8 Finite difference methods for parabolic partial differential equations

8.1 Introduction

A linear parabolic PDE has the following form

$$u_t = Lu \tag{8.1}$$

where L is a linear elliptic differential operator. We list some examples below.

- One dimensional heat equation with a source.

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

  The dimension is referred to the space variable even though there are two independent variables x and t.

- The general one dimensional second order partial differential equations the following form:

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

  If $b^2 - ac \equiv 0$ in the entire domain, then the equation is parabolic.

- A general heat equation in any dimensions can be written as

  $$u_t = \nabla u \cdot (\beta \nabla u) + f(\mathbf{x}, t) \tag{8.2}$$

  where β is called the heat conductivity, $f(\mathbf{x}, t)$ is called a source term (including a sink as well).

- A diffusion and advection equation has the following form

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

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

- A canonical form of a diffusion and reaction equation is

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

  The non-linear source term is called a reaction term.

- A steady state solution meaning $u_t = 0$ of a parabolic PDE is an elliptic PDE.

8.2 Initial and boundary conditions and dynamic stability.

For time dependent problems, we should have an initial condition, usually at $t = 0$, that is, $u(\mathbf{x}, 0) = u_0(\mathbf{x})$ is given. We also need boundary conditions as well. As an example, for one-dimensional heat equation $u_t = u_{xx}$, $a < x < b$, we should have boundary conditions at $x = a$ and $x = b$, and an initial condition at $t = 0$. There is consistency condition at $(a, 0)$ and $(b, 0)$. For example, if a Dirichlet boundary condition is prescribed at $x = a$ and $x = b$ such that $u(a, t) = g_1(t)$ and $u(b, t) = g_2(t)$, then the consistency condition is $u_0(a) = g_1(0)$, and $u_0(b) = g_2(0)$.

The dynamical stability.

The fundamental solution for one dimensional heat equation is $u_t = u_{xx}$ is $\frac{e^{-x^2/4t}}{\sqrt{4\pi t}}$. The solution is uniformly bounded. However, for the backward heat equation, $u_t = -u_{xx}$, if $u(x, 0 \neq 0$, then $\lim_{t\to\infty} u(x, t) = \infty$. We call the solution is dynamically *unstable* if $u(x, t)$ is not uniformly bounded. In other words, we can not find a positive constant such that $|u(x, t)| \leq C$. There are some applications of unstable problems, sometimes called blow-up problems. We will not discuss how to solve those dynamically unstable problems numerically here. We will only concentrate dynamically stable problems.

Some commonly used finite difference methods will be discussed in this section are listed below:

- the forward, backward Euler's methods;
- the Crank-Nicolson and the-θ method;
- the method of line (MOL) if a good ODE solver can be applied;
- the alternating directional implicit (ADI) method for high dimensional problems.

We can use the finite difference methods for elliptic problems to take care of the spatial discretization and the boundary conditions. The crucial discussion here is the time discretization. In terms of the stability of numerical methods, we will use the Fourier transformation and the von Neumann stability analysis.

8.3 The Euler's method

Consider the heat equation:

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

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

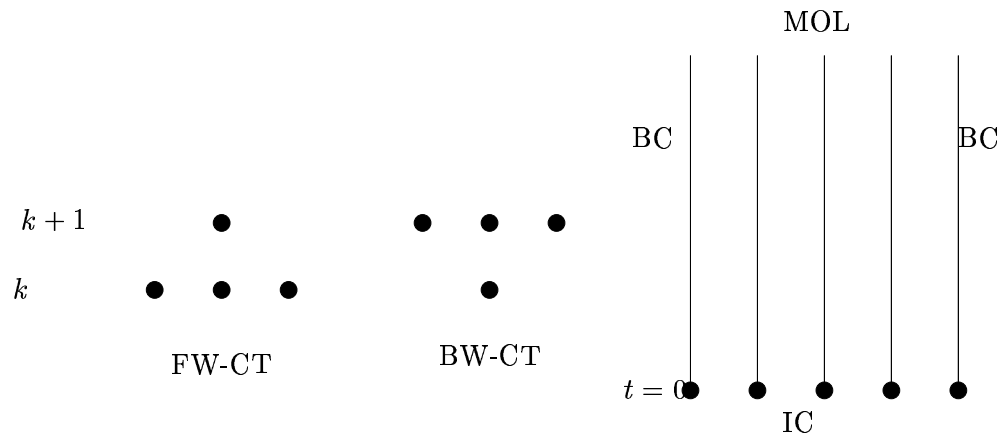

Figure 12: A diagram of the finite difference stencils of the forward, backward Euler method, and the method of line (MOL) approach.

We want to approximate the solution at certain time T, $0 < T$, or all the solution between $0 < t < T$.

As the first step, we generate a grid

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

However, we should know that we can not use arbitrary Δt or n for explicit methods because of the stability concerns.

The second step is to approximate the derivatives with finite difference formulas. We know how to discretize the spatial derivatives. Let us try different finite difference formulas for the time derivative.

Forward Euler's method (FT-CT).

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

The local truncation error is

$$T(x_i, t^k) = \frac{h^2 \beta}{12} u_{xxxx} + \frac{\Delta t}{2} u_{tt} + \text{h.o.t.}$$

The discretization is order $O(h^2+\Delta t)$. Often we call the discretization is first order in time, second order in space. The finite difference equation then is

$$\frac{U_i^{k+1}-U_i^k}{\Delta t}=\beta\,\frac{U_{i-1}^k-2U_i^k+U_{i+1}^k}{h^2}+f_i^k, \tag{8.3}$$

where $f_i^k=f(x_i,t^k)$. The solution of the finite difference equations U_i^k is an approximation to the exact $u(x_i,t^k)$. Note that when $k=0$, we have the initial condition at the grid points $(x_i,0)$. If we know the solution at the time level k, the solution of the finite difference equation at the next time level is

$$U_i^{k+1}=U_i^k+\Delta t\left(\beta\frac{U_{i-1}^k-2U_i^k+U_{i+1}^k}{h^2}+f_i^k\right),\quad i=1,2,\cdots,m-1. \tag{8.4}$$

Therefore we can get the solution of the finite difference equations directly from the approximate solution at previous time steps. We do not need to solve a system of equations. Such a method is called *explicit* time marching method.

Remark 8.1 *According to our definition, the local truncation is*

$$T(x,t)=\frac{u(x,t+\Delta t)-u(x,t)}{\Delta t}-\beta\,\frac{u(x-h,t)-2u(x,t)+u(x+h,t)}{h^2}-f(x,t)=O(h^2+\Delta t)$$

However, in the literature, there is another definition using

$$\begin{aligned}T(x,t) &= u(x,t+\Delta t)-u(x,t)-\Delta t\left(\beta\frac{u(x-h,t)-2u(x,t)+u(x+h,t)}{h^2}-f(x,t)\right)\\ &= O\left(\Delta t(h^2+\Delta t)\right).\end{aligned}$$

The difference is the factor of Δt. According to this definition, the local truncation error is one order of Δt higher.

Remark 8.2 *If $f(x,t)\equiv 0$ and β is a constant, then $u_t=\beta u_{xx}$, and $u_{tt}=\beta\frac{\partial u_{xx}}{\partial t}=\beta\frac{\partial^2 u_t}{\partial x^2}=\beta^2 u_{xxxx}$, and the local truncation error is*

$$T(x,t)=\left(\frac{\beta^2\Delta t}{2}-\frac{\beta h^2}{12}\right)u_{xxxx}+O\left((\Delta t)^2+h^4\right). \tag{8.5}$$

Therefore if β is a constant, we can choose $\Delta t=\frac{h^2}{6\beta}$ to get $O(h^4+(\Delta t)^2)=O(h^4)=O(\Delta t^2)$ order of accuracy without increase computational complexity. This is significant for the explicit *method.*

If we try the numerical method with different Δt and check the error against a problem that we know its exact solution, we see the method works for some Δt and blows up for some other Δt. Since the method is consistent, it has something to do with the stability. Intuitively, we can see that to prevent the errors in u_i^k from getting amplified, we should set

$$\frac{2\beta\Delta t}{h^2}\le 1,\quad \text{or}\quad \Delta t\le\frac{h^2}{2\beta}. \tag{8.6}$$

The backward Euler's method (BW-CT).

If we use the backward finite difference formula for u_t at (x_i, t^k), we get

$$\frac{U_i^k - U_i^{k-1}}{\Delta t} = \beta \frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2} + f_i^k, \quad k = 1, 2, \cdots .$$

However, this is equivalent to the conventional expression

$$\frac{U_i^{k+1} - U_i^k}{\Delta t} = \beta \frac{U_{i-1}^{k+1} - 2U_i^{k+1} + U_{i+1}^{k+1}}{h^2} + f_i^{k+1}. \tag{8.7}$$

The backward Euler's method is consistent. The discretization error is still $O(\Delta t + h^2)$.
Question: *Can we choose Δt to increase the order of accuracy*?

However, we can not get u_i^{k+1} with a few simple algebraic operations because all u_i^{k+1}'s are coupled together. We need to solve the following tri-diagonal system in order to get the approximate solution at time level $k+1$:

$$\begin{bmatrix} 1+2\mu & -\mu & & & & & \\ -\mu & 1+2\mu & -\mu & & & & \\ & -\mu & 1+2\mu & -\mu & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -\mu & 1+2\mu & -\mu & \\ & & & & -\mu & 1+2\mu & \end{bmatrix} \begin{bmatrix} U_1^{k+1} \\ U_2^{k+1} \\ U_3^{k+1} \\ \vdots \\ U_{m-2}^{k+1} \\ U_{m-1}^{k+1} \end{bmatrix} = \begin{bmatrix} U_1^k + \Delta t\, f_1^{k+1} + \mu g_1^{k+1} \\ U_2^k + \Delta t\, f_2^{k+1} \\ U_3^k + \Delta t\, f_3^{k+1} \\ \vdots \\ U_{m-2}^k + \Delta t f_{m-2}^{k+1} \\ U_{m-1}^k + \Delta t\, f_{m-1}^{k+1} + \mu g_2^{k+1} \end{bmatrix} \tag{8.8}$$

where $\mu = \frac{\beta \Delta t}{h^2}$ and $f_i^{k+1} = f(x_i, t^{k+1})$. Note that we can use $f(x_i, t^k)$ instead of $f(x_i, t^{k+1})$ if the method is first order accurate in time. Such a numerical method is called an implicit time marching method because the solution at time level $k+1$ are coupled together. What is the advantage of the backward Euler's method? It is stable for any choice of Δt. For one dimensional problems, the computational cost is only slightly more than the explicit Euler's method if we can use an efficient tridiagonal solver.

8.4 The method of line (MOL).

If there is a good solver for ordinary differential equations/systems, we can use the method of line (MOL) to solve the parabolic partial differential equations.

Given a general parabolic equation of the form

$$u_t(x,t) = Lu(x,t) + f(x,t),$$

where L is an elliptic operator. Let L_h be a finite difference operator acting on a grid $x_i = a + ih$. We can form a semi-discrete system of ordinary differential equations of the

following form

$$\frac{\partial U_i}{\partial t} = L_h U_i(t) + f_i(t).$$

In other words, we only discretize the spatial variable. For the heat equation with a source $u_t = \beta u_{xx} + f$, we have $L = \frac{\partial^2}{\partial x^2}$, $L_h = \delta_{xx}^2$, the discretize system of ODE is:

$$\begin{aligned}
\frac{\partial U_1(t)}{\partial t} &= \beta \frac{-2U_1(t) + U_2(t)}{h^2} + \frac{g_1(t)}{h^2} + f(x_1, t) \\
\frac{\partial U_i(t)}{\partial t} &= \beta \frac{U_{i-1}(t) - 2U_2(t) + U_{i+1}(t)}{h^2} + f(x_i, t), \quad i = 2, 3, \cdots, m-2, \\
\frac{\partial U_{m-1}(t)}{\partial t} &= \beta \frac{U_{m-2}(t) - 2U_{i-1}(t)}{h^2} + \frac{g_2(t)}{h^2} + f(x_{m-1}, t).
\end{aligned} \tag{8.9}$$

The initial condition is

$$U_i(0) = u_0(x_i, 0), \quad i = 1, 2, \cdots, m-1. \tag{8.10}$$

The ODE system can be written as a vector form

$$\frac{d\mathbf{y}}{dt} = f(\mathbf{y}, t), \qquad \mathbf{y}(0) = \mathbf{y}_0. \tag{8.11}$$

The MOL is especially useful for non-linear PDEs of the form $u_t = f(\frac{\partial}{\partial x}, u, t)$. For linear problems, we typically have

$$\frac{d\mathbf{y}}{dt} = A\mathbf{y} + \mathbf{c}$$

where A is a matrix, and $\mathbf{c}$ is a vector. Both A and $\mathbf{c}$ may depend on t.

There are many efficient solvers for a system of ODES. Most of them are based on high order Runge-Kutta methods with adaptive time steps. For example, we can use **ODE suits** in Matlab, and *dsode.f*, which is available through *Netlib*, in Fortran.

It is important to know that the ODE system from the MOL is typically *stiff* meaning that the eigenvalues of A has very different scales. For example, for the heat equation, the eigenvalues are between $O(1)$ and $O(1/h^2)$.

In Matlab, we can call the ODE solver using the format

```
[t,y] = ode23s('yfun-mol', [0, t_final], y0);
```

The solution in the last row of **y** which can be extracted using

```
[mr,nc] = size(y);
ysol = y(mr,:);
```

is the approximate solution at time $t = t\_final$. To define the ODE system of the MOL, we should create a Matlab file, yfun-mol.m whose contends contain the following

```
function yp = yfun-mol(t,y)
global m h x
k = length(y); yp=size(k,1);
yp(1) = (-2*y(1) + y(2))/(h*h) + f(t,x(1)) + g1(t)/(h*h);
for i=2:m-2
   yp(i) = (y(i-1) -2*y(1) + y(2))/(h*h) + f(t,x(i));
end
yp(m-1) = (y(m-2) -2*y(m-1) )/(h*h) + f(t,x(i)) + g2(t)/(h*h);
```

where $g1(t)$ and $g2(t)$ are two matlab functions for the boundary condition at $x = a$ and $x = b$; and $f(t, x(i)$ is the source term.

The initial condition can be defined as

```
global m h x
for i=1:m-1
    y0(i) = u_0(x(i));
end
```

where $u_0(x)$ is a Matlab function of the initial condition.

8.5 The Crank-Nicolson scheme.

The time step constraint $\Delta t = h^2/(2\beta)$ for the explicit Euler's method is generally considered as a severe restriction. If $h = 0.01$ and the final time is $T = 10$, and $\beta = 100$, we need 210^7 steps. The backward Euler's method does not have the time step constraint but it is only first order accurate. If we want second order accuracy $O(h^2)$, we need to take $\Delta t = O(h^2)$. Can we derive a finite difference scheme which is second order accurate both in time without compromise the stability and computational complexity? The answer is the Crank-Nicolson scheme.

The Crank-Nicolson scheme is based on the following lemma which can be proved easily using the Taylor expansion.

Lemma 8.1 *Let $\phi(t)$ be a function that has continuous first order derivative, that is $\phi(t) \in C^1$, then*

$$\phi(t) = \frac{1}{2}\left(\phi(t - \frac{\Delta t}{2}) + \phi(t + \frac{\Delta t}{2})\right) + \frac{(\Delta t)^2}{8}u''(t) + h.o.t. \tag{8.12}$$

The Crank-Nicolson scheme approximate the partial differential equation

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

at $(x_i, t^k + \Delta t/2)$ and use the averaging lemma above to approximate the spatial derivative $\nabla \cdot (\beta \nabla u))$ and $f(x,t)$. Therefore it has the following form

$$\begin{aligned}\frac{U_i^{k+1} - U_i^k}{\Delta t} &= \frac{\beta_{i-\frac{1}{2}}^k U_{i-1}^k - (\beta_{i-\frac{1}{2}}^k + \beta_{i+\frac{1}{2}}^k) U_i^k + \beta_{i+\frac{1}{2}}^k U_{i+1}^k}{2h^2} \\ &+ \frac{\beta_{i-\frac{1}{2}}^{k+1} U_{i-1}^{k+1} - (\beta_{i-\frac{1}{2}}^{k+1} + \beta_{i+\frac{1}{2}}^{k+1}) U_i^{k+1} + \beta_{i+\frac{1}{2}}^{k+1} U_{i+1}^{k+1}}{2h^2} + \frac{1}{2}\left(f_i^k + f_i^{k+1}\right).\end{aligned} \tag{8.13}$$

The discretization is second order in time and second in space. This can be easily proved using the following (we take $\beta = 1$ for simplicity of the proof):

$$\begin{aligned}&\frac{u(x,t+\Delta t) - u(x,t)}{\Delta} = u_t(x, t + \Delta/2) + \frac{1}{3}(\frac{\Delta t}{2})^2 + O((\Delta t)^4), \\ &\frac{u(x-h,t) - 2u(x,t) + u(x+h,t)}{2h^2} = u_{xx}(x,t) + O(h^2), \\ &\frac{u(x-h,t+\Delta t) - 2u(x,t+\Delta t) + u(x+h,t+\Delta t)}{2h^2} = u_{xx}(x,t+\Delta t) + O(h^2), \\ &\frac{1}{2}\Big(u_{xx}(x,t) + u_{xx}(x,t+\Delta t)\Big) = u_{xx}(x, t + \Delta t/2) + O((\Delta t)^2), \\ &\frac{1}{2}\Big(f(x,t) + f(x,t+\Delta t)\Big) = f(x, t + \Delta t/2) + O((\Delta t)^2).\end{aligned}$$

The Crank-Nicolson scheme is an implicit method. In the next section, we will prove that it is unconditional stable for the heat equation.

At each time step, we need to solve a tridiagonal system of equations to get u_i^{k+1}. The computational cost is only slightly more than the explicit Euler's method, we can take $\Delta t \sim h$ and have second order accuracy. It is much more efficient than the explicit Euler's method.

The θ-method.

The θ- method for the hear equation $u_t = u_{xx} + f(x,t)$ has the following form

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

When $\theta = 1$, the method is the explicit Euler's method. When $\theta = 0$, the method is the backward Euler's method. When $\theta = 1/2$, the method is the Crank-Nicolson scheme. If the $\theta \leq \frac{1}{2}$, then the method is unconditional stable, otherwise, it is conditional stable meaning there is a time step constraint. The θ-method is generally first order in time and second order in space except for $\theta = 1/2$.

8.6 Stability analysis for time-dependent problems–Discrete Fourier transform and von-Neumann analysis.

Let us first review Fourier transform (FT) in continuous space. Let $u(x) \in L^2(-\infty, \infty)$, that is $\int_{-\infty}^{\infty} u^2 dx < \infty$ or $\|u\|_2 < \infty$. The Fourier transform is defined as

$$\hat{u}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega x} u(x) dx, \tag{8.14}$$

where $i = \sqrt{-1}$. We can $u(x)$ is defined in the space domain while $\hat{u}(\omega)$ is defined in the frequency domain. Note that if a function is defined in the domain $(0, \infty)$, usually we can use the Laplace transform.

The inverse Fourier transform is defined as

$$u(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega x} \hat{u}(\omega) d\omega. \tag{8.15}$$

Parseval's relation: Under the Fourier transform, we have $\|\hat{u}\|_2 = \|u\|_2$ or

$$\int_{-\infty}^{\infty} |\hat{u}|^2 d\omega = \int_{-\infty}^{\infty} |u|^2 dx. \tag{8.16}$$

Fourier transform is a useful tool for theoretical and numerical analysis. Use the Fourier transform, we can get rid of one derivative.

From the definition of Fourier transform (FT), we have

$$\widehat{\left(\frac{\partial \hat{u}}{\partial \omega}\right)} = -ixu, \qquad \widehat{\frac{\partial u}{\partial x}} = i\omega \hat{u}. \tag{8.17}$$

To show the equalities above, we use the definition of the the inverse Fourier transform for $\frac{\partial u}{\partial x}(x)$ to get

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

On the other hand, since $u(x)$ and $\hat{u}(\omega)$ are both in $L^2(-\infty, \infty)$, we can take the partial derivative of the inverse Fourier transform with respect to x to get

$$\frac{\partial u}{\partial x}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \frac{\partial}{\partial x} \left(e^{i\omega x} \, \widehat{u}\right) d\omega = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} i\omega \widehat{u} e^{i\omega x} \, d\omega$$

Since the Fourier transform and its inverse transform are unique, we have $\widehat{\frac{\partial u}{\partial x}} = i\omega \widehat{u}$. The proof of the first equality will be left as an exercise. It is easy to generalize the equality to

$$\widehat{\frac{\partial^m u}{\partial x^m}} = (i\omega)^m \, \hat{u}. \tag{8.18}$$

In other words, we can get rid of the derivatives of one variable. The Fourier transform can be used to study the behavior of differential equations.

Example 1:

The wave equation

$$u_t + au_x = 0, \qquad -\infty < x < \infty, \quad t > 0, \quad u(x,0) = u_0(x).$$

If we apply the Fourier transform to the equation and the initial condition, we get

$$\widehat{u_t} + \widehat{au_x} = 0, \quad \text{or} \quad \widehat{u}_t + ai\omega\hat{u} = 0.$$

Therefore we get an ODE for $\hat{u}(\omega)$ which can be solved easily

$$\hat{u}(\omega,t) = \hat{u}(\omega,0)\, e^{-ia\omega t} = \hat{u}_0(\omega)\, e^{-ia\omega t}.$$

Therefore the solution to the original wave equation is

$$\begin{aligned} u(x,t) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega x}\, \hat{u}_0(\omega)\, e^{-ia\omega t}\, d\omega \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega(x-at)}\, \hat{u}_0(\omega)\, d\omega \\ &= u(x-at,0), \end{aligned}$$

according to the definition of the inverse Fourier transform. Thus we have solved the problem. The solution tell us that for a pure wave equation, the solution does not change shape but simply propagates along the characteristic line $x - at = 0$. Also note that

$$\|u\|_2 = \|\hat{u}\|_2 = \|\hat{u}(\omega,0)e^{-ia\omega t}\|_2 = \|\hat{u}(\omega,0)\|_2 = \|u_0\|_2$$

Example 2: The heat equation (diffusion equation).

Consider

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

We apply the Fourier transform to the equation and the initial condition to get

$$\widehat{u_t} = \widehat{\beta u_{xx}}, \quad \text{or} \quad \widehat{u}_t = \beta(i\omega)^2\hat{u} = -\beta\omega^2\hat{u}.$$

The solution of the ODE then is

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

Therefore

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

if $\beta > 0$. Actually, it can be shown that $\lim_{t\to\infty} \|u\|_2 = 0$ if $\beta > 0$, and $\lim_{t\to\infty} \|u\|_2 = \infty$ if $\beta < 0$ and $\|u_0\|_2 \neq 0$.

Example 3: Dispersive waves.

Consider

$$u_t = \frac{\partial^{2m+1} u}{\partial x^{2m+1}} + \frac{\partial^{2m} u}{\partial x^{2m}} + l.o.t.,$$

where m is a non-negative integer. For example $u_t = u_{xxx}$, we have

$$\widehat{u_t} = \widehat{\beta u_{xxx}}, \quad \text{or} \quad \widehat{u}_t = \beta(i\omega)^3 \hat{u} = -i\omega^3 \hat{u}.$$

The solution of the ODE then is

$$\hat{u}(\omega, t) = \hat{u}(\omega, 0)\, e^{-i\omega^3 t}.$$

Therefore

$$\|u\|_2 = \|\hat{u}\|_2 = \|\hat{u}(\omega, 0)\|_2 = \|u(\omega, 0)\|_2,$$

The solution to the original PDE can be expressed as

$$\begin{aligned} u(x,t) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega x}\, \hat{u}_0(\omega)\, e^{-i\omega^3 t}\, d\omega \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega(x-\omega^2 t)}\, \hat{u}_0(\omega)\, d\omega. \end{aligned}$$

It can be interpreted in the following way. The Fourier component with wave number ω is propagating with the velocity ω^2. Waves interact with each other, but there is no diffusion.

Example 4: The PDE with even order derivatives (highest).

Consider

$$u_t = \alpha \frac{\partial^{2m} u}{\partial x^{2m}} + \frac{\partial^{2m-1} u}{\partial x^{2m-1}} + l.o.t.,$$

where m is a non-negative integer. Then we know that

$$\widehat{u_t} = \alpha(i\omega)^{2m} \hat{u} + \cdots = \begin{cases} -\alpha\omega^{2m} \hat{u} + \cdots & \text{if } m = 2k+1 \\ \alpha\omega^{2m} \hat{u} + \cdots & \text{if } m = 2k. \end{cases}$$

Therefore we have

$$\widehat{u} = \begin{cases} \hat{u}(\omega, 0)\, e^{-\alpha i\omega^{2m} t} + \cdots & \text{if } m = 2k+1 \\ \hat{u}(\omega, 0)\, e^{\alpha i\omega^{2m} t} + \cdots & \text{if } m = 2k. \end{cases}$$

From the equality above, we can conclude that $u_t = u_{xx}$ and $u_t = -u_{xxxx}$ are dynamically stable, while $u_t = -u_{xx}$ and $u_t = u_{xxxx}$ are dynamically unstable.

8.7 Discrete Fourier transform.

The motivations to study a discrete Fourier transform include the stability analysis of finite difference schemes, data analysis in frequency domain, filtering techniques etc.

Definition of a grid function: Let

$$\cdots v_{-2}, v_{-1}, v_0, v_1, v_2, \cdots ,$$

be a continuous function $v(x)$ at $x_i = i * h$, the discrete Fourier transform is defined as

$$\hat{v}(\xi) = \frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} h\, e^{-i\xi jh}\, v_j. \tag{8.19}$$

Remark 8.3

- *The definition is a quadrature approximation to the continuous case, that is, we approximate $\int$ by $\sum$, and dx by h.*
- *$\hat{v}(\xi)$ is continuous, and periodic function of ξ with period $2\pi/h$:*

$$e^{-ijh(\xi+2\pi/h)} = e^{-ijh\xi} e^{2ij\pi} = e^{-i\xi jh}. \tag{8.20}$$

So we can focus on $\hat{v}(\xi)$ in the interval $[-\frac{\pi}{h}, -\frac{\pi}{h}]$.

Definition of discrete inverse Fourier transform:

$$v_j = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} e^{i\xi jh}\, \hat{v}(\xi)\, d\xi. \tag{8.21}$$

Definition of discrete Fourier and inverse Fourier transform for a finite sequence: (h is not involved):

$$\cdots, 0, 0, v_1, v_2, \cdots, v_M, 0, 0, \cdots$$

We define

$$\hat{v}(\xi) = \frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} e^{-i\xi j} v_j = \sum_{j=0}^{M} e^{-i\xi j}\, v_j \tag{8.22}$$

$$v_j = \frac{1}{\sqrt{2\pi}} \int_{-\pi}^{\pi} e^{i\xi j}\, \hat{v}(\xi)\, d\xi. \tag{8.23}$$

Discrete Norm:

$$\|\mathbf{v}\|_h = \sqrt{\sum_{j=-\infty}^{\infty} v_j^2\, h}, \tag{8.24}$$

it is often denoted as $\|v\|_2$ as well. The Parseval's relation is also true

$$\|\hat{v}\|_h^2 = \int_{-\pi/h}^{\pi/h} |\hat{v}(\xi)|^2 d\xi = \sum_{j=-\infty}^{\infty} |v_j|^2 h = \|v\|_h^2 \tag{8.25}$$

8.8 Definition of stability of a finite difference scheme.

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

$$\|\mathbf{v}^n\|_h \leq C_T \sum_{j=0}^{J} \|\mathbf{v}^j\|_h \tag{8.26}$$

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

Remark 8.4

1. *The stability is usually independent of the source terms.*
2. *A stable finite difference scheme means that the growth of the solution is at the most of a constant multiple of the sum of the norms of the solution at the first $J+1$ steps.*
3. *The stability region are all possible Δt and h.*

The following theorem provide a simple way to check the stability of a finite difference scheme.

Theorem 8.1 *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

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

So if we take $J = 0$, $C_T = 1$, we have the stability.

8.9 Von Neumann stability analysis of finite difference methods.

The von Neumann stability analysis of a finite difference scheme can be described as the following process:

Discrete scheme $\Longrightarrow$ discrete Fourier transform $\Longrightarrow$ growth factor $g(\xi) \Longrightarrow$ stability ($|g(\xi)| \leq 1$?) And the simplification of the von-Neumann analysis.

Example 1.

The forward Euler method (FW-CT) for the heat equation $u_t = \beta u_{xx}$ is:

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

From the discrete Fourier transform, we know

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

Similarly

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

Plugging these relations into the finite difference scheme, we obtain

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

On the other hand, according to the definition, we have

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

The discrete Fourier transform is unique which implies

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

where

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

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

Let us examine $|g(\xi)|$ now:

$$\begin{aligned} g(\xi) &= 1 + \mu\left(\cos(-\xi h) - i\sin(\xi h) - 2 + \cos(\xi h) + i\sin(\xi h)\right) \\ &= 1 + 2\mu\left(\cos(\xi h) - 1\right) = 1 - 4\mu \sin^2(\xi h)/2 \leq 1. \end{aligned}$$

But we need that $|g(\xi)| \leq 1$, or $-1 \leq g(\xi) \leq 1$. Note that

$$-1 \leq 1 - 4\mu \leq 1 - 4\mu \sin^2(\xi h)/2 = g(\xi) \leq 1.$$

So if we take $-1 \leq 1 - 4\mu$, then we can guarantee that $|g(\xi)| \leq 1$, so the stability. Therefore a sufficient condition for the stability of the forward Euler's method is

$$-1 \leq 1 - 4\mu, \quad \text{or} \quad 4\mu \leq 2, \quad \text{or} \quad \Delta t \leq \frac{h^2}{2\beta}, \tag{8.27}$$

while we can not claim what will happen if the condition is violated, it is a reasonable good upper bound for the stability.

Simplification of von Neumann stability analysis for one step time marching method.

Assume that we have a one step time marching method $\mathbf{U}^{k+1} = f(\mathbf{U}^k, \mathbf{U}^{k+1})$. We have the following theorem that tells the stability of a finite difference method.

Theorem 8.2 *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*

$$|g(\theta, \Delta t, h)| \leq 1 + K\Delta t \tag{8.28}$$

for all θ, $0 < k \leq k_0$, and $0 < h \leq h_0$. If $g(\theta, \Delta t, h)$ is independent of h and Δt, the stability condition (8.28) can be replaced with

$$|g(\theta| \leq 1. \tag{8.29}$$

This theorem shows that to determine the stability of a finite difference scheme we need consider only the amplification factor $g(h\xi) = g(\theta)$. This observation is due to von Neumann, and because of that, this analysis is usually called von Neumann analysis. We present some examples below.

We can follow the following steps for the von Neumann analysis:

1. Set $U_j^k = e^{ijh\xi}$. Plugging the expression into the finite difference scheme.
2. Express U_j^{k+1} as $U_j^{k+1} = g(\xi)e^{ijh\xi}$.
3. Solve $g(\xi)$ and determine whether or when $|g(\xi)| \leq 1$ for the stability.

Example 2.

The stability of backward Euler's method for the heat equation $u_t = \beta u_{xx}$ is:

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

Follow the procedure mentioned above, we have

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

We solve $g(\xi)$ to get

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

for any h and $\Delta t > 0$. Also it is obvious that $-1 < 0 \le g(\xi)$. Therefore, $|g(\xi)| \le 1$, and the backward Euler's method is unconditionally stable!

Example 3.

The Leapfrog scheme (two-stage method) for the heat equation $u_t = u_{xx}$ is

$$\frac{U_i^{k+1} - U_i^{k+1}}{2\Delta t} = \frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2}. \tag{8.30}$$

This method is unconditionally unstable! To show this, we follow the stability analysis. Note that we need to use $U_j^{k-1} = e^{ijh\xi}/g(\xi)$. We can get

$$\begin{aligned} g(\xi)e^{ijh\xi} &= \frac{1}{g(\xi)}e^{ijh\xi} + e^{i\xi jh}\left(\mu(e^{-i\xi h}-2+e^{i\xi h})\right) \\ &= \frac{1}{g(\xi)}e^{ijh\xi} - e^{ijh\xi}4\mu\sin^2(h\xi/2). \end{aligned}$$

We get a quadratic equation for $g(\xi)$.

$$(g(\xi))^2 + 4\mu\sin^2(h\xi/2)\, g(\xi) - 1 = 0. \tag{8.31}$$

The solution are

$$g(\xi) = -2\mu\sin^2(h\xi/2) \pm \sqrt{4\mu^2\sin^4(h\xi/2)+1}.$$

One of roots is

$$g(\xi) = -2\mu\sin^2(h\xi/2) - \sqrt{4\mu^2\sin^4(h\xi/2)+1}$$

whose magnitude $g(\xi) \ge 1$. While we do not have any conclusion about the stability of the method, the method is known to be a unstable method in the literature.

8.10 Finite difference methods and analysis for 2D parabolic equations.

The general form of equations is

$$u_t + a_1 u_x + a_2 u_y = (\beta u_x)_x + (\beta u_y)_y + \kappa u + f(x,y,t)$$

with boundary conditions and an initial condition. It can be written as

$$u_t = Lu + f$$

where L is the spatial differential operator. Therefore, the method of line (MOL) can be used if one can find a good ODE solver for stiff ODE systems. Note that the system is large ($O(m^2)$).

For simplicity, we discuss the heat equation $u_t = \nabla \cdot (\beta \nabla u) + f(x, y, t)$. We assume β is a constant. The simplest method is the forward Euler's method:

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

where $\mu = \beta \Delta t / h^2$. The method is first order in time and second order in space, and it is conditionally stable. The stability condition is

$$\Delta t \leq \frac{h^2}{4\beta}. \tag{8.32}$$

Note that factor of 4 instead of 2. To show this using von Neumann analysis with $f = 0$, we should set

$$u_{lj}^k = e^{i\,(lh_x\xi_1 + jh_y\xi_2)} = e^{i\,\xi\cdot\mathbf{x}}, \tag{8.33}$$

where $\xi = [\xi_1, \xi_2]^T$, $\mathbf{x} = [h_x l, h_y j]^T$.

$$U_{lj}^{k+1} = g(\xi_1, \xi_2) e^{i\,\xi\cdot\mathbf{x}}. \tag{8.34}$$

Substituting these expressions into the finite difference scheme, we can get

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

where we assume $h_x = h_y = h$ for simplicity. Therefore to get the stability, we enforce

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

The inequality from the very left is

$$\frac{4\Delta t \beta\, 2}{h^2} \leq 2, \quad \text{or} \quad \Delta t \leq \frac{h^2}{4\beta}.$$

Backward Euler's method (BW-CT).

The scheme can be written as

$$\frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta t} = \frac{U_{i-1,j}^{k+1} + U_{i+1,j}^{k+1} + U_{i,j-1}^{k+1} + U_{i,j+1}^{k+1} - 4U_{ij}^{k+1}}{h^2} + f_{ij}^{k+1}. \tag{8.35}$$

The method is first order in time and second order in space and it is unconditionally stable. The coefficient matrix for the unknown U_{ij}^{k+1} is a block tridiagonal and it is strictly row diagonally dominant if we use the natural row ordering.

Crank-Nicolson scheme (C-N).

The scheme can be written as

$$\begin{aligned}\frac{U_{ij}^{k+1}-U_{ij}^{k}}{\Delta t} &= \frac{1}{2}\left(\frac{U_{i-1,j}^{k+1}+U_{i+1,j}^{k+1}+U_{i,j-1}^{k+1}+U_{i,j+1}^{k+1}-4U_{ij}^{k+1}}{h^2}+f_{ij}^{k+1}\right.\\ &\left.+\frac{U_{i-1,j}^{k}+U_{i+1,j}^{k}+U_{i,j-1}^{k}+U_{i,j+1}^{k}-4U_{ij}^{k}}{h^2}+f_{ij}^{k}\right).\end{aligned} \tag{8.36}$$

Both the local truncation error and global error are $O\left((\Delta t)^2+h^2\right)$. The scheme is unconditionally stable for linear problems. However, we need to solve a system of equations whose coefficient matrix is a strictly row diagonally dominant and block tridiagonal matrix if we use the natural row ordering.

8.11 The alternating directional Implicit (ADI) method.

The ADI is one special case of *time splitting* or *fractional step* methods. The idea is to use implicit discretization in one direction while using explicit in another direction. For the heat equation $u_t = u_{xx} + u_{yy} + f(x,y,t)$, the ADI method is:

$$\begin{aligned}\frac{U_{ij}^{k+\frac{1}{2}}-U_{ij}^{k}}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}}-2U_{ij}^{k+\frac{1}{2}}+U_{i+1,j}^{k+\frac{1}{2}}}{h^2}+\frac{U_{i,j-1}^{k}-2U_{ij}^{k}+U_{i,j+1}^{k}}{h^2}+f_{ij}^{k+\frac{1}{2}},\\ \frac{U_{ij}^{k+1}-U_{ij}^{k+\frac{1}{2}}}{(\Delta t)/2} &= \frac{U_{i-1,j}^{k+\frac{1}{2}}-2U_{ij}^{k+\frac{1}{2}}+U_{i+1,j}^{k+\frac{1}{2}}}{h^2}+\frac{U_{i,j-1}^{k+1}-2U_{ij}^{k+1}+U_{i,j+1}^{k+1}}{h^2}+f_{ij}^{k+\frac{1}{2}}.\end{aligned} \tag{8.37}$$

The method is second order in time and space is $u(x,y,t) \in C^4$ in space. It is unconditionally stable for linear problems. We can use symbolic expressions to discuss the method. The method can be written as

$$\begin{aligned}U_{ij}^{k+\frac{1}{2}} &= U_{ij}^{k}+\frac{\Delta t}{2}\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}}+\frac{\Delta t}{2}\delta_{yy}^2 U_{ij}^{k}+\frac{\Delta t}{2}f_{ij}^{k+\frac{1}{2}}\\ U_{ij}^{k+1} &= U_{ij}^{k+\frac{1}{2}}+\frac{\Delta t}{2}\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}}+\frac{\Delta t}{2}\delta_{yy}^2 U_{ij}^{k+1}+\frac{\Delta t}{2}f_{ij}^{k+\frac{1}{2}}.\end{aligned} \tag{8.38}$$

In the matrix-vector form, if we move unknowns to the left hand side, then we get

$$\begin{aligned}\left(I-\frac{\Delta t}{2}D_x^2\right)\mathbf{U}^{k+\frac{1}{2}} &= \left(I+\frac{\Delta t}{2}D_y^2\right)\mathbf{U}^{k}+\frac{\Delta t}{2}\mathbf{F}^{k+\frac{1}{2}}\\ \left(I-\frac{\Delta t}{2}D_y^2\right)\mathbf{U}^{k+1} &= \left(I+\frac{\Delta t}{2}D_x^2\right)\mathbf{U}^{k+\frac{1}{2}}+\frac{\Delta t}{2}\mathbf{F}^{k+\frac{1}{2}}.\end{aligned} \tag{8.39}$$

The following derivation is to get a simple form for convenience of analysis. Solve for $\mathbf{U}^{k+\frac{1}{2}}$ to get

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

Plugging the expression into the second equation in the ADI method to get

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

We can go further to get:

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

Note that in the derivation, we have used the fact that

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

and other communitive operations.

Implementation:

We take advantages of the tridiagonal solver.

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

For a fixed j, we get a tridiagonal system of equations for $U_{1j}^{k+\frac{1}{2}}, U_{1j}^{k+\frac{1}{2}}, \cdots, U_{m-1,j}^{k+\frac{1}{2}}$ assuming a Dirichlet boundary condition at $x=a$ and $x=b$. The system of equations in the matrix-vector form is:

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

where

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

where $\mu = \frac{\beta \Delta t}{2h^2}$ and $f_i^{k+\frac{1}{2}} = f(x_i, t^{k+\frac{1}{2}})$. For each j, we need to solve a symmetric tridiagonal system of equations.

Pseudo code in Matlab.

```
for j = 2:n,                                  % Look for fixed y(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;
      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
      A(i-1,i-1) = 2/dt + 2/h1;
    end
    ut = A\b;                                 % Solve the diagonal matrix.

%-------------- loop in y -direction -------------------------------
```

```
for i = 2:m,
   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;
      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
      A(j-1,j-1) = 2/dt + 2/h1;              % Solve the system
    end
    ut = A\b;
```

8.11.1 Truncation error analysis.

If we add the two equations in (8.37) together, we get

$$\frac{U_{ij}^{k+1} - U_{ij}^{k}}{(\Delta t)/2} = 2\delta_{xx}^2 U_{ij}^{k+\frac{1}{2}} + \delta_{yy}^2 \left(U_{ij}^{k+1} + U_{ij}^{k} \right) + 2 f_{ij}^{k+\frac{1}{2}}. \tag{8.40}$$

If we subtract the first equation from the second equation in (8.37), we get

$$4U_{ij}^{k+\frac{1}{2}} = 2\left(U_{ij}^{k+1} + U_{ij}^{k} \right) - \Delta t \delta_{yy}^2 \left(U_{ij}^{k+1} - U_{ij}^{k} \right). \tag{8.41}$$

Plugging this into (8.40), we can get

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

Now we can see clearly that the discretization is second order accurate both in space and time, that is $T_{ij}^{k} = O((\Delta t)^2 + h^2)$.

8.11.2 The stability analysis.

We take $f = 0$ and set

$$U_{lj}^{k} = e^{-i(\xi_1 h_1 l + \xi_2 h_2 j)}, \qquad U_{lj}^{k+1} = g(\xi_1, \xi_2)\, e^{-i(\xi_1 h_1 l + \xi_2 h_2 j)}. \tag{8.42}$$

Using the operator form, we have:

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

or

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

After some manipulations, we can get

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

where $\mu=\frac{\Delta t}{2h^2}$ and we have set $h_1=h_2=h$ for simplicity. So we have $|g(\xi_1,\xi_2)|\le 1$ no matter what Δt and h are. Therefore the ADI method is unconditionally stable for linear heat equations.

8.12 An implicit-explicit mixed method for diffusion and and advection equations.

Consider the equation

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

It is not so easy to get second order implicit scheme that the coefficient matrix is diagonally dominant or symmetric positive/negative definite due to the advection term. One solution is to use implicit scheme for the diffusion term and an explicit scheme for the advection term. The scheme has the following form from time level t^k to t^{k+1}:

$$\frac{u^{k+1}-u^k}{\Delta t}+(\mathbf{w}\cdot\nabla_h u)^{k+\frac{1}{2}}=\frac{1}{2}\left((\nabla_h\cdot\beta\nabla_h u)^k+(\nabla_h\cdot\beta\nabla_h u)^{k+1}\right)+f^{k+\frac{1}{2}}, \tag{8.43}$$

where

$$(\mathbf{w}\cdot\nabla_h u)^{k+\frac{1}{2}}=\frac{3}{2}(\mathbf{w}\cdot\nabla_h u)^k-\frac{1}{2}(\mathbf{w}\cdot\nabla_h u)^{k-1}. \tag{8.44}$$

The time step constraint can be chosen as

$$\Delta t\le\frac{h}{2\|\mathbf{w}\|_2}. \tag{8.45}$$

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

$$(\nabla\cdot\beta\nabla u)^{k+1}-\frac{2u^{k+1}}{\Delta t}=-\frac{2u^k}{\Delta t}+2(\mathbf{w}\cdot\nabla u)^{k+\frac{1}{2}}+(\nabla\cdot\beta\nabla u)^k+2f^{k+\frac{1}{2}}. \tag{8.46}$$

We need $\mathbf{u}^1$ to get this method started. We can use the explicit Euler's method (FW-CT) to approximate $\mathbf{u}^1$. It should have no effect on the stability and global error $O((\Delta t)^2+h^2)$.

8.13 Solving elliptic PDEs using the numerical methods for parabolic PDEs.

We know that the steady state solution of a parabolic PDE is the solution of the corresponding elliptic PDE. For example, the steady state solution of the parabolic PDE:

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

is the solution to the elliptic PDE:

$$\nabla \cdot (\beta \nabla u) + \mathbf{w} \cdot u + \bar{f}(\mathbf{x}) = 0,$$

if the limit

$$\bar{f}(\mathbf{x}) = \lim_{t \to \infty} f(\mathbf{x}, t)$$

exists. The initial condition is irrevalent to the steady state solution. But the boundary condition is. There are some advantages of this approach especially for some non-linear problems in which the solution is not unique. Using this approach, we can control the variation in the intermediate solutions. The linear system of equations are more diagonally dominant. Since we only care about the steady state solution, we prefer to use implicit methods with large time steps. The accuracy in time in unimportant.