$\frac{u(x_i, t^k + \Delta t) - u(x_i, t^k)}{\Delta t} = \beta \frac{u(x_{i-1}, t^k) - 2u(x_i, t^k) + u(x_{i+1}, t^k)}{h^2} + f(x_i, t^k) + T(x_i, t^k).$$

The local truncation error:

$$T(x_i, t^k) = -\frac{h^2 \beta}{12} u_{xxxxx}(x_i, t^k) + \frac{\Delta t}{2} u_{tt}(x_i, t^k) + \dots \rightarrow \boxed{O(h^2 + \Delta t)}$$

FW-CT scheme:
$$\frac{U_i^{k+1} - U_i^k}{\Delta t} = \beta \frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2} + f_i^k$$

When $k = 0$, U_i^0 is the initial condition at the grid point $(x_i, 0)$;



The method is called *explicit* because the solution at the next time level $k + 1$ is obtained from the solution at previous time steps.

$$U_i^{k+1} = U_i^k + \Delta t \left(\beta \frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2} + f_i^k \right), \quad i = 1, 2, \dots, m - 1. \quad (4.4)$$

Remark 4.2. If $f(x, t) \equiv 0$ and β is a constant, then from $u_t = \beta u_{xx}$ and $u_{tt} = \beta \partial u_{xx} / \partial t = \beta \partial^2 u_t / \partial x^2 = \beta^2 u_{xxxx}$, the local truncation error is

$$T(x, t) = \left(\frac{\beta^2 \Delta t}{2} - \frac{\beta h^2}{12} \right) u_{xxxx} + O\left((\Delta t)^2 + h^4\right). \quad (4.5)$$

Thus if β is constant we can choose $\Delta t = h^2 / (6\beta)$ to get $O(h^4 + (\Delta t)^2) = O(h^4)$, *i.e.*, the local truncation error is fourth-order accurate without further computational complexity, which is significant for an *explicit* method.

Example

$$u_t = \beta u_{xx} + f(x, t), \quad a < x < b, \quad t > 0,$$

$$u(a, t) = 0 \quad u(b, t) = 0$$

$$\beta = 1, \quad f(x) = -\sin t \sin(\pi x) + \cos t \sin(\pi x) \pi^2.$$

The true solution is $u(x, t) = \cos t \sin(\pi x)$.


```
a=0; b=1; t = 0;
m=10; n=20;
h = (b-a)/m; % space step size
k = (h*h)/2; % time step size
tau = k/(h*h); %  $\beta$ 
for i=1:m+1,
    x(i) = a + (i-1)*h;
    y1(i) = uexact(t,x(i));
    y2(i) = 0; % for next time level
end
plot(x,y1); hold
```

```
for j=1:n,
    y1(1)=0; % boundary condition
    y1(m+1)=0; % boundary condition
    for i=2:m
        y2(i) = y1(i) + tau*(y1(i-1)-
            2*y1(i)+y1(i+1)) + k*f(t,x(i));
    end
    plot(x,y2);
    pause(0.25)
    t=t+k;
    y1=y2; % k=k+1
end
```

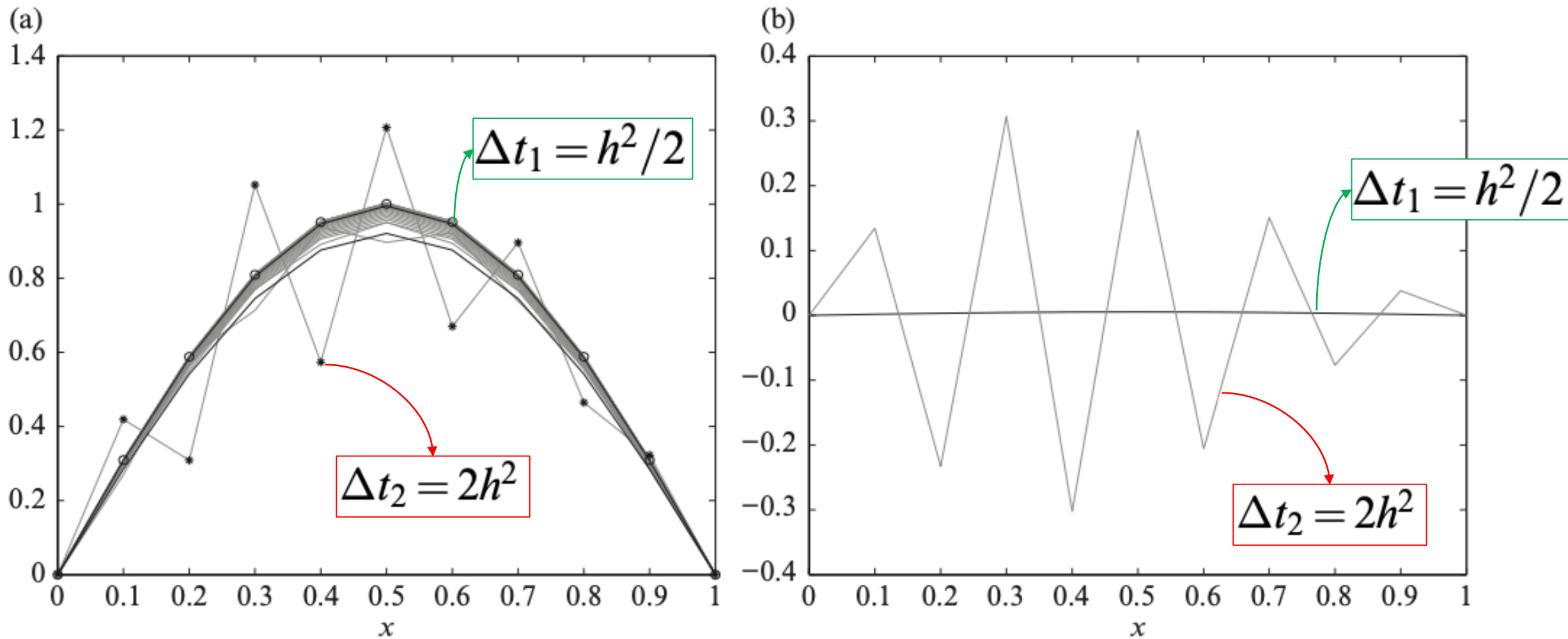


Figure 4.2. (a) Plot of the computed solutions using two different time step sizes and the exact solution at some time for the test problem. (b) Error plots of the computed solution using the two different step sizes: one is stable and the error is small; the other one is unstable and the error grows rapidly.

The method works well when $0 < \Delta t \leq \frac{h^2}{2\beta}$, but blows up when $\Delta t > \frac{h^2}{2\beta}$

The method is consistent $T(x, t) = O(h^2 + \Delta t)$

→ This is a question of **numerical stability**



A time step constraint, often called the **CFL** (Courant–Friedrichs–Lewy) stability condition

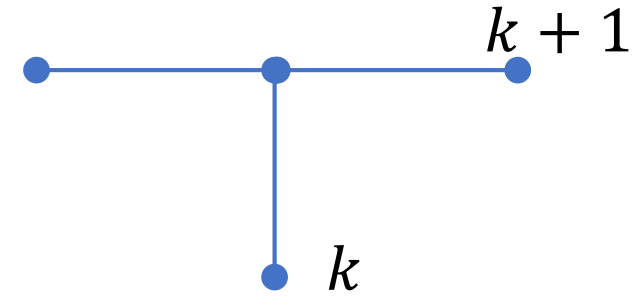
$$0 < \frac{2\beta\Delta t}{h^2} \leq 1, \quad \text{or} \quad 0 < \Delta t \leq \frac{h^2}{2\beta}$$

If not satisfying the the CFL condition, the error increases as time evolves, **blow-up** happens due to the instability of the algorithm.

4.1.2 The Backward Euler Method (BW-CT)

Backward FDM for u_t Central FDM for u_{xx}

$$\frac{U_i^{k+1} - U_i^k}{\Delta t} = \beta \frac{U_{i-1}^{k+1} - 2U_i^{k+1} + U_{i+1}^{k+1}}{h^2} + f_i^{k+1}, \quad k = 0, 1, \dots \quad (4.7)$$



To get the approximate solution at the time level $k + 1$: (implicit, coupled)

$$\begin{bmatrix} 1 + 2\mu & -\mu & & & \\ -\mu & 1 + 2\mu & -\mu & & \\ & -\mu & 1 + 2\mu & -\mu & \\ & & \ddots & \ddots & \ddots \\ & & & -\mu & 1 + 2\mu & -\mu \\ & & & & -\mu & 1 + 2\mu \end{bmatrix} \begin{bmatrix} U_1^{k+1} \\ U_2^{k+1} \\ U_3^{k+1} \\ \vdots \\ U_{m-2}^{k+1} \\ U_{m-1}^{k+1} \end{bmatrix} = \begin{bmatrix} U_1^k + \Delta t f_1^{k+1} + \mu g_1^{k+1} \\ U_2^k + \Delta t f_2^{k+1} \\ U_3^k + \Delta t f_3^{k+1} \\ \vdots \\ U_{m-2}^k + \Delta t f_{m-2}^{k+1} \\ U_{m-1}^k + \Delta t f_{m-1}^{k+1} + \mu g_2^{k+1} \end{bmatrix}$$

$\frac{\beta \Delta t}{h^2}$
 $-\mu$

Forward Euler method (FW-CT): first-order in time

The time step size $\Delta t = h^2 / (2\beta)$ to meet the CFL condition.

If $h = 0.01$, the final time is $T = 10$ and $\beta = 100$, then we need 2×10^7 steps to get the solution at the final time.

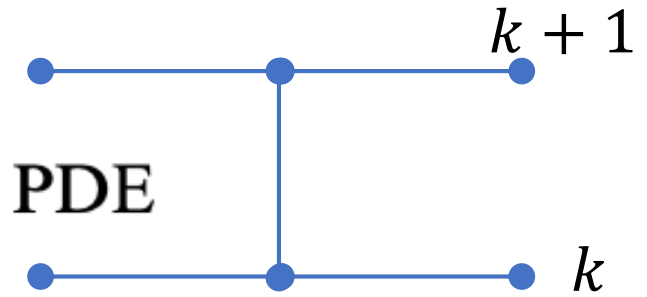
Backward Euler method (BW-CT): first-order in time

The truncation error: $T(x, t) = O(h^2 + \Delta t)$

If we want second-order accuracy $O(h^2)$, we need to take $\Delta t = O(h^2)$.

Not much better than $\Delta t = h^2 / (2\beta)$

4.3 The Crank–Nicolson Scheme



Intuitively, the Crank–Nicolson scheme approximates the PDE

$$u_t = (\beta u_x)_x + f(x, t)$$

at $(x_i, t^k + \Delta t/2)$, by averaging the time level t^k and t^{k+1} of the spatial derivative $\nabla \cdot (\beta \nabla u)$ and $f(x, t)$. Thus it has the following form

$$\begin{aligned} \frac{U_i^{k+1} - U_i^k}{\Delta t} = & \frac{\beta_{i-\frac{1}{2}}^k U_{i-1}^k - (\beta_{i-\frac{1}{2}}^k + \beta_{i+\frac{1}{2}}^k) U_i^k + \beta_{i+\frac{1}{2}}^k U_{i+1}^k}{2h^2} \\ & + \frac{\beta_{i-\frac{1}{2}}^{k+1} U_{i-1}^{k+1} - (\beta_{i-\frac{1}{2}}^{k+1} + \beta_{i+\frac{1}{2}}^{k+1}) U_i^{k+1} + \beta_{i+\frac{1}{2}}^{k+1} U_{i+1}^{k+1}}{2h^2} + \frac{1}{2} (f_i^k + f_i^{k+1}). \end{aligned} \quad (4.13)$$

The discretization is **second-order in time**.

LHS:
$$\frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} = u_t(x, t + \Delta t/2) + \frac{1}{3} \left(\frac{\Delta t}{2}\right)^2 u_{ttt}(x, t + \Delta t/2) + O((\Delta t)^4),$$

Recall that we need LHS – RHS so that u_t and u_{xx} are cancelled.

RHS:
$$\frac{u(x - h, t) - 2u(x, t) + u(x + h, t)}{2h^2} = u_{xx}(x, t) + O(h^2),$$
$$\frac{u(x - h, t + \Delta t) - 2u(x, t + \Delta t) + u(x + h, t + \Delta t)}{2h^2} = u_{xx}(x, t + \Delta t) + O(h^2),$$
$$\frac{1}{2} \left(u_{xx}(x, t) + u_{xx}(x, t + \Delta t) \right) = u_{xx}(x, t + \Delta t/2) + O((\Delta t)^2),$$

- At each time step, we need to solve a tridiagonal system of equations to get U^{k+1} . With fast tridiagonal solver (Burden and Faires 2010), the computation complexity is similar with the explicit Euler method.
- Although the Crank–Nicolson scheme is an implicit method, it is much **more efficient** than the explicit Euler method since it is **second-order** accurate.

	Forward Euler scheme	Crank–Nicolson scheme
Order in time	First-order	Second-order
Time step size	$\Delta t = \frac{h^2}{2\beta}$	$\Delta t = h$
Time steps	$\text{int} \left(\frac{2\beta T}{h^2} \right)$	$\text{int} \left(\frac{T}{h} \right)$

4.3.1 The θ -Method

For the heat equation $u_t = u_{xx} + f(x, t)$

$$\frac{U_i^{k+1} - U_i^k}{\Delta t} = \theta \delta_{xx}^2 U_i^k + (1 - \theta) \delta_{xx}^2 U_i^{k+1} + \theta f_i^k + (1 - \theta) f_i^{k+1}$$

$\theta = 1$	The explicit Euler scheme	First-order
$1/2 < \theta < 1$	The method is conditionally stable	First-order
$\theta = 1/2$	The Crank–Nicolson scheme	Second-order
$0 < \theta \leq 1/2$	The method is unconditionally stable	First-order
$\theta = 0$	The backward Euler scheme	First-order

4.4 Stability Analysis for Time-dependent Problems

----- von Neumann method based on the discrete Fourier transform

Consider $u(x) \in L^2(-\infty, \infty)$, i.e., $\int_{-\infty}^{\infty} u^2 dx < \infty$ or $\|u\|_2 < \infty$.

The space domain

$u(x)$



The frequency domain

$\hat{u}(\omega)$

The Fourier transform

$$\hat{u}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega x} u(x) dx$$

The inverse Fourier transform

$$u(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega x} \hat{u}(\omega) d\omega$$

Properties of Fourier transform

$$1. \quad \widehat{\frac{\partial u}{\partial x}} = i\omega \hat{u}$$

$$2. \quad \widehat{\frac{\partial^m u}{\partial x^m}} = (i\omega)^m \hat{u}$$

$$3. \quad \widehat{\left(\frac{\partial \hat{u}}{\partial \omega}\right)} = -ixu$$

4. The Parseval's relation

$$\|\hat{u}\|_2 = \|u\|_2 \quad \text{or} \quad \int_{-\infty}^{\infty} |\hat{u}|^2 d\omega = \int_{-\infty}^{\infty} |u|^2 dx$$

Example

$$u_t = \beta u_{xx}, \quad -\infty < x < \infty, \quad t > 0, \quad u(x, 0) = u_0(x), \quad \lim_{|x| \rightarrow \infty} u = 0,$$

Apply the Fourier transform to the PDE and the initial condition,

ODE: $\widehat{u}_t = \widehat{\beta u_{xx}}$, or $\widehat{u}_t = \beta(i\omega)^2 \widehat{u} = -\beta\omega^2 \widehat{u}$, $\widehat{u}(\omega, 0) = \widehat{u}_0(\omega)$,

and the solution of this ODE is

$$\widehat{u}(\omega, t) = \widehat{u}(\omega, 0) e^{-\beta\omega^2 t}.$$

Consequently, if $\beta > 0$, from the Parseval's relation, we have


$$\|u\|_2 = \|\widehat{u}\|_2 = \|\widehat{u}(\omega, 0) e^{-\beta\omega^2 t}\|_2 \leq \|u_0\|_2.$$

Actually, it can be seen that $\lim_{t \rightarrow \infty} \|u\|_2 = 0$ and the second-order partial derivative term is called a diffusion or dissipative. If $\beta < 0$, then $\lim_{t \rightarrow \infty} \|u\|_2 = \infty$, the PDE is dynamically unstable.

4.4.2 The Discrete Fourier Transform


Definition 4.8. If $\dots, v_{-2}, v_{-1}, v_0, v_1, v_2, \dots$ denote the values of a continuous function $v(x)$ at $x_i = ih$, the discrete Fourier transform is defined as

$$\hat{v}(\xi) = \frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} h e^{-i\xi jh} v_j. \quad (4.19)$$

$$\hat{u}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega x} u(x) dx$$


Definition 4.10. The inverse discrete Fourier transform is

$$v_j = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} \hat{v}(\xi) d\xi. \quad (4.21)$$

$$u(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega x} \hat{u}(\omega) d\omega$$


We also define the *discrete norm* as

$$\|\mathbf{v}\|_h = \sqrt{\sum_{j=-\infty}^{\infty} v_j^2 h}, \quad (4.24)$$

which is often denoted by $\|v\|_2$. Parseval's relation is also valid, *i.e.*,

$$\|\hat{v}\|_h^2 = \int_{-\pi/h}^{\pi/h} |\hat{v}(\xi)|^2 d\xi = \sum_{j=-\infty}^{\infty} h |v_j|^2 = \|\mathbf{v}\|_h^2. \quad (4.25)$$

4.4.3 Definition of the Stability of a FD Scheme

A finite difference scheme $P_{\Delta t, h} v_j^k = 0$ is stable in a stability region Λ if for any positive time T there is an integer J and a constant C_T independent of Δt and h such that

$$\|\mathbf{v}^n\|_h \leq C_T \sum_{j=0}^J \|\mathbf{v}^j\|_h, \quad (4.26)$$

for any n that satisfies $0 \leq n\Delta t \leq T$ with $(\Delta t, h) \in \Lambda$.

- The growth of the solution is at most a constant multiple of the sum of the norms of the solution at the first $J + 1$ steps.
- Λ corresponds to all possible Δt and h

Theorem 4.12. *If $\|\mathbf{v}^{k+1}\|_h \leq \|\mathbf{v}^k\|_h$ is true for any k , then the finite difference scheme is stable.*

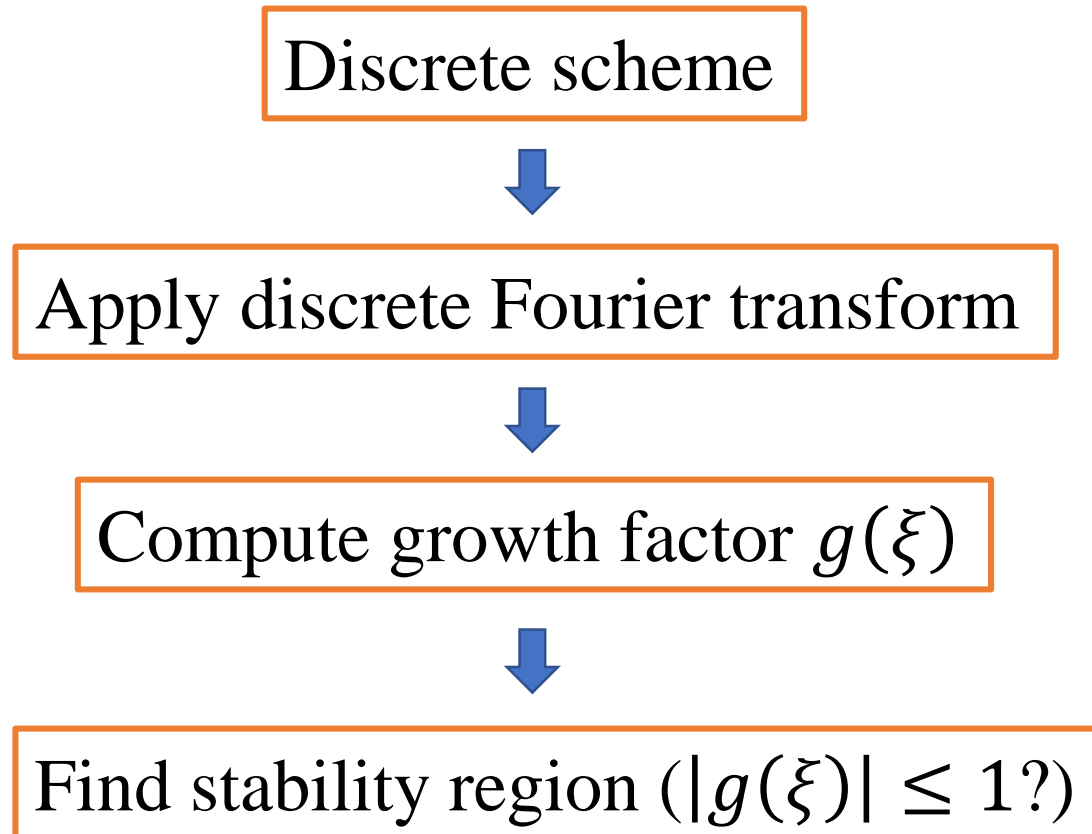
Proof From the condition, we have

$$\|\mathbf{v}^n\|_h \leq \|\mathbf{v}^{n-1}\|_h \leq \cdots \leq \|\mathbf{v}^1\|_h \leq \|\mathbf{v}^0\|_h,$$

and hence stability for $J=0$ and $C_T=1$.

This condition is stronger than (4.26), but it is more useful.

4.4.4 The von Neumann Stability Analysis for FD Methods



Example The forward Euler method (FW-CT) for the heat equation

$$u_t = \beta u_{xx}$$

Discrete scheme

$$U_j^{k+1} = U_j^k + \mu \left(U_{j-1}^k - 2U_j^k + U_{j+1}^k \right), \quad \mu = \frac{\beta \Delta t}{h^2}.$$

Apply discrete Fourier transform

$$U_j^k = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} \hat{U}^k(\xi) d\xi, \quad U_{j+1}^k = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} e^{i\xi h} \hat{U}^k(\xi) d\xi,$$

$$U_{j-1}^k = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} e^{-i\xi h} \hat{U}^k(\xi) d\xi \quad U_j^{k+1} = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} \hat{U}^{k+1}(\xi) d\xi$$

Substituting these relations into the forward Euler finite difference scheme

$$\frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} \hat{U}^{k+1}(\xi) d\xi = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh} \left(1 + \mu(e^{-i\xi h} - 2 + e^{i\xi h})\right) \hat{U}^k(\xi) d\xi$$

Growth factor

The discrete Fourier transform is **unique**, which implies

$$\hat{U}^{k+1}(\xi) = \left(1 + \mu(e^{-i\xi h} - 2 + e^{i\xi h})\right) \hat{U}^k(\xi) = g(\xi) \hat{U}^k(\xi), \quad (4.33)$$

where

Find stability region

$$g(\xi) = 1 + \mu(e^{-i\xi h} - 2 + e^{i\xi h}) \quad (4.34)$$

is called the *growth factor*. If $|g(\xi)| \leq 1$, then $|\hat{U}^{k+1}| \leq |\hat{U}^k|$ and thus $\|\hat{U}^{k+1}\|_h \leq \|\hat{U}^k\|_h$, so the finite difference scheme is stable.

From the Euler equation

$$\begin{aligned}g(\xi) &= 1 + \mu (\cos(-\xi h) - i \sin(\xi h) - 2 + \cos(\xi h) + i \sin(\xi h)) \\ &= 1 + 2\mu (\cos(\xi h) - 1) = 1 - 4\mu \sin^2(\xi h)/2,\end{aligned}\tag{4.35}$$

Note that

$$-1 \leq 1 - 4\mu \leq 1 - 4\mu \sin^2(\xi h)/2 = g(\xi) \leq 1,\tag{4.36}$$

Thus a sufficient condition for the stability of the forward Euler method is

$$-1 \leq 1 - 4\mu \quad \text{or} \quad 4\mu \leq 2, \quad \text{or} \quad \Delta t \leq \frac{h^2}{2\beta}.\tag{4.37}$$

4.4.5 Simplification of the von Neumann Stability Analysis

$$\mathbf{U}^{k+1} = f(\mathbf{U}^k, \mathbf{U}^{k+1})$$

Theorem 4.14. Let $\theta = h\xi$. A *one-step finite difference scheme* (with constant coefficients) is stable if and only if there is a constant K (independent of θ , Δt , and h) and some positive grid spacing Δt_0 and h_0 such that

$$|g(\theta, \Delta t, h)| \leq 1 + K\Delta t \quad (4.38)$$

for all θ and $0 < h \leq h_0$. If $g(\theta, \Delta t, h)$ is independent of h and Δt , then the stability condition (4.38) can be replaced by

$$|g(\theta)| \leq 1. \quad (4.39)$$

Thus only the amplification factor $g(h\xi) = g(\theta)$ needs to be considered, as observed by von Neumann.

The von Neumann stability analysis usually involves the following steps:

1. set $U_j^k = e^{ijh\xi}$ and substitute it into the finite difference scheme;

2. express U_j^{k+1} as $U_j^{k+1} = g(\xi)e^{ijh\xi}$, *etc.*;

3. solve for $g(\xi)$ and determine whether or when $|g(\xi)| \leq 1$ (for stability);

4. if there are some ξ such that $|g(\xi)| > 1$, then the method is unstable.

Example The **backward** Euler method for the heat equation $u_t = \beta u_{xx}$

Step 1. $U_j^k = e^{ijh\xi} \rightarrow U_j^{k+1} = g(\xi)e^{ijh\xi}$

Step 2. $U_j^{k+1} = U_j^k + \mu \left(U_{j-1}^{k+1} - 2U_j^{k+1} + U_{j+1}^{k+1} \right), \quad \mu = \frac{\beta \Delta t}{h^2}.$

Step 3. $g(\xi)e^{ijh\xi} = e^{ijh\xi} + \mu \left(e^{i\xi(j-1)h} - 2e^{i\xi jh} + e^{i\xi(j+1)h} \right) g(\xi)$
 $= e^{i\xi jh} \left(1 + \mu \left(e^{-i\xi h} - 2 + e^{i\xi h} \right) g(\xi) \right),$

$$g(\xi) = \frac{1}{1 - \mu(e^{-i\xi h} - 2 + e^{i\xi h})}$$

The backward Euler method is unconditionally stable.

$$= \frac{1}{1 - \mu(2 \cos(h\xi) - 2)} = \frac{1}{1 + 4\mu \sin^2(h\xi)/2} \leq 1,$$

4.5 FD Methods and Analysis for 2D Parabolic Equations

$$\left\{ \begin{array}{l} u_t + a_1 u_x + a_2 u_y = (\beta u_x)_x + (\beta u_y)_y + \kappa u + f(x, y, t), \quad \beta \geq \beta_0 > 0 \\ \text{boundary conditions} \\ \text{initial condition} \end{array} \right.$$

For the dynamic stability

The PDE can be written as $u_t = Lu + f$, where L is the spatial differential operator.

For simplicity, let us consider the heat equation $u_t = \nabla \cdot (\beta \nabla u) + f(x, y, t)$

suppose β is a constant

The forward Euler method:

$$U_{ij}^{k+1} = U_{ij}^k + \mu \left(U_{l-1,j}^k + U_{l+1,j}^k + U_{l,j-1}^k + U_{l,j+1}^k - 4U_{l,j}^k \right) + \Delta t f_{ij}^k$$

$$\mu = \beta \Delta t / h^2$$

- The method is first order in time and second order in space.
- The stability condition is $\Delta t \leq \frac{h^2}{4\beta}$.

To show stability using the von Neumann analysis with $f = 0$:

Step 1. $U_{lj}^k = e^{i(lh_x\xi_1 + jh_y\xi_2)} = e^{i\xi \cdot \mathbf{x}}$ where $\xi = [\xi_1, \xi_2]^T$ and $\mathbf{x} = [h_x l, h_y j]^T$

Step 2. $U_{lj}^{k+1} = g(\xi_1, \xi_2) e^{i\xi \cdot \mathbf{x}}$

Step 3. $g(\xi_1, \xi_2) = 1 - 4\mu \left(\sin^2(\xi_1 h/2) + \sin^2(\xi_2 h/2) \right),$

where $h_x = h_y = h$ for simplicity. If we enforce

$$-1 \leq 1 - 8\mu \leq 1 - 4\mu \left(\sin^2(\xi_1 h/2) + \sin^2(\xi_2 h/2) \right) \leq 1$$

and take $-1 \leq 1 - 8\mu$, we can guarantee that $|g(\xi_1, \xi_2)| \leq 1$

$$\left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \Delta t \leq \frac{h^2}{4\beta}$$

The backward Euler scheme

$$\frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \frac{U_{i-1,j}^{k+1} + U_{i+1,j}^{k+1} + U_{i,j-1}^{k+1} + U_{i,j+1}^{k+1} - 4U_{ij}^{k+1}}{h^2} + f_{ij}^{k+1}, \quad (4.48)$$

Local truncation error

$$\mathcal{O}((\Delta t) + h^2)$$

The Crank–Nicolson scheme

$$\begin{aligned} \frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \frac{1}{2} & \left(\frac{U_{i-1,j}^{k+1} + U_{i+1,j}^{k+1} + U_{i,j-1}^{k+1} + U_{i,j+1}^{k+1} - 4U_{ij}^{k+1}}{h^2} + f_{ij}^{k+1} \right. \\ & \left. + \frac{U_{i-1,j}^k + U_{i+1,j}^k + U_{i,j-1}^k + U_{i,j+1}^k - 4U_{ij}^k}{h^2} + f_{ij}^k \right). \quad (4.49) \end{aligned}$$

$$\mathcal{O}((\Delta t)^2 + h^2)$$

- These schemes are unconditionally stable for linear problems.
- Need to solve a system of equations with a block tridiagonal coefficient matrix.

4.6 The Alternating Direction Implicit (ADI) Method

The ADI is a *time splitting* or *fractional step* method.

For the heat equation $u_t = u_{xx} + u_{yy} + f(x, y, t)$, the ADI method is

$$\begin{aligned} \frac{U_{ij}^{k+\frac{1}{2}} - U_{ij}^k}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^k - 2U_{ij}^k + U_{i,j+1}^k}{h_y^2} + f_{ij}^{k+\frac{1}{2}}, \\ \frac{U_{ij}^{k+1} - U_{ij}^{k+\frac{1}{2}}}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i,j+1}^{k+\frac{1}{2}}}{h_y^2} + f_{ij}^{k+\frac{1}{2}}, \end{aligned} \quad (4.50)$$

- Second order in time and in space
- Unconditionally stable for linear problems
- Fast tridiagonal solvers can be used!

We can use symbolic expressions to discuss the method

$$\begin{aligned}U_{ij}^{k+\frac{1}{2}} &= U_{ij}^k + \frac{\Delta t}{2} \delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \delta_{yy}^2 U_{ij}^k + \frac{\Delta t}{2} f_{ij}^{k+\frac{1}{2}}, \\U_{ij}^{k+1} &= U_{ij}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \delta_{yy}^2 U_{ij}^{k+1} + \frac{\Delta t}{2} f_{ij}^{k+\frac{1}{2}}.\end{aligned}\tag{4.51}$$

Thus on moving unknowns to the left-hand side, in **matrix-vector form** we have

$$\begin{aligned}\left(I - \frac{\Delta t}{2} D_x^2\right) \mathbf{U}^{k+\frac{1}{2}} &= \left(I + \frac{\Delta t}{2} D_y^2\right) \mathbf{U}^k + \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}}, \\ \left(I - \frac{\Delta t}{2} D_y^2\right) \mathbf{U}^{k+1} &= \left(I + \frac{\Delta t}{2} D_x^2\right) \mathbf{U}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}},\end{aligned}\tag{4.52}$$

Recall that

The k -th finite difference equation corresponding to (i, j)

$$k = i + (m - 1)(j - 1), \quad i = 1, 2, \dots, m - 1, \quad j = 1, 2, \dots, n - 1 \quad (3.21)$$

7	8	9	
4	5	6	
1	2	3	

$$h_x = h_y = h$$

$$\mathbf{U} = \begin{bmatrix} U_{11} \\ U_{21} \\ U_{31} \\ \vdots \\ U_{23} \\ U_{33} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_8 \\ x_9 \end{bmatrix}$$

From the **first equation** we get

$$\mathbf{U}^{k+\frac{1}{2}} = \left(I - \frac{\Delta t}{2} D_x^2 \right)^{-1} \left(I + \frac{\Delta t}{2} D_y^2 \right) \mathbf{U}^k + \left(I - \frac{\Delta t}{2} D_x^2 \right)^{-1} \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}},$$

and substituting into the **second equation** to have

$$\begin{aligned} \left(I - \frac{\Delta t}{2} D_y^2 \right) \mathbf{U}^{k+1} &= \left(I + \frac{\Delta t}{2} D_x^2 \right) \left(I - \frac{\Delta t}{2} D_x^2 \right)^{-1} \left(I + \frac{\Delta t}{2} D_y^2 \right) \mathbf{U}^k \\ &\quad + \left(I + \frac{\Delta t}{2} D_x^2 \right) \left(I - \frac{\Delta t}{2} D_x^2 \right)^{-1} \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}}. \end{aligned}$$

Applying $\left(I - \frac{\Delta t}{2} D_x^2\right)$ to both sides and using the following commutative operation

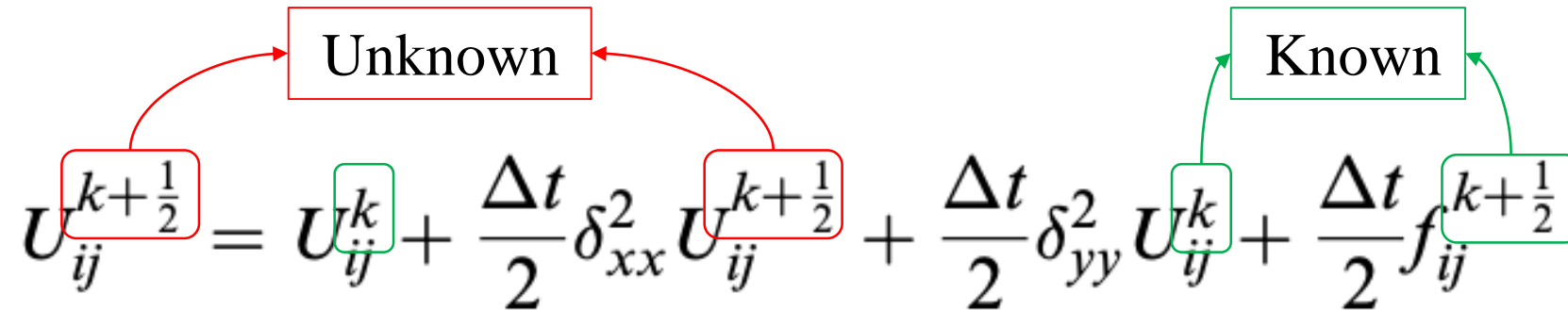
$$\left(I - \frac{\Delta t}{2} D_x^2\right) \left(I + \frac{\Delta t}{2} D_x^2\right) = \left(I + \frac{\Delta t}{2} D_x^2\right) \left(I - \frac{\Delta t}{2} D_x^2\right)$$

we can go further to get

$$\begin{aligned} \left(I - \frac{\Delta t}{2} D_x^2\right) \left(I - \frac{\Delta t}{2} D_y^2\right) \mathbf{U}^{k+1} &= \left(I + \frac{\Delta t}{2} D_x^2\right) \left(I + \frac{\Delta t}{2} D_y^2\right) \mathbf{U}^k \\ &+ \left(I + \frac{\Delta t}{2} D_x^2\right) \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}} + \left(I - \frac{\Delta t}{2} D_x^2\right) \frac{\Delta t}{2} \mathbf{F}^{k+\frac{1}{2}}. \end{aligned}$$

4.6.1 Implementation of the ADI Algorithm

In the x –direction,

$$U_{ij}^{k+\frac{1}{2}} = U_{ij}^k + \frac{\Delta t}{2} \delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \frac{\Delta t}{2} \delta_{yy}^2 U_{ij}^k + \frac{\Delta t}{2} f_{ij}^{k+\frac{1}{2}}$$


For a fixed j , we get a tridiagonal system of equations for $U_{1j}^{k+\frac{1}{2}}, U_{2j}^{k+\frac{1}{2}}, \dots, U_{m-1,j}^{k+\frac{1}{2}}$, assuming a Dirichlet boundary condition at $x=a$ and $x=b$.

$$\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} = \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2}$$

Let $\mu = \frac{\beta \Delta t}{2h^2}$, and $f_i^{k+\frac{1}{2}} = f(x_i, t^{k+\frac{1}{2}})$.

The system of equations in matrix-vector form is

$$\begin{bmatrix}
 1 + 2\mu & -\mu & & & & & \\
 -\mu & 1 + 2\mu & -\mu & & & & \\
 & -\mu & 1 + 2\mu & -\mu & & & \\
 & & & \ddots & \ddots & \ddots & \\
 & & & & -\mu & 1 + 2\mu & -\mu \\
 & & & & & -\mu & 1 + 2\mu
 \end{bmatrix}
 \begin{bmatrix}
 U_{1j}^{k+\frac{1}{2}} \\
 U_{2j}^{k+\frac{1}{2}} \\
 U_{3j}^{k+\frac{1}{2}} \\
 \vdots \\
 U_{m-2,j}^{k+\frac{1}{2}} \\
 U_{m-1,j}^{k+\frac{1}{2}}
 \end{bmatrix}
 = \hat{\mathbf{F}},$$

For fixed j

where

Boundary condition

$$\hat{\mathbf{F}} = \begin{bmatrix} U_{1,j}^k + \frac{\Delta t}{2} f_{1j}^{k+\frac{1}{2}} + \mu u_{bc}(a, y_j)^{k+\frac{1}{2}} + \mu (U_{1,j-1}^k - 2U_{1,j}^k + U_{1,j+1}^k) \\ U_{2,j}^k + \frac{\Delta t}{2} f_{2j}^{k+\frac{1}{2}} + \mu (U_{2,j-1}^k - 2U_{2,j}^k + U_{2,j+1}^k) \\ U_{3j}^k + \frac{\Delta t}{2} f_{3j}^{k+\frac{1}{2}} + \mu (U_{3,j-1}^k - 2U_{3,j}^k + U_{3,j+1}^k) \\ \vdots \\ U_{m-2,j}^k + \frac{\Delta t}{2} f_{m-2,j}^{k+\frac{1}{2}} + \mu (U_{m-2,j-1}^k - 2U_{m-2,j}^k + U_{m-2,j+1}^k) \\ U_{m-1,j}^k + \frac{\Delta t}{2} f_{m-1,j}^{k+\frac{1}{2}} + \mu (U_{m-1,j-1}^k - 2U_{m-1,j}^k + U_{m-1,j+1}^k) + \mu u_{bc}(b, y_j)^{k+\frac{1}{2}} \end{bmatrix}$$

Loop in the x direction

```
%----- loop in the x direction
for j = 2:n, % Loop for fixed j
    A = sparse(m-1,m-1);
    b=zeros(m-1,1);
    for i=2:m,
        b(i-1) = (u1(i,j-1) -2*u1(i,j) + u1(i,j+1))/h1
                + f(t2,x(i),y(j)) + 2*u1(i,j)/dt;

        ...

        A(i-1,i-1) = 2/dt + 2/h1;
    end
end
ut = A\b; % Solve the diagonal matrix
%----- loop in the y direction
...
```

```
if i == 2
    b(i-1) = b(i-1) + uexact(t2,x(i-1),y(j))/h1;
    A(i-1,i) = -1/h1;
else
    if i==m
        b(i-1) = b(i-1) + uexact(t2,x(i+1),y(j))/h1;
        A(i-1,i-2) = -1/h1;
    else
        A(i-1,i) = -1/h1;
        A(i-1,i-2) = -1/h1;
    end
end
end
```

Boundary condition

- Why $i-1$? Because A is created with size $(m-1) \times (m-1)$, (i,j) here is the point position
- Just regard it as “ i ”

Loop in the y direction

```
%----- loop in the y direction
for i = 2:m, % Loop for fixed i
    A = sparse(m-1,m-1);
    b=zeros(m-1,1);
    for j=2:n,
        b(j-1) = (u2(i-1,j) -2*u2(i,j) + u2(i+1,j))/h1
                + f(t2,x(i),y(j)) + 2*u2(i,j)/dt;
        ...
        A(j-1,j-1) = 2/dt + 2/h1;
    end
end
ut = A\b; % Solve the diagonal matrix
```

```
if j == 2
    b(j-1) = b(j-1) + uexact(t1,x(i),y(j-1))/h1;
    A(j-1,j) = -1/h1;
else
    if j==n
        b(j-1) = b(j-1) + uexact(t1,x(i),y(j+1))/h1;
        A(j-1,j-2) = -1/h1;
    else
        A(j-1,j) = -1/h1;
        A(j-1,j-2) = -1/h1;
    end
end
end
```

Boundary condition

4.6.2 Consistency of the ADI Method

$$\begin{aligned} \frac{U_{ij}^{k+\frac{1}{2}} - U_{ij}^k}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^k - 2U_{ij}^k + U_{i,j+1}^k}{h_y^2} + f_{ij}^{k+\frac{1}{2}}, \\ \frac{U_{ij}^{k+1} - U_{ij}^{k+\frac{1}{2}}}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^{k+1} - 2U_{ij}^{k+1} + U_{i,j+1}^{k+1}}{h_y^2} + f_{ij}^{k+\frac{1}{2}}, \end{aligned} \quad (4.50)$$

Adding the two equations in (4.50) together, we get

$$\frac{U_{ij}^{k+1} - U_{ij}^k}{(\Delta t)/2} = 2\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \delta_{yy}^2 (U_{ij}^{k+1} + U_{ij}^k) + 2f_{ij}^{k+\frac{1}{2}}; \quad (4.53)$$

$$\frac{U_{ij}^{k+\frac{1}{2}} - U_{ij}^k}{(\Delta t)/2} = \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^k - 2U_{ij}^k + U_{i,j+1}^k}{h_y^2} + f_{ij}^{k+\frac{1}{2}},$$

$$\frac{U_{ij}^{k+1} - U_{ij}^{k+\frac{1}{2}}}{(\Delta t)/2} = \frac{U_{i-1,j}^{k+\frac{1}{2}} - 2U_{ij}^{k+\frac{1}{2}} + U_{i+1,j}^{k+\frac{1}{2}}}{h_x^2} + \frac{U_{i,j-1}^{k+1} - 2U_{ij}^{k+1} + U_{i,j+1}^{k+1}}{h_y^2} + f_{ij}^{k+\frac{1}{2}},$$

(4.50)

and if we subtract the first equation from the second equation, we get

$$4U_{ij}^{k+\frac{1}{2}} = 2 \left(U_{ij}^{k+1} + U_{ij}^k \right) - \Delta t \delta_{yy}^2 \left(U_{ij}^{k+1} - U_{ij}^k \right). \quad (4.54)$$

Substituting into (4.53):

$$\frac{U_{ij}^{k+1} - U_{ij}^k}{(\Delta t)/2} = 2\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \delta_{yy}^2 (U_{ij}^{k+1} + U_{ij}^k) + 2f_{ij}^{k+\frac{1}{2}}; \quad (4.53)$$

$$4U_{ij}^{k+\frac{1}{2}} = 2(U_{ij}^{k+1} + U_{ij}^k) - \Delta t \delta_{yy}^2 (U_{ij}^{k+1} - U_{ij}^k). \quad (4.54)$$

$$\left(1 + \frac{(\Delta t)^2}{4} \delta_{xx}^2 \delta_{yy}^2\right) \frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \left(\delta_{xx}^2 + \delta_{yy}^2\right) \frac{U_{ij}^{k+1} + U_{ij}^k}{2} + f_{ij}^{k+\frac{1}{2}}, \quad (4.55)$$

The discretization is **second-order in time and space** *i.e.*, $T_{ij}^k = O((\Delta t)^2 + h^2)$.

$$\left(1 + \frac{(\Delta t)^2}{4} \delta_{xx}^2 \delta_{yy}^2\right) \frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \left(\delta_{xx}^2 + \delta_{yy}^2\right) \frac{U_{ij}^{k+1} + U_{ij}^k}{2} + f_{ij}^{k+\frac{1}{2}}, \quad (4.55)$$

LHS:

$$\frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} = u_t(x, t + \Delta t/2) + \frac{1}{3} \left(\frac{\Delta t}{2}\right)^2 u_{ttt}(x, t + \Delta t/2) + O((\Delta t)^4),$$

$$\frac{(\Delta t)^2}{4} \left(u_t(x, t + \Delta t/2) + \dots \right)$$

$$\left(1 + \frac{(\Delta t)^2}{4} \delta_{xx}^2 \delta_{yy}^2\right) \frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \left(\delta_{xx}^2 + \delta_{yy}^2\right) \frac{U_{ij}^{k+1} + U_{ij}^k}{2} + f_{ij}^{k+\frac{1}{2}}, \quad (4.55)$$

RHS:

$$\frac{u(x-h, t) - 2u(x, t) + u(x+h, t)}{2h^2} = u_{xx}(x, t) + O(h^2),$$

$$\frac{u(x-h, t + \Delta t) - 2u(x, t + \Delta t) + u(x+h, t + \Delta t)}{2h^2} = u_{xx}(x, t + \Delta t) + O(h^2),$$

$$\frac{1}{2} \left(u_{xx}(x, t) + u_{xx}(x, t + \Delta t) \right) = u_{xx}(x, t + \Delta t/2) + O((\Delta t)^2),$$

4.6.3 Stability Analysis of the ADI Method

Taking $f = 0$ and setting

$$U_{lj}^k = e^{i(\xi_1 h_1 l + \xi_2 h_2 j)}, \quad U_{lj}^{k+1} = g(\xi_1, \xi_2) e^{i(\xi_1 h_1 l + \xi_2 h_2 j)}, \quad (4.56)$$

from Page 39, we have

$$\left(1 - \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 - \frac{\Delta t}{2} \delta_{yy}^2\right) \mathbf{U}_{jl}^{k+1} = \left(1 + \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 + \frac{\Delta t}{2} \delta_{yy}^2\right) \mathbf{U}_{jl}^k,$$

Substitute into the equation

$$\begin{aligned} & \left(1 - \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 - \frac{\Delta t}{2} \delta_{yy}^2\right) g(\xi_1, \xi_2) e^{i(\xi_1 h_1 l + \xi_2 h_2 j)} \\ &= \left(1 + \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 + \frac{\Delta t}{2} \delta_{yy}^2\right) e^{i(\xi_1 h_1 l + \xi_2 h_2 j)}. \end{aligned}$$

Example The backward Euler method for the 1D heat equation $u_t = \beta u_{xx}$

Step 1.

$$U_j^k = e^{ijh\xi} \quad U_j^{k+1} = g(\xi) e^{ijh\xi}$$

Step 2.

$$U_i^{k+1} = U_i^k + \mu \left(U_{i-1}^{k+1} - 2U_i^{k+1} + U_{i+1}^{k+1} \right), \quad \mu = \frac{\beta \Delta t}{h^2}.$$

Step 3.

$$\begin{aligned} g(\xi) e^{ijh\xi} &= e^{ijh\xi} + \mu \left(e^{i\xi(j-1)h} - 2e^{i\xi jh} + e^{i\xi(j+1)h} \right) g(\xi) \\ &= e^{i\xi jh} \left(1 + \mu \left(e^{-i\xi h} - 2 + e^{i\xi h} \right) g(\xi) \right), \end{aligned}$$

$$g(\xi) = \frac{1}{1 - \mu(e^{-i\xi h} - 2 + e^{i\xi h})}$$

The backward Euler method is unconditionally stable.

$$= \frac{1}{1 - \mu(2 \cos(h\xi) - 2)} = \frac{1}{1 + 4\mu \sin^2(h\xi)/2} \leq 1,$$



$$\begin{aligned} & \left(1 - \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 - \frac{\Delta t}{2} \delta_{yy}^2\right) g(\xi_1, \xi_2) e^{i(\xi_1 h_1 l + \xi_2 h_2 j)} \\ &= \left(1 + \frac{\Delta t}{2} \delta_{xx}^2\right) \left(1 + \frac{\Delta t}{2} \delta_{yy}^2\right) e^{i(\xi_1 h_1 l + \xi_2 h_2 j)}. \end{aligned}$$

After some manipulations, we get

$$g(\xi_1, \xi_2) = \frac{\left(1 - 4\mu \sin^2(\xi_1 h/2)\right) \left(1 - 4\mu \sin^2(\xi_2 h/2)\right)}{\left(1 + 4\mu \sin^2(\xi_1 h/2)\right) \left(1 + 4\mu \sin^2(\xi_2 h/2)\right)},$$

where $\mu = \frac{\Delta t}{2h^2}$ and for simplicity we have set $h_x = h_y = h$. Thus $|g(\xi_1, \xi_2)| \leq 1$, no matter what Δt and h are, so the ADI method is unconditionally stable for linear heat equations.

4.7 An Implicit–explicit Method for Diffusion and Advection Equations

$$u_t + \mathbf{w} \cdot \nabla u = \nabla \cdot (\beta \nabla u) + f(x, y, t)$$

a 2D vector the gradient operator $\nabla = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right]^T$

- An **implicit** scheme for the **diffusion** term ()
- An **explicit** scheme for the **advection** term

$$\frac{u^{k+1} - u^k}{\Delta t} + (\mathbf{w} \cdot \nabla_h u)^{k+\frac{1}{2}} = \frac{1}{2} \left((\nabla_h \cdot \beta \nabla_h u)^k + (\nabla_h \cdot \beta \nabla_h u)^{k+1} \right) + f^{k+\frac{1}{2}}, \quad (4.57)$$

Where $(\mathbf{w} \cdot \nabla_h u)^{k+\frac{1}{2}} = \frac{3}{2} (\mathbf{w} \cdot \nabla_h u)^k - \frac{1}{2} (\mathbf{w} \cdot \nabla_h u)^{k-1},$

$$\nabla_h u = [\delta_x u, \delta_y u]^T, \quad \delta_x u = \frac{u_{i+1,j} - u_{i-1,j}}{2h_x}; \quad \delta_y u = \frac{u_{i,j+1} - u_{i,j-1}}{2h_y}.$$

The CFL constraint is not a main concern unless $\| \mathbf{w} \|$ is very large. The time step constraint is

$$\Delta t \leq \frac{h}{2\|\mathbf{w}\|_2}. \quad (4.60)$$

At each time step, we need to solve a **generalized Helmholtz** equation

$$(\nabla \cdot \beta \nabla u)^{k+1} - \frac{2u^{k+1}}{\Delta t} = -\frac{2u^k}{\Delta t} + 2(\mathbf{w} \cdot \nabla u)^{k+\frac{1}{2}} - (\nabla \cdot \beta \nabla u)^k - 2f^{k+\frac{1}{2}}. \quad (4.61)$$

We need \mathbf{u}^1 to get the scheme above started. We can use the **explicit Euler method (FW-CT)** to approximate \mathbf{u}^1 .

