

MATH5311 - Advanced Numerical Methods I

Taught by Wang Xiao Ping

Notes by Aaron Wang

Contents

1	September 8th, 2020	3
1.1	Introduction	3
1.1.1	Numerical Differentiation	3
1.1.2	Numerical Integration	3
1.1.3	Simpson's Rule	4
1.2	Solution of ODE's	4
1.2.1	Solving Linear System	4
2	September 10th, 2020	4
2.1	Outline of the Course	4
2.2	Parabolic Equations in 1D	5
2.2.1	Explicit Scheme	5
2.2.2	Error Analysis for Explicit Scheme	6
3	September 15th, 2020	7
3.1	More on the Explicit Method for 1D Heat Equation	7
3.2	Fourier Analysis of Error	8
3.3	Implicit Scheme for $u_t = u_{xx}$	9
4	September 17th, 2020	10
4.1	Implicit Scheme for $u_t = u_{xx}$	10
4.2	Stability Analysis for Implicit Scheme	10
4.3	The θ -Method	11
5	September 22nd, 2020	12
5.1	Three Time Level Scheme	12
5.2	Boundary Conditions	13
6	September 24th, 2020	15
6.1	Parabolic Equation in Two and Three Dimensions	15
6.2	Implicit Scheme	17

7	September 29th, 2020	18
7.1	ADI Method	18
7.2	Summary of 2D Parabolic (Heat) Equation	20
7.3	Hyperbolic Equations in 1D	20
7.4	Method of Characteristics	21
8	October 6th, 2020	22
8.1	Explicit Scheme for Hyperbolic Equation	22
8.2	Lax-Wendroff Scheme	23
9	October 8th, 2020	25
9.1	Upwind and Lax-Wendroff Analysis	25
10	October 15th, 2020	27
10.1	Box Scheme	27
10.2	Leap Frog Scheme	29
10.3	Leap-Frog Scheme for Wave Equation	29
11	October 20th, 2020	31
11.1	Elliptic Equation	31
11.2	Finite Element Method	31
12	October 22th, 2020	34
12.1	Finite Element Method Cont.	34

1 September 8th, 2020

1.1 Introduction

1.1.1 Numerical Differentiation

Recall that the derivative is defined as:

$$u'(x_0) = \lim_{h \rightarrow 0} \frac{u(x_0 + h) - u(x_0)}{h}.$$

Thus we can approximate the difference

Definition 1.1. Central Difference

$$u_x(x_i) = \frac{u_{i+1} - u_{i-1}}{2h}.$$

Theorem 1.2

The central difference method is second order accurate.

Proof. Using Taylor Expansion, we have:

$$u_{i+1} = u_i + u_x h + \frac{1}{2} u_{xx} h^2 + \dots$$

□

Remark 1.3 — Note that if the derivative is

1.1.2 Numerical Integration

Numerical Integration can be approximation by the **Riemann Sum**

$$\int_a^b f(x) dx \approx \sum_{i=1}^N f(x_i) \Delta x_i.$$

We can also use the **Trapezoidal Rule**, as:

$$\int_{x_i}^{x_{i+1}} f(x) dx \approx \frac{h}{2} [f(x_i) + f(x_{i+1})].$$

Remark 1.4 — The above equation is approximating the curve in interval $[x_i, x_{i+1}]$ with a straight line

Thus if we add up over all intervals x_i , we'd get:

$$\int_a^b f(x) dx \approx \sum_{i=0}^{N-1} \frac{h}{2} [f(x_i) + f(x_{i+1})].$$

This is **second order accurate**.

1.1.3 Simpson's Rule

1.2 Solution of ODE's

Say we have an ODE:

$$\begin{cases} \dot{x} = f(x, y) \\ x(0) = x_0 \end{cases}.$$

One simple approximation is the **Euler method**:

$$\dot{x} \approx \frac{x^{n+1} - x^n}{\Delta t} = f(x^n, t_n).$$

with time step Δt and:

$$x^{n+1} = x^n + \Delta t f(x^n, t_n).$$

This is a first order accurate scheme.

Remark 1.5 — There are some disadvantages to this method, as Δt must be chosen carefully to be stable.

The Euler method can be modified to make higher order methods, such as the **Modified Euler**

$$\begin{cases} x^* = x^n + \Delta t f(x^n, t_n) \\ x^{n+1} = x^n + \frac{\Delta t}{2} (f(x^n, t_n) + f(x^*, t_{n+1})) \end{cases}.$$

Essentially, x^* is a predictor of the This is second order accurate.

Remark 1.6 — There is also a **Runge-Katta Method**, which is a 4th order method.

1.2.1 Solving Linear System

We have a linear system:

$$Ax = b.$$

We can solve this with a variety of methods, such as

- Gaussian Eliminations
- LU factorization
- Iterative methods

2 September 10th, 2020

2.1 Outline of the Course

1. Parabolic equations, 1D, 2D
2. Hyperbolic equation (1D)
3. Elliptical equation (2D)
4. Iterative methods for linear systems

2.2 Parabolic Equations in 1D

The model problem is the **heat equation** that describes heat conduction.

Definition 2.1. The **heat equation** in 1D is:

$$u_t = ku_{xx}, \quad t > 0, 0 < x < t$$

with boundary conditions

$$\begin{aligned} u(0, t) &= a, & u(1, t) &= b \\ u(x, 0) &= u_0(x). \end{aligned}$$

For simplicity, let's first consider $k = 1$, i.e. $u_t = u_{xx}$, and boundary conditions $u(0, t) = u(1, t) = 0$ the method will be the same.

This can be interrelated as considering a rod with unit length with some initial temperature distribution, and then observing the temperature over time.

Definition 2.2. The above heat equation has **Dirichlet boundary conditions**, meaning that they are fixed at the boundaries, i.e. specifying the temperature.

Definition 2.3. **Neumann boundary conditions** would be specifying the derivative at the boundaries, i.e.

$$u_x(0) = a, \quad u_x(1) = b.$$

Here a and b would be the heat flux at the boundaries.

2.2.1 Explicit Scheme

Like before let's discretized it by considering the uniform grid in space in time, with:

$$x_j = j\Delta x, \quad t_n = n\Delta t,$$

for $j = 0, 1, \dots, J$, $n = 0, 1, \dots$ and $\Delta x = \frac{1}{J}$. Thus we seek to approximate:

$$U_j^n \approx u(x_j, t_n).$$

The first approximation would be using an explicit method, with:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} \approx u_t(x_j, t_n).$$

meaning we use the forward difference for the time derivative. For space derivative, we have:

$$\frac{U_{j+1}^n + U_{j-1}^n - 2U_j^n}{(\Delta x)^2} \approx u_{xx}(x_j, t_n).$$

which is derived from the central difference.

Remark 2.4 — Remembering Taylor expansion, this is because:

$$\begin{aligned} u_{i+1} + u_{i-1} &= 2u_i + u_{xx}h^2 + \frac{2}{4!}u^{(4)}h^4 + \dots \\ \implies \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} &= u_{xx} + O(h^2). \end{aligned}$$

Using this, we can discretize the PDE with:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2}.$$

Now that we have a discrete finite difference formula, we can solve it for U_j^n :

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{(\Delta x)^2} (U_{j+1}^n - 2U_j^n + U_{j-1}^n).$$

Using the notation $\nu = \frac{\Delta t}{(\Delta x)^2}$, we have:

$$U_j^{n+1} = (1 - 2\nu)U_j^n + \nu (U_{j+1}^n + U_{j-1}^n) \quad (1)$$

This is an **explicit scheme**, as each value at time level t_{n+1} can be independently calculated from the values at time t_n . As such, we can start with $n = 0$ and calculate for each next value of n step by step.

2.2.2 Error Analysis for Explicit Scheme

Since we are using U_j^n to approximate $u(x_j, t_n)$, if we replace U_j^n by the exact solution, the truncation error would be:

$$T_j^n = \frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\Delta t} - \frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{(\Delta x)^2} \quad (2)$$

Since this is smooth, we can use the Taylor expansion, giving us:

$$\begin{aligned} &\approx \frac{u_t(x_j, t_n)\cancel{\Delta t} + \frac{1}{2}u_{tt}(x_j, t_n)(\cancel{\Delta t})^2}{\cancel{\Delta t}} - \frac{u_{xx}(\cancel{\Delta x})^2 + \frac{2}{4!}u_{xxxx}(\cancel{\Delta x})^4}{(\cancel{\Delta x})^2} \\ &= u_t + \frac{1}{2}u_{tt}(\Delta t) - u_{xx} - \frac{2}{4!}u_{xxxx}(\Delta x)^2 + \dots \\ &= \underbrace{u_t}_{=0} + \frac{1}{2}u_{tt}\Delta t - \frac{2}{4!}u_{xxxx}(\Delta x)^2 + \dots \\ &= \frac{1}{2}u_{tt}\Delta t - \frac{2}{4!}u_{xxxx}(\Delta x)^2 + \dots \end{aligned}$$

Note that:

$$T^n \rightarrow 0, \quad \text{as } \Delta t \rightarrow 0, \quad (\Delta x) \rightarrow 0.$$

Since the truncation error goes to zero, this scheme is consistent. Note that this scheme is first order accurate in time and second order accurate in space. Although this scheme is consistent, this is only a necessary condition for convergence.

Definition 2.5. The scheme is **convergent** if as $\Delta t \rightarrow 0$, $\Delta x \rightarrow 0$,

$$\forall (x^*, t^*): x_j \rightarrow x^* \text{ and } t_n \rightarrow t^* \text{ implies } U_j^t \rightarrow u(x^*, t^*)$$

Let $e_j^n = U_j^n - u(x_j, t_n)$, meaning that convergence is equivalent to $e_j^n \rightarrow 0$ as $\Delta t \rightarrow 0$ and $\Delta x \rightarrow 0$. Rearranging 1 and 2. We have:

$$\begin{aligned} e_j^{n+1} &= e_j^n + \nu (e_{j+1}^n - 2e_j^n + e_{j-1}^n) + T_j^n \Delta t \\ \implies e_j^{n+1} &= (1 - 2\nu)e_j^n + \nu e_{j+1}^n + \nu e_{j-1}^n + T_j^n \Delta t \\ \implies |e_j^{n+1}| &\leq |1 - 2\nu||e_j^n| + |\nu||e_{j+1}^n| + |\nu||e_{j-1}^n| + |T_j^n|\Delta t \\ \implies |e_j^{n+1}| &\leq |1 - 2\nu|E^n + |\nu|E^n + |\nu|E^n + |T_j^n|\Delta t \end{aligned}$$

where $E^n = \max_j \{|e_j^n|\}$. If $\nu \leq \frac{1}{2}$, we have:

$$\max_j \{|e_j^{n+1}|\} \leq |1 - 2\nu|E^n + 2|\nu|E^n + \tilde{T}\Delta t = E^n + \tilde{T}\Delta t.$$

where $\tilde{T} = \max_{j,n} \{|T_j^n|\}$. Now we have:

$$\begin{aligned} E^{n+1} &\leq E^n + \tilde{T}\Delta t \leq (E^{n-1} + \tilde{T}\Delta t) + \tilde{T}\Delta t \\ &\leq E^{n-1} + 2\tilde{T}\Delta t \\ &\leq \dots \\ &\leq \underbrace{E^0}_{=0} + n\tilde{T}\Delta t \\ &= t_F \tilde{T} \end{aligned}$$

where $t_F = n\Delta t$ which is a constant. Since $\tilde{T} \rightarrow 0$ as $\Delta x \rightarrow 0$ and $\Delta t \rightarrow 0$ because it is consistent, we have that E^n goes to zero. Note that this means that we have the condition that:

$$\nu = \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2} \implies 2\Delta t \leq (\Delta x)^2,$$

meaning that our scheme is convergent provided that $\nu \leq \frac{1}{2}$. Next class we will consider $\nu > \frac{1}{2}$ and show that it will not converge.

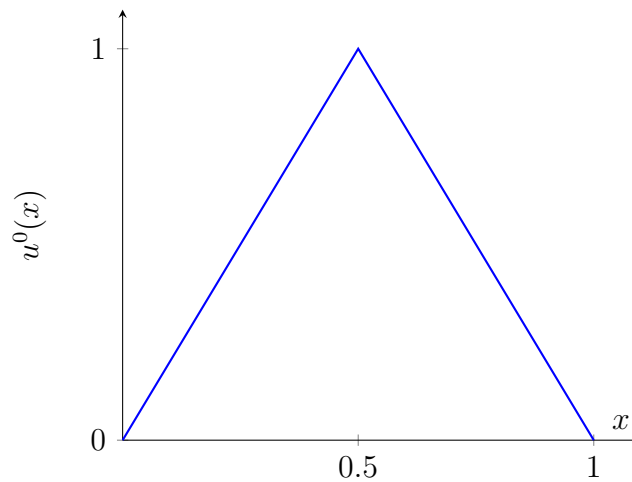
3 September 15th, 2020

3.1 More on the Explicit Method for 1D Heat Equation

Last time we showed an explicit method for evaluating u_t and u_{xx} for the 1D heat equation. It is a explicit scheme because each value at time level t_{n+1} can be independently calculated from values at time level t_n . We also showed that the scheme converges for $\nu \leq \frac{1}{2}$, with $\nu = \frac{\Delta t}{(\Delta x)^2}$.

Let's consider the example with initial conditions:

$$u^0(x) = \begin{cases} 2x, & 0 \leq x \leq \frac{1}{2} \\ 2 - 2x, & \frac{1}{2} \leq x \leq 1 \end{cases}.$$



Let's assume that the grid size is $\Delta x = 0.05$, i.e. there are 20 grid points in the space dimension. From the condition, we have:

$$\Delta t \leq \frac{1}{2} (\Delta x)^2 = 0.0125.$$

From this, let's consider the cases:

$$\Delta t = 0.0012 \implies \nu < \frac{1}{2} \text{ and } \Delta t = 0.0013 \implies \nu > \frac{1}{2}.$$

Running some MATLAB simulation reveals that U_j^n converges to 0 if $t = 0.0012$, but it does not for $t = 0.0013$. Thus, the condition $\nu \leq \frac{1}{2}$ is a stability condition.

3.2 Fourier Analysis of Error

Any smooth function can be expanded into a Fourier series:

$$f(x, t) = \sum_{n=-\infty}^{+\infty} a_n(t) e^{inx},$$

for some complex function $a_n(t)$ Where:

$$e^{inx} = \cos(nx) + i \sin(nx).$$

Thus we can write our equation as:

$$U_j^n = \lambda^n e^{ik(j\Delta x)}, \quad j = 0, 1, 2, \dots, N.$$

Remember our numerical explicit scheme is:

$$U_j^{n+1} = U_j^n + \nu(U_{j+1}^n - 2U_j^n + U_{j-1}^n), \quad \nu = \frac{\Delta t}{(\Delta x)^2}.$$

$$\implies \lambda^{n+1} e^{ik(j\Delta x)} = \lambda^n e^{ik(j\Delta x)} + \nu \lambda^n (e^{ik(j+1)\Delta x} - 2e^{ikj\Delta x} + e^{ik(j-1)\Delta x}).$$

$$\implies \lambda = 1 + \nu (e^{ik\Delta x} - 2 + e^{-ik\Delta x})$$

Using the fact that $e^{ik\Delta x} + e^{-ik\Delta x} = 2 \cos k\Delta x$, we have:

$$\implies \lambda = 1 + \nu (2 \cos(k\Delta x) - 2).$$

Using the double angle formula: $2(\cos k\Delta x - 1) = 2(-2 \sin^2 \frac{k\Delta x}{2})$, we have:

$$\lambda = 1 - 4\nu \sin^2 \frac{k\Delta x}{2}.$$

For the solution to be well behaved, we need $|\lambda| < 1$, since otherwise $|\lambda|^n$ will grow to infinity, i.e.:

$$\begin{aligned} \left| 1 - 4\nu \sin^2 \frac{k\Delta x}{2} \right| &\leq 1. \\ \implies 1 - 4\nu \sin^2 \frac{k\Delta x}{2} &\geq -1. \\ \nu \sin^2 \frac{k\Delta x}{2} &\leq \frac{1}{2}. \end{aligned}$$

which holds if $\nu \leq \frac{1}{2}$.

To reiterate, in order to ensure that the amplitude (λ) does not grow, we require $\nu \leq \frac{1}{2}$.

The method is stable if there exists a constant K independent of k , s.t.:

$$|\lambda^n| \leq K \text{ for } n\Delta t \leq t_F.$$

Definition 3.1 (Von Neumann Stability Condition). The method is stable if:

$$|\lambda k| \leq 1 + c\Delta t,$$

for some constant c and for all k .

Proof. Note that:

$$|\lambda|^n \leq (1 + c\Delta t)^n \leq \left(1 + \frac{ct_F}{n}\right)^n \leq e^{ct_F} = K.$$

Thus it is a necessary and sufficient condition for the convergence of a consistent difference scheme. \square

Essentially the Von Neumann condition is investigating the condition for converging amplitude of the Fourier series of the numerical scheme. As such, to find the stability, just assume the scheme has the form:

$$U_j^n = \lambda^n e^{ik(j\Delta x)}.$$

3.3 Implicit Scheme for $u_t = u_{xx}$

For this we use backward time difference:

$$\frac{U^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1}}{(\Delta x)^2}, \quad j = 1, 2, \dots, N+1.$$

This is an implicit scheme since it involves 3 unknown values of U on the new level $n+1$. This gives us $N-1$ equations and $N-1$ unknowns:

$$(1 + 2\nu)U_j^{n+1} - \nu U_{j+1}^{n+1} - \nu U_{j-1}^{n+1} = U_j^n, \quad j = 1, 2, \dots, N-1.$$

This gives a tridiagonal matrix:

$$AU = b.$$

with :

$$U = \begin{bmatrix} U_1^{n+1} \\ \vdots \\ U_{N-1}^{n+1} \end{bmatrix}, \quad b = \begin{bmatrix} U_1^n \\ \vdots \\ U_{N-1}^n \end{bmatrix}, \quad A = \begin{bmatrix} 1+2\nu & -\nu & 0 & & \\ -\nu & 1+2\nu & \ddots & & \\ 0 & \ddots & \ddots & \ddots & \\ & & & -\nu & 1+2\nu \end{bmatrix}.$$

4 September 17th, 2020

4.1 Implicit Scheme for $u_t = u_{xx}$

Recall that the implicit scheme is:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1}}{(\Delta x)^2}.$$

Note that when compared to the explicit scheme, the implicit scheme involves 3 unknown values of U on the new level $n+1$. This is in contrast to the explicit scheme, for which the values of U_j^{n+1} only depend on U^n . Thus there are $N-1$ unknowns: $U_1^{n+1}, U_2^{n+1}, \dots, U_{N-1}^{n+1}$, and $N-1$ equations:

$$(1 + 2\gamma)U_j^{n+1} - \gamma U_{j+1}^{n+1} - \gamma U_{j-1}^{n+1} = U_j^n.$$

This can be expressed as a linear system $AU = b$, with A being tridiagonal.

The simplest way to solve this linear system is Gaussian elimination, which for a tridiagonal matrix is similar to Thomas algorithm which solves the equation:

$$-a_j U_{j-1} + b_j U_j - c_j U_{j+1} = d_j, \quad j = 1, \dots, N-1.$$

While assuming diagonally dominance:

$$a_j > 0, b_j > 0, c_j > 0, \quad b_j > a_j + c_j.$$

Remark 4.1 — This diagonal dominance is to ensure there is a solution (not singular).

4.2 Stability Analysis for Implicit Scheme

Recall we are considering the equation:

$$\begin{cases} u_t = u_{xx} \\ u(0, t) = u(1, t) = 0 \\ u(x, 0) = u_0(x) \end{cases}.$$

Assuming we can do separation of variables, we have:

$$u(x, t) = Z(x) \cdot T(t).$$

Taking the Fourier series of the original equation, we have

$$\begin{aligned} u(x, t) &= \sum_{k=1}^{\infty} a_k(t) \sin k\pi x \\ \implies \sum_{k=1}^{\infty} a_k(t) \sin k\pi x &= - \sum_{k=1}^{\infty} a_k(t) (k\pi)^2 \sin k\pi x. \end{aligned}$$

Since $\sin k\pi x$ forms a basis, the coefficients must match, giving us:

$$\begin{aligned} a'_k(t) &= -(k\pi)^2 a_k(t) \\ \implies a_k(t) &= a_0 e^{-(k\pi)^2 t}. \end{aligned}$$

Note that the evolution of a_k is independent of other values of k . Thus in order to study how amplitude evolves with time, we don't need to look at the whole series, only how the amplitude decays with k . Thus for an exact solution of $u_t = u_{xx}$, we know that the amplitude decays exponentially fast.

For the discretized case, we want to see how the numeric scheme propagates the Fourier mode. Thus we let:

$$U_j^n = \lambda^n e^{ik(j\Delta x)}.$$

Plugging into the numerical implicit scheme, we have:

$$\begin{aligned} (1 + 2\nu)\lambda^{n+1} e^{ik(j\Delta x)} - \nu\lambda^{n+1} e^{ik(k+1)\Delta x} - \nu\lambda^{n+1} e^{ik(j-1)\Delta x} &= \lambda^n e^{ikj\Delta x}. \\ \implies \lambda [(1 + 2\nu) - \nu e^{ik\Delta x} - \nu e^{-ik\Delta x}] &= 1. \\ \implies \lambda (1 + 2\nu - 2\nu \cos k\Delta x) &= 1. \\ \implies \lambda \left(1 + 4\nu \sin^2 \frac{k\Delta x}{2} \right) &= 1. \\ \implies \lambda = \frac{1}{1 + 4\nu \sin^2 \frac{k\Delta x}{2}} < 1. \end{aligned}$$

Thus this implicit scheme unconditionally stable, meaning there is no condition on ν . Remember that for the explicit scheme, we needed the condition $\nu \leq \frac{1}{2}$.

4.3 The θ -Method

Recall we have learned two schemes:

- Explicit Scheme:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2}.$$

- Implicit Scheme:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1}}{(\Delta x)^2}.$$

Both schemes have first order error in time t and second order in space. This can be seen the truncation error T_j^n using taylor expansion.

Definition 4.2. The θ -method is a weighted average of explicit and implicit scheme. For the heat equation this is:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = (1 - \theta) \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2} + \theta \frac{U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1}}{(\Delta x)^2}, \quad 0 \leq \theta \leq 1.$$

Remark 4.3 — If $\theta = 0$, we have a explicit scheme, and if $\theta = 1$ we have the implicit scheme, both with 1st order in time and 2nd order in space.

However, if use $\theta = \frac{1}{2}$, we have 2nd order in time and space. This is because there is some cancellation when $\theta = \frac{1}{2}$. For any other values of θ , this will not be true. To calculate the truncation error for the θ method, we expand terms at $(x_j, t_{n+\frac{1}{2}})$:

$$\begin{aligned} u(x_j, t_n) &= u(x_j, t_{n+\frac{1}{2}}) - u_t\left(\frac{1}{2}\Delta t\right) + \frac{1}{2}u_{tt}\left(-\frac{1}{2}\Delta t\right)^2 \\ u(x_j, t_n) &= u(x_j, t_{n+\frac{1}{2}}) - u_t\left(\frac{1}{2}\Delta t\right) + \frac{1}{2}u_{tt}\left(-\frac{1}{2}\Delta t\right)^2 \end{aligned}$$

This gives truncation error:

$$\begin{aligned} T_j^{n+\frac{1}{2}} &= \underbrace{(u_t - u_{xx})}_{=0} + \left[\left(\frac{1}{2} - \theta\right)\Delta t u_{xxt} - \frac{1}{12}(\Delta x)^2 u_{xxxx} \right] + \frac{1}{4!} \left(\frac{1}{2} - \theta\right) \Delta t u_{xxxxt} (\Delta x)^2 \\ &\quad + O(\Delta t)^2 + O((\Delta x)^2) \end{aligned}$$

Note that when $\theta = \frac{1}{2}$, the truncation error is second order in both time and space. This is called the Crank-Nicolson scheme. Now the natural question is what is the stability of the this θ -method. We have:

- $0 \leq \theta \leq \frac{1}{2}$: stable $\iff \nu < \frac{1}{2}(1 - 2\theta)^{-1}$
- $\frac{1}{2} \leq \theta \leq 1$: stable for all ν

Thus the Crank-Nicolson scheme is unconditionally stable.

5 September 22nd, 2020

5.1 Three Time Level Scheme

Recalling the θ -method, when $\theta = \frac{1}{2}$, note that both the left and right hand sides are symmetric with respect to $n + \frac{1}{2}$, making it similar to the central difference. This is the intuition for why it has second order with respect to both time and space.

For the previous schemes, we use forward difference to expand u_t . Another way to express this is using the central difference instead:

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2}.$$

Note that both sides are symmetric with respect to n . This scheme is second order in both time and space, and is an explicit scheme. Note that it is a three-time level scheme, meaning we need the values at $n-1$ and n , and after that we can get the values for $n+1$.

Investigating the stability using Von Neumann stability analysis, we have:

$$\begin{aligned} U_j^n = \lambda^n e^{ik(j\Delta x)} &\implies \frac{\lambda - \lambda^{-1}}{2\Delta t} = \frac{-4 \sin^2(\frac{1}{2}k\Delta x)}{(\Delta x)^2}. \\ &\implies \lambda^2 + 8\lambda \mu \sin^2(\frac{1}{2}k\Delta x) - 1 = 0. \end{aligned}$$

This has two roots: λ_1, λ_2 with $\lambda_1 \cdot \lambda_2 = -1$. This means that $|\lambda_1| > 1$ or $|\lambda_2| > 1$, meaning that the scheme is always unstable. As such, this three level scheme is explicit and second order in both time and space, it is unstable.

Remark 5.1 — Note that since $n+1$ requires n and $n-1$, we'd have to use the forward difference explicit scheme for the first iteration, and then we can use the three time level scheme.

As a summary, we've investigated the following schemes:

- **Explicit Scheme**

- Advantage: simple and easy to implement
- Disadvantage: has a stability constraint of $\mu = \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$.

- **Implicit Scheme**

- Advantage: unconditionally stable, meaning there is no restriction on Δt
- Disadvantage: needs to solve a linear system at each time step

Remark 5.2 — In high dimensions, e.g. $d = 2, 3$ then implicit scheme is preferred, as it has a lower computational cost.

5.2 Boundary Conditions

Recall we are looking at the 1D heat equation:

$$u_t = u_{xx}, \quad 0 \leq x \leq 1.$$

Currently, we are solving with zero boundary conditions, i.e.:

$$u(0, t) = u(1, t) = 0.$$

and with initial conditions:

$$u(x, 0) = u_0(x).$$

Recall for the finite difference methods before, we only calculate for the interior points, $j = 1, 2, \dots, N - 1$. This is because for $j = 0$ and $j = N$, the solution is known (since it is the boundary condition).

Note that we do use the boundary data for $j = 1$ and $j = N - 1$, for example:

$$\frac{U_1^{n+1} - U_1^n}{\Delta t} = \frac{U_2^n + 2U_1^n + \cancel{U_0^n}}{(\Delta x)^2}.$$

If we instead consider the case where the boundary conditions are non zero but constant, i.e.:

$$u(0, t) = a, \quad u(1, t) = b, \quad a, b > 0.$$

Then, we can still calculate it, e.g.:

$$\frac{U_1^{n+1} - U_1^n}{\Delta t} = \frac{U_2^n + 2U_1^n + a}{(\Delta x)^2}.$$

Definition 5.3. The above boundary conditions are **Dirichlet boundary conditions**, as the function value is specified at the boundaries.

Definition 5.4. We can also have the **Neumann boundary conditions**, where we specify the derivative of the function at the boundaries

Example 5.5

An example of Neumann boundary condition would be

$$u_x(0, t) = 0, \quad u_x(1, t) = 0.$$

Remark 5.6 — The physical representation of Neumann boundary condition would be the heat flux, for example if $u_x(0, t) = u_x(1, t) = 0$, then there is no heat entering or exiting, meaning it is insulating.

Let us consider the insulating case with the explicit scheme:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2}.$$

In particular, consider $j = 1$, where:

$$\frac{U_1^{n+1} - U_1^n}{\Delta t} = \frac{U_2^n + 2U_1^n + U_0^n}{(\Delta x)^2}.$$

When $n = 0$, U_0^n is known (just the initial condition), but U_0^n is not known for $n = 1, 2, \dots$. To fix this, we can use finite difference to discretize the boundary conditions.

To use central difference at the boundaries, we can introduce a ghost point at $x_{-1} = -\Delta x$ and $x_{N+1} = 1 + \Delta x$, meaning that at $x = 0$:

$$x_0 = 0 \approx \frac{U_1 - U_{-1}}{2\Delta x}.$$

which has second order accuracy. Meanwhile at $x = 1$, we have:

$$u_x = 0 \approx \frac{U_{N+1} - U_{N-1}}{2\Delta x}.$$

Now instead of only evaluating interior points, we also evaluate at the boundary points.

In particular when $j = 0$, we have:

$$\frac{U_0^{n+1} - U_0^n}{\Delta t} = \frac{U_1^n - U_0^n + U_{-1}^n}{(\Delta x)^2}.$$

With left boundary condition:

$$\frac{U_1^{n+1} - U_{-1}^{n+1}}{2\Delta x} = 0.$$

Thus, we first update the solution

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{(\Delta x)^2} (U_{j+1}^n - 2U_j^n + U_{j-1}^n), \quad j = 0, 1, \dots, N.$$

then we update the boundaries:

$$\begin{aligned} U_{-1}^{n+1} &= U_1^{n+1}. \\ U_{N+1}^{n+1} &= U_{N-1}^{n+1}. \end{aligned}$$

We can do the same thing for the implicit scheme too. In summary, for Neumann boundary we need to introduce ghost points in order to discretize the boundary condition to maintain the second order accuracy.

6 September 24th, 2020

6.1 Parabolic Equation in Two and Three Dimensions

Now let's consider the heat equation in 2 dimensions:

$$\begin{cases} u_t = \sigma(u_{xx} + u_{yy}) = \sigma \Delta u \\ u|_{\partial\Omega} = f_0 \\ u(x, 0) = u_0(x) \end{cases} \quad \begin{array}{l} \text{(boundary condition)} \\ \text{(initial temperature distribution)} \end{array}.$$

Remark 6.1 — Δu is the Laplacian of u , and is $u_{xx} + u_{yy}$

To solve it numerically, we once again discretize it:

$$x_i = i\Delta x, \quad y_j = j\Delta y$$

for $0 \leq i \leq J_x$ and $0 \leq j \leq J_y$, with the step sizes Δx and Δy being the size of the corresponding domain divided by the number of grid points J_x and J_y . We once again have finite difference:

$$\begin{aligned}\frac{U_{i,j}^{n+1} - U_{i,j}^n}{\Delta t} &\approx u_t(x_i, y_j, t_n). \\ \frac{U_{i+1,j}^n + U_{i-1,j}^n - 2U_{i,j}^n}{(\Delta x)^2} &\approx u_{xx}(x_i, y_j, t_n). \\ \frac{U_{i,j+1}^n + U_{i,j-1}^n - 2U_{i,j}^n}{(\Delta y)^2} &\approx u_{yy}(x_i, y_j, t_n).\end{aligned}$$

Plugging into the heat equation, we have explicit scheme:

$$\begin{aligned}\frac{U_{i,j}^{n+1} - U_{i,j}^n}{\Delta t} &= \sigma \left(\frac{U_{i+1,j}^n + U_{i-1,j}^n - 2U_{i,j}^n}{(\Delta x)^2} + \frac{U_{i,j+1}^n + U_{i,j-1}^n - 2U_{i,j}^n}{(\Delta y)^2} \right). \\ \implies U_{i,j}^{n+1} &= \sigma \frac{\Delta t}{(\Delta x)^2} (U_{i+1,j}^n + U_{i-1,j}^n - 2U_{i,j}^n) + \sigma \frac{\Delta t}{(\Delta y)^2} (U_{i,j+1}^n + U_{i,j-1}^n - 2U_{i,j}^n) + U_{i,j}^n.\end{aligned}$$

Since this is explicit, we expect some condition on

$$\nu_x = \frac{\Delta t}{(\Delta x)^2}, \text{ and } \nu_y = \frac{\Delta t}{(\Delta y)^2}.$$

Note that this is also a consistent scheme.

Now calculating the Truncation error by Taylor expansion, we have:

$$T(x, t) = \frac{1}{2} \Delta t u_{xx} - \frac{1}{12} \sigma [(\Delta x)^2 u_{xxxx} + (\Delta y)^2 u_{yyyy}] + \dots$$

Note that $T_{ij}^n \rightarrow 0$ as $\Delta t \rightarrow 0$ and $(\Delta x, \Delta y) \rightarrow (0, 0)$. In addition, it is first order in time, and second order in space.

Theorem 6.2

The explicit scheme converges under the condition $\nu_x + \nu_y \leq \frac{1}{2}$.

Proof. Homework. □

Remark 6.3 — Note that if we let $\Delta x = \Delta y$, the above condition would be:

$$\frac{\Delta t}{(\Delta x)^2} + \frac{\Delta t}{(\Delta y)^2} \leq \frac{1}{2} \implies 2 \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2} \implies \Delta t \leq \frac{1}{4} (\Delta x)^2.$$

This means that in 2D, the condition for convergence is even more restricted than the 1D case. Because of this, in higher dimensions, we don't want to use explicit schemes.

Let's now consider this scheme using Von Neumann stability analysis, for $\sigma = 1$, we have:

$$\implies U_{i,j}^{n+1} = U_{i,j}^n + \frac{\Delta t}{(\Delta x)^2} (U_{i+1,j}^n + U_{i-1,j}^n - 2U_{i,j}^n) + \frac{\Delta t}{(\Delta y)^2} (U_{i,j+1}^n + U_{i,j-1}^n - 2U_{i,j}^n).$$

We have:

$$U_{ij}^n \sim \lambda^n e^{i(k_x x_i + k_y y_j)}.$$

Substituting, we have:

$$\begin{aligned} \lambda^{n+1} e^{i(k_x x_i + k_y y_j)} &= \lambda^n e^{i(k_x x_i + k_y y_j)} + \nu_x \lambda^n (e^{ik_x x_{i+1}} - 2e^{ik_x x_i} + e^{ik_x x_{i-1}}) e^{ik_y y_j} \\ &\quad + \nu_y \lambda^n (e^{ik_y y_{j+1}} - 2e^{ik_y y_j} + e^{ik_y y_{j-1}}) e^{ik_x x_i}. \\ &= \lambda^n e^{i(k_x x_i + k_y y_j)} + \nu_x \lambda^n (e^{ik_x \Delta x} - 2 + e^{-ik_x \Delta x}) e^{i(k_x x_i + k_y y_j)} + \nu_y \lambda^n (e^{ik_y \Delta y} - 2 + e^{-ik_y \Delta y}) e^{i(k_x x_i + k_y y_j)}. \\ &\implies \lambda = 1 + \nu_x \left(-4 \sin^2 \frac{k_x \Delta x}{2} \right) + \nu_y \left(-4 \sin^2 \frac{k_y \Delta y}{2} \right). \end{aligned}$$

Let us consider the case where $\Delta x = \Delta y$, meaning that $\nu_x = \nu_y$ and $k_x = k_y$, thus we have:

$$\lambda = 1 + 2\nu_x \left(-4 \sin^2 \frac{k_x \Delta x}{2} \right) = 1 - 8\nu_x \sin^2 \frac{k_x \Delta x}{2} \leq 1.$$

Since we want $|\lambda| \leq 1$, we need:

$$1 - 8\nu_x \sin^2 \frac{k_x \Delta x}{2} \geq -1 \implies \nu_x \sin^2 \frac{k_x \Delta x}{2} \leq \frac{1}{4}.$$

If $\Delta x \neq \Delta y$, we can still do it, but just a bit more complicated.

6.2 Implicit Scheme

The implicit scheme of the 2D heat equation is of the form:

$$\implies U_{i,j}^{n+1} = U_{i,j}^n + \frac{\Delta t}{(\Delta x)^2} (U_{i+1,j}^{n+1} + U_{i-1,j}^{n+1} - 2U_{i,j}^{n+1}) + \frac{\Delta t}{(\Delta y)^2} (U_{i,j+1}^{n+1} + U_{i,j-1}^{n+1} - 2U_{i,j}^{n+1}).$$

To solve this we need to solve a linear system. Since we discretize $i = 1, 2, \dots, J_x - 1$ and $j = 1, 2, \dots, J_y - 1$, we have a total of $(J_x - 1) \times (J_y - 1) = M$ equations, with the same number of unknowns. If we write this linear system in a matrix form, we have:

$$Ax = b$$

Where

$$A = \begin{bmatrix} D & L & & \\ L & D & \ddots & \\ & \ddots & \ddots & L \\ & & L & D \end{bmatrix}_{M \times M}$$

With

$$A = \begin{bmatrix} 1 + 2\nu_x + 2\nu_y & -\nu_x & & \\ -\nu_x & 1 + 2\nu_x + 2\nu_y & \ddots & \\ & \ddots & \ddots & -\nu_x \\ & & -\nu_x & 1 + 2\nu_x + 2\nu_y \end{bmatrix}$$

and

$$L = \begin{bmatrix} -\nu_y & & & \\ & \ddots & & \\ & & \ddots & \\ & & & -\nu_y \end{bmatrix}$$

Remark 6.4 — A is a symmetric-positive definite matrix.

Remark 6.5 — The form of matrix A , D , and L will depend on how you order x .

Rearrange the implicit scheme, we have:

$$-\nu_y U_{i,j-1}^{n+1} - \nu_x U_{i-1,j}^{n+1} + (1 - 2\nu_x - 2\nu_y)U_{i,j}^{n+1} - \nu_x U_{i+1,j}^{n+1} - \nu_y U_{i,j+1}^{n+1} = U_{i,j}^n.$$

To order x , we will first order them by y and then x , i.e.:

$$x = \begin{bmatrix} U_{11} \\ U_{21} \\ U_{31} \\ \vdots \\ U_{12} \\ U_{22} \\ \vdots \end{bmatrix}.$$

Note that x is a vector of lengths $(J_x - 1) \times (J_y - 1)$, as we only consider the interior points. Similar to the 1D case, the benefit of this implicit scheme is stability.

Remark 6.6 — If we use gaussian elimination to solve this linear system, the computation cost would be $O((M \times M)^2)$, which is unfeasible, especially for large M .

7 September 29th, 2020

7.1 ADI Method

The **ADI Method** or Alternative Directional Implicit Method has two steps

- Step 1: solve for $u_{ij}^{n+\frac{1}{2}}$ with

$$\frac{U_{i,j}^{n+\frac{1}{2}} - U_{i,j}^n}{\Delta t/2} = \frac{U_{i+1,j}^{n+\frac{1}{2}} + U_{i-1,j}^{n+\frac{1}{2}} - 2U_{i,j}^{n+\frac{1}{2}}}{(\Delta x)^2} + \frac{U_{i,j+1}^{n+\frac{1}{2}} + U_{i,j-1}^{n+\frac{1}{2}} - 2U_{i,j}^{n+\frac{1}{2}}}{(\Delta y)^2}.$$

- Step 2: solve for u_{ij}^{n+1} with

$$\frac{U_{i,j}^{n+1} - U_{i,j}^{n+\frac{1}{2}}}{\Delta t/2} = \frac{U_{i+1,j}^{n+\frac{1}{2}} + U_{i-1,j}^{n+\frac{1}{2}} - 2U_{i,j}^{n+\frac{1}{2}}}{(\Delta x)^2} + \frac{U_{i,j+1}^n + U_{i,j-1}^n - 2U_{i,j}^n}{(\Delta y)^2}.$$

In both steps, we only need to solve a tridiagonal system.

Remark 7.1 — Note that Step 1 is explicit in x , where as Step 2 is implicit in y . As such, instead of having to solve a huge system with 5 diagonals (which is not easy to solve), we only need to solve two tridiagonal systems.

If we use δ_x^2 and δ_y^2 to denote the finite difference in x and y , i.e.:

$$\delta_x^2 U = \frac{U_{i+1,j} + U_{i-1,j} - 2U_{i,j}}{(\Delta x)^2}.$$

$$\delta_y^2 U = \frac{U_{i,j+1} + U_{i,j-1} - 2U_{i,j}}{(\Delta y)^2}.$$

and let

$$\nu_x = \frac{\Delta t}{(\Delta x)^2}, \quad \nu_y = \frac{\Delta t}{(\Delta y)^2}.$$

we can write the ADI method as:

$$\begin{aligned} & \begin{cases} U^{n+1} - U^n = \frac{1}{2}\nu_x \delta_x^2 U^{n+\frac{1}{2}} + \frac{1}{2}\nu_y \delta_y^2 U^n \\ U^{n+1} - U^{n+\frac{1}{2}} = \frac{1}{2}\nu_x \delta_x^2 U^{n+\frac{1}{2}} + \frac{1}{2}\nu_y \delta_y^2 U^{n+\frac{1}{2}} \end{cases} \\ & \Rightarrow \begin{cases} (1 - \frac{1}{2}\nu_x \delta_x^2) U^{n+\frac{1}{2}} = (1 + \frac{1}{2}\nu_y \delta_y^2) U^n \\ (1 - \frac{1}{2}\nu_y \delta_y^2) U^{n+\frac{1}{2}} = (1 + \frac{1}{2}\nu_x \delta_x^2) U^{n+\frac{1}{2}} \end{cases} \\ & U^{n+1} = (1 - \frac{1}{2}\nu_y \delta_y^2)^{-1} (1 + \frac{1}{2}\nu_x \delta_x^2) \underbrace{(1 - \frac{1}{2}\nu_x \delta_x^2)^{-1} (1 + \frac{1}{2}\nu_y \delta_y^2) U^n}_{U^{n+\frac{1}{2}}}. \end{aligned}$$

which is in a matrix form.

Remark 7.2 — The truncation error of the ADI method is second order in both time and space.

Proof. Homework. □

To verify the stability, we once again use Von Neumann stability analysis with:

$$U^n \sim \lambda^n \exp(ik_x x_i + jk_y y_j), \quad x_i = i\Delta x, y_j = j\Delta y.$$

Solving, we would get:

$$\lambda = \frac{(1 - 2\nu_x \sin^2(\frac{1}{2}k_x \Delta x))(1 - 2\nu_y \sin^2(\frac{1}{2}k_y \Delta y))}{(1 + 2\nu_x \sin^2(\frac{1}{2}k_x \Delta x))(1 + 2\nu_y \sin^2(\frac{1}{2}k_y \Delta y))} = \frac{(1 - a)(1 - b)}{(1 + a)(1 + b)}.$$

where:

$$a = 2\nu_x \sin^2(\frac{1}{2}k_x \Delta x) \geq 0 \quad b = 2\nu_y \sin^2(\frac{1}{2}k_y \Delta y) \geq 0.$$

We need to show that $|\lambda| \leq 1$, and as such:

$$\begin{aligned} |\lambda| &= \left| \frac{(1 - a)(1 - b)}{(1 + a)(1 + b)} \right| = \left| \frac{1 - a}{1 + a} \right| \cdot \left| \frac{1 - b}{1 + b} \right|. \\ &= \frac{(1 - a)^2}{(1 + a)^2} = \frac{1 - 2a + a^2}{1 + 2a + a^2} \leq 1. \end{aligned}$$

Thus $|\lambda| \leq 1$ meaning that the ADI method is unconditionally stable.

7.2 Summary of 2D Parabolic (Heat) Equation

- The case for 2D heat equation is quite similar to 1D but with higher computational cost.
- We can still achieve second order accuracy both in time and space by using the ADI method, which combines implicit and explicit methods.
- The analysis is quite similar.

Remark 7.3 — From the analysis of PDE's the solution to parabolic PDEs are smooth. By smooth, this means that it is infinitely differentiable, which is needed for finite difference to work well since Taylor expansion is the basis of finite difference. If the solution is not smooth, then the solution can not be approximated by the Taylor expansion, thus the finite difference methods do not work. We will see that this is the case for hyperbolic equations.

7.3 Hyperbolic Equations in 1D

For the heat equation, if the boundary conditions are fixed at 0, then the whole function will eventually go to zero and reach a steady state. However this is different for hyperbolic equations. Let us consider the simplest hyperbolic equation:

$$\begin{cases} \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \\ u(x, 0) = f(x) \end{cases}.$$

This has a solution

$$u(x, t) = f(x - at).$$

Proof. We have:

$$u_t = f' \cdot (-a) \text{ and } u_x = f' \cdot (1).$$

Adding this up, we have:

$$u_t + au_x = -af' + af' = 0.$$

which is the equation. □

Remark 7.4 — This is called the travelling wave solution, since it is the initial condition $f(x)$ shifted by at . The speed is a .

Remark 7.5 — If $a > 0$ the wave will shift to the right, but if $a < 0$ it will shift to the left.

Unlike the heat equation which describes the diffusion process, the hyperbolic (transport) equation describes wave propagation.

7.4 Method of Characteristics

To solve the hyperbolic equation, we can use the method of characteristic. The characteristic line of the equation above is:

$$\frac{dx}{dt} = a \implies x = at + x_0.$$

Moving along the characteristic line, we have:

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} \cdot \frac{dx}{dt} = \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0.$$

This means that u is constant along the characteristic line:

$$u(x, t) = u_0(x_0) = u_0(x - at) = f(x - at) = f(x_0).$$

Generalizing this to multidimensional systems, we have:

$$\begin{cases} \frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(f(u, v)) = 0 \\ \frac{\partial v}{\partial t} + \frac{\partial}{\partial x}(g(u, v)) = 0 \end{cases}.$$

Which in vector form is:

$$\begin{bmatrix} u \\ v \end{bmatrix}_t = \begin{bmatrix} f \\ g \end{bmatrix}_x.$$

We can also write this as:

$$\begin{cases} \frac{\partial}{\partial x}(f(u, v)) = \frac{\partial f}{\partial u}u_x + \frac{\partial f}{\partial v}v_x \\ \frac{\partial}{\partial x}(g(u, v)) = \frac{\partial g}{\partial u}u_x + \frac{\partial g}{\partial v}v_x \end{cases} \implies \begin{bmatrix} \frac{\partial f}{\partial u} & \frac{\partial f}{\partial v} \\ \frac{\partial g}{\partial u} & \frac{\partial g}{\partial v} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}.$$

As such, we have:

$$\begin{bmatrix} u \\ v \end{bmatrix}_t + A \begin{bmatrix} u \\ v \end{bmatrix}_x \quad A = \begin{bmatrix} \frac{\partial f}{\partial u} & \frac{\partial f}{\partial v} \\ \frac{\partial g}{\partial u} & \frac{\partial g}{\partial v} \end{bmatrix}.$$

This is hyperbolic if A has real eigenvalues and a full set of eigenfunctions, which means that A is diagonalizable:

$$A = S^{-1}\Lambda S.$$

Where Λ is a diagonal matrix and S is invertible.

Remark 7.6 — If A is a symmetric matrix, it will satisfy this property.

Plugging this into the original equation, we have:

$$Su_t + \Lambda Su_x = 0.$$

With this, we can define a **Riemann invariant**:

$$r = r(u) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}.$$

such that :

$$\begin{cases} r_t = Su_t \\ r_x = Su_x \end{cases} \implies r_t + \Lambda r_x = 0.$$

Which would allow us to decouple the equation into two transport equations:

$$\begin{cases} r_{1t} + \lambda_1 r_{1x} = 0 \\ r_{2t} + \lambda_2 r_{2x} = 0 \end{cases}.$$

Which leads to two characteristic lines:

$$\frac{dx}{dt} = \lambda_1 \text{ and } \frac{dx}{dt} = \lambda_2.$$

8 October 6th, 2020

8.1 Explicit Scheme for Hyperbolic Equation

Applying the explicit scheme method, we have:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + a \frac{U_{j+1}^n - U_j^n}{\Delta x} = 0.$$

If we set $\nu = \frac{a\Delta t}{\Delta x}$, we have:

$$U_j^{n+1} = U_j^n - \nu(U_{j+1}^n - U_j^n) = (1 + \nu)U_j^n - \nu U_{j+1}^n.$$

Definition 8.1 (CFL Condition). For a convergent scheme, the domain of dependence of the PDE must lie within the domain of dependence of the numerical scheme.

Example 8.2

The scheme above cannot converge for $a > 0$. This is because if $a > 0$, U_j^{n+1} depends on U_j^n , U_{j+1}^n , U_{j+1}^{n-1} , U_{j+2}^{n-1} , etc. However, the characteristic line $x = at + x_0$ does not lie in this domain, thus the scheme does not hold.

Example 8.3

If we instead use a backward difference:

$$U_j^{n+1} = U_j^n - \nu(U_j^n - U_{j-1}^n) = (1 + \nu)U_j^n - \nu U_{j-1}^n.$$

Then the characteristic line falls within the domain.

Example 8.4

For $a < 0$, we must have $a \frac{\Delta t}{\Delta x} \leq 1$, in order to have the characteristic line to lie within the domain.

If we attempt to use central difference, we can eliminate the sign condition by using a symmetric scheme in space:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + a \frac{U_{j+1}^n - U_{j-1}^n}{2\Delta x} = 0.$$

Note that for this scheme, CFL condition holds for $|a \frac{\Delta t}{\Delta x}| \leq 1$. However, if we look at this from stability analysis, we have:

$$\begin{aligned} \frac{\lambda - 1}{\Delta t} \lambda^n e^{ik(j\Delta x)} + \frac{a}{2\Delta x} \lambda^n e^{ik(j\Delta x)} (e^{ik\Delta x} - e^{-ik\Delta x}) &= 0. \\ \implies \lambda &= 1 - a \frac{\Delta t}{\Delta x} i \sin k\Delta x. \end{aligned}$$

which is complex. Thus we have:

$$|\lambda| = \sqrt{1 + a^2 \left(\frac{\Delta t}{\Delta x} \right)^2 \sin^2 k\Delta x} > 1.$$

meaning it does not converge. Thus we cannot central difference for an explicit scheme for hyperbolic equation. Instead, we have to use forward or backwards depending on the value of a :

$$U_j^{n+1} = \begin{cases} U_j^n - \nu(U_{j+1}^n - U_j^n) & a < 0 \\ U_j^n - \nu(U_j^n - U_{j-1}^n) & a > 0 \end{cases}.$$

which is called the **upwind scheme**.

Example 8.5

In laymen terms, if the wave is going forward ($a > 0$) we use backward difference, if the wave is going backward we use forward difference.

Analyzing the stability, for $a > 0$, (backward difference) we have:

$$\begin{aligned} \lambda &= 1 - \nu(1 - e^{-ik\Delta x}) \\ &= 1 - \nu(1 - (\cos k\Delta x - i \sin k\Delta x)) \\ &= 1 - \nu(1 - \cos k\Delta x) + i\nu \sin k\Delta x \\ \implies |\lambda|^2 &= (1 - \nu(1 - \cos k\Delta x))^2 + \nu^2 \sin^2 k\Delta x \\ &= 1 - 2\nu(1 - \nu)(1 - \cos k\Delta x) \\ &= 1 - 4\nu(1 - \nu) \sin^2 \left(\frac{1}{2} k\Delta x \right) < 1 \quad \text{if } \nu < 1. \end{aligned}$$

This matches the CFL condition.

Remark 8.6 — Since this only uses first order derivatives, this scheme is only first order in time and space.

Previously to get second order accuracy, we used central difference. However, as we showed above, this is not a stable scheme. To achieve this accuracy, we will use quadratic interpolation.

8.2 Lax-Wendroff Scheme

Suppose we want to find the value at P in Figure 1. Using the characteristic line, we have that $U(P) = U(Q)$, and as such, we want to find the value at Q . To do this, we

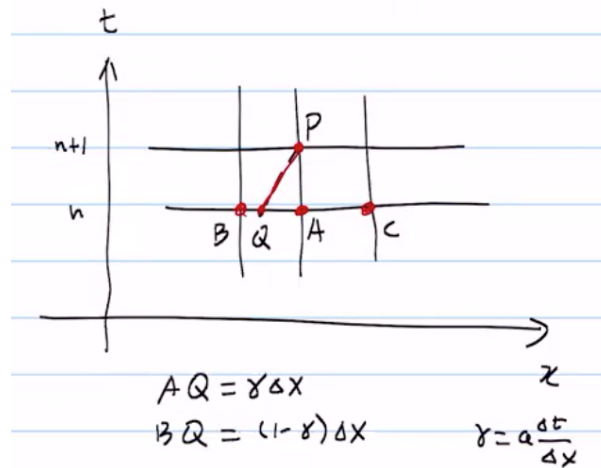


Figure 1

can try linear interpolation with points $B = U_{j-1}^n$, $A = U_j^n$, and $C = U_{j+1}^n$ that lie on the lattice:

$$\begin{aligned} U(Q) &= \nu U(B) + (1 - \nu)U(A) \\ \implies U_j^{n+1} &= \nu U_{j-1}^n + (1 - \nu)U_j^n. \end{aligned}$$

which is exactly the upwind scheme. If we try quadratic interpolation, we would have: In other words

$$P(x) = U_{j-1} \frac{(x - \Delta x)}{(0 - \Delta x)} \cdot \frac{(x - 2\Delta x)}{(0 - 2\Delta x)} + U_j \frac{(x - 0)}{(\Delta x - 0)} \cdot \frac{(x - 2\Delta x)}{(\Delta x - 2\Delta x)} + U_{j+1} \frac{(x - 0)}{(2\Delta x - 0)} \cdot \frac{(x - \Delta x)}{(2\Delta x - \Delta x)}.$$

Plugging in $Q = (1 - \nu)\Delta x$, we have:

$$\begin{aligned} P(Q) &= \frac{\nu}{2}(1 + \nu)U_{j-1} + (1 - \nu)(1 + \nu)U_j - \frac{1}{2}\nu(1 - \nu)U_{j+1} \\ \implies U_j^{n+1} &= U_j^n + \frac{\nu}{2}(U_{j-1}^n U_{j+1}^n) + \frac{\nu^2}{2}(U_{j-1}^n - 2U_j^n + U_{j+1}^n). \end{aligned}$$

Remember we need the characteristic line to lie within the scheme to satisfy the CFL condition, meaning we need $|\nu| \leq 1$. This scheme is called the **Lax-Wendroff method**.

Remark 8.7 — Note that this scheme is similar to the central difference but with a higher order correction term.

Remark 8.8 — The above scheme is second order in space. This can be verified by calculating the truncation error.

Checking the stability analysis, we would have:

$$\begin{aligned}
 \lambda &= \left(1 - 2\nu^2 \sin^2 \frac{k\Delta x}{2}\right)^2 + \nu^2 \sin^2 k\Delta x \\
 |\lambda|^2 &= \left(1 - 2\nu^2 \sin^2 \frac{k\Delta x}{2}\right)^2 + \nu^2 \sin^2 k\Delta x \\
 &= 1 - 4\nu^2 \sin^2 \frac{k\Delta x}{2} + 4\nu^4 \sin^4 \frac{k\Delta x}{2} + 4\nu^2 \sin^2 \frac{k\Delta x}{2} \cos^2 \frac{k\Delta x}{2} \\
 &= 1 - 4\nu^2(1 - \nu^2) \sin^4 \left(\frac{1}{2}k\Delta x\right) \leq 1 \quad \text{if } \nu \leq 1
 \end{aligned}$$

This means the scheme is stable for $|\nu| \leq 1$ (CFL condition).

To summarize, the design of numerical methods for hyperbolic equations is very different than for parabolic equations, as the issues we have to consider are very different.

9 October 8th, 2020

9.1 Upwind and Lax-Wendroff Analysis

The solutions to the upwind and Lax-Wendroff methods have different properties. For example:

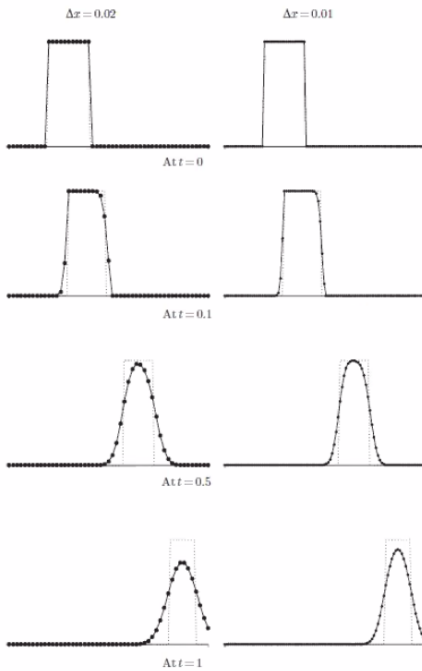


Fig. 4.6. Linear advection by the upwind method: problem (4.33), (4.34).

(a) Upwind Method

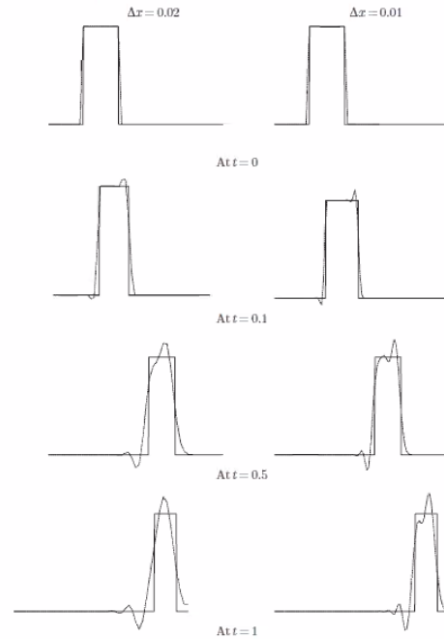


Fig. 4.7. Linear advection by the Lax-Wendroff method: problem (4.33), (4.34).

(b) Lax-Wendroff Method

Figure 2: Modified Equation Analysis

Remark 9.1 — The upwind does not have oscillation, but has smoothing and decaying effect. For the Lax-Wendroff, the result has an oscillation.

Let us look at upwind scheme for $a < 0$, we have:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = -a \frac{U_{j+1}^n - U_j^n}{\Delta x}.$$

Using Taylor expansion, we have:

$$\begin{aligned} u(x, t + \Delta t) &= u(x, t) + u_t(x, t)\Delta t + \frac{1}{2}u_{tt}(x, t)(\Delta t)^2 + \frac{1}{6}u_{ttt}(x, t)(\Delta t)^3 \\ u(x + \Delta x, t) &= u(x, t) + u_x(x, t)\Delta x + \frac{1}{2}u_{xx}(x, t)(\Delta x)^2 + \frac{1}{6}u_{xxx}(x, t)(\Delta x)^3 \end{aligned}$$

Thus, we have:

$$\begin{aligned} \frac{U_j^{n+1} - U_j^n}{\Delta t} &\approx u_t + \frac{1}{2}u_{tt}(\Delta t) + \dots \\ \frac{U_{j+1}^n - U_j^n}{\Delta x} &\approx u_x + \frac{1}{2}u_{xx}(\Delta x) + \dots \end{aligned}$$

Since $u_{tt} = -au_{tx}$, we have:

$$u_t = -au_x \quad u_{tt} = a^2u_{xx} \quad u_{ttt} = -a^3u_{xxx}.$$

Thus:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + a \frac{U_{j+1}^n - U_j^n}{\Delta x} \approx u_t + au_x + \frac{1}{2}(\Delta ta^2 + \Delta xa)u_{xx} = 0.$$

In other words, we are approximating:

$$u_t + au_x = \epsilon u_{xx}.$$

for some

$$\epsilon = -\frac{a\Delta x}{2} \left(a \frac{\Delta t}{\Delta x} + 1 \right) > 0.$$

Remark 9.2 — Note that this is similar to a parabolic equation with $u_t = \epsilon u_{xx}$. In other words, the effect of the leading order error is diffusion or damping, which is why the upwind has a diffusion behavior. Even though ϵ can be very small, it is non-zero, thus the diffusion effect is always there.

To see why, let's use Fourier analysis. Assume the solution of form:

$$u(x, t) = e^{i(kx - \omega t)}.$$

This is a wave traveling with a certain speed. We have:

$$\begin{aligned} u_t &= -i\omega e^{i(kx - \omega t)} \\ u_x &= ik e^{i(kx - \omega t)} \\ u_{xx} &= (ik)^2 e^{i(kx - \omega t)}. \end{aligned}$$

Plugging this into the equation, we have:

$$e^{i(kx-\omega t)} [-i\omega + aik] = e^{i(kx-\omega t)} [-\epsilon k^2].$$

Thus, we have:

$$\omega = ak - i\epsilon k^2.$$

This is a dispersion relation, since it is a relation between the frequency in time and the wave length. Plugging this into the exact solution, we have:

$$u(x, t) = e^{i(kx-akt+i\epsilon k^2 t)} = e^{ik(x-at)} e^{-\epsilon k^2 t}.$$

Note that the first factor is a traveling wave with speed a . The second term decays to 0 exponentially fast if $\epsilon > 0$, which is why the upwind method also has a decaying effect.

Remark 9.3 — This is called **modified equation analysis**, as we introduce error when we use finite difference. Therefore, instead of solving the original equation, we are solving a modified equation with some error. The form of this modified equation reveals the behavior of the numerical solution.

Looking at Lax-Wendroff, the modified equation is:

$$u_t + au_x = \frac{a}{6}(\Delta x)^2(\nu^2 - 1)u_{xxx} = \epsilon u_{xxx}.$$

This gives the leading order error a dispersion effect.

If we apply the same concept, we have $u_{xxx} = -ik^3 e^{i(kx-\omega t)}$. Plugging into the modified equation, we have:

$$\omega = ak + \epsilon k^3.$$

If we plug this into the exact solution, we would have a wave speed of $a + \epsilon k^2$ for $a > 0, \nu < 1, \epsilon < 0$. Note that this does not decay to 0 as in the case of the upwind scheme.

Remark 9.4 — $\epsilon < 0$ because $a > 0$ to satisfy CFL condition.

This means that the numerical wave speed will always be slower than the actual wave speed. The higher a wave number, the lower the wave speed. Because the speed of the higher frequency wave is slower than the original wave speed a , the oscillation is always behind.

Remark 9.5 — The reason why we don't see the oscillation in the exact solution is because the waves of each frequency travel together at the same speed. However, for the numerical scheme the speeds are different depending on the frequency.

10 October 15th, 2020

10.1 Box Scheme

From last time, we looked at the modified equations for the upwind scheme and the Lax-Wendroff schemes and saw that they showed different properties because of the leading

order error term. For upwind, this was u_{xx} , giving it a dispersive effect, while for Lax-Wendroff this was u_{xxx} , giving us an oscillatory effect.

Let us now consider the **Box Scheme**:

$$\frac{1}{2} \left(\frac{U_j^{n+1} - U_j^n}{\Delta t} + \frac{U_{j+1}^{n+1} - U_{j+1}^n}{\Delta t} \right) + \frac{a}{2} \left(\frac{U_{j+1}^n - U_j^n}{\Delta x} + \frac{U_{j+1}^{n+1} - U_j^{n+1}}{\Delta x} \right) = 0.$$

Remark 10.1 — This is essentially taking the average of the time and space derivatives.

If we let $\nu = \frac{\Delta t}{\Delta x}$, we have:

$$(1 - a\nu)U_j^{n+1} + (1 + a\nu)U_{j+1}^{n+1} = (1 + a\nu)U_j^n + (1 - a\nu)U_{j+1}^n.$$

Assume for $j = 0$, U_j^{n+1} is given for the boundary condition. If this is the case, then we have:

$$(1 + a\nu)U_{j+1}^{n+1} = -(1 - a\nu)U_j^{n+1} + (1 + a\nu)U_j^n + (1 - a\nu)U_{j+1}^n.$$

allowing us to solve for U_{j+1}^{n+1} recursively. If the boundary condition is given on the right boundary, i.e. U_N^{n+1} is given, then we can solve recursively from the right, as:

$$(1 - a\nu)U_j^{n+1} = -(1 + a\nu)U_{j+1}^{n+1} + (1 + a\nu)U_j^n + (1 - a\nu)U_{j+1}^n.$$

Remark 10.2 — Note that for a transport equation to be solved, we need to specify the left boundary condition $a > 0$ or the right boundary condition if $a < 0$.

For the stability, let us consider:

$$U_j^n \sim \lambda^n e^{ikj\Delta x}.$$

Plugging this into the equation, we have:

$$\begin{aligned} (1 - a\nu)\lambda^{n+1}e^{ikj\Delta x} &= -(1 + a\nu)\lambda^{n+1}e^{ik(j+1)\Delta x} + (1 + a\nu)\lambda^n e^{ikj\Delta x} + (1 - a\nu)\lambda^n e^{ik(j+1)\Delta x} \\ \implies (1 - a\nu)\lambda - (1 + a\nu)\lambda e^{ik\Delta x} &+ (1 + a\nu) + (1 - a\nu)e^{ik\Delta x} \\ \implies \lambda &= \frac{(1 + a\nu) + (1 - a\nu)e^{ik\Delta x}}{(1 + a\nu)e^{ik\Delta x} + (1 - a\nu)}. \end{aligned}$$

If we let $a_1 = 1 + a\nu$ and $a_2 = 1 - a\nu$, we have:

$$\begin{aligned} \lambda &= \frac{a_1 + a_2 e^{ik\Delta x}}{a_1 e^{ik\Delta x} + a_2} \\ \implies |\lambda| &= \left| \frac{a_1 + a_2 \cos x + ia_2 \sin x}{a_1 \cos x + ia_1 \sin x + a_2} \right| = \frac{(a_1 + a_2 \cos x)^2 + a_2^2 \sin^2 x}{(a_1 \cos x + a_2)^2 + a_1^2 \sin^2 x} \\ &= \frac{a_1^2 + 2a_1 a_2 \cos x + a_2^2}{a_1^2 + 2a_1 a_2 \cos x + a_2^2} = 1. \end{aligned}$$

and thus the box scheme is unconditionally stable.

Remark 10.3 — The box scheme is also second order in time and space.

Remark 10.4 — The box scheme has a phase advance error if $|a\nu| \leq 1$ and a phase lag error if $|a\nu| > 1$.

10.2 Leap Frog Scheme

The leap-frog scheme takes the central difference in both time and space. The equation we want to solve is:

$$u_t + f(a)_x = 0.$$

or

$$u_t + au_x = 0.$$

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} + \frac{f(U_{j+1}^n) - f(U_{j-1}^n)}{2\Delta x} = 0.$$

This has a CFL condition of $|\nu| < 1$ where $\nu = a \frac{\Delta t}{\Delta x}$.

Remark 10.5 — The leap frog scheme is second order in time and space, as it is using the central difference.

For the stability, we have:

$$\begin{aligned} \frac{1}{2\Delta t}(\lambda^{n+1} - \lambda^{n-1})e^{ikj\Delta x} + a \frac{1}{\Delta x}(e^{ik\Delta x} - e^{-ik\Delta x})\lambda^n e^{ikj\Delta x} &= 0 \\ \implies \left(\lambda - \frac{1}{\lambda}\right) + a \frac{\Delta t}{\Delta x}(2i \sin k\Delta x) &= 0 \\ \implies \lambda^2 + (2\nu i \sin k\Delta x)\lambda - 1 &= 0 \\ \implies \lambda = \frac{-2\nu i \sin k\Delta x \pm \sqrt{-4\nu^2 \sin^2 k\Delta x + 4}}{2} &= -\nu i \sin k\Delta x \pm \sqrt{1 - \nu^2 \sin^2 k\Delta x} \\ \implies |\lambda| = \nu^2 \sin^2 k\Delta x + 1 - \nu^2 \sin^2 k\Delta x &= 1. \end{aligned}$$

Thus the leap frog is stable if $|\nu| \leq 1$.

10.3 Leap-Frog Scheme for Wave Equation

Now let's consider the leap frog scheme for:

$$u_{tt} - a^2 u_{xx} = 0.$$

Remark 10.6 — This second order equation can be converted into a system of first order equation. Let $u_t = -av_x$. We have:

$$u_{tt} = -av_{tx} = a^2 u_{xx}.$$

Thus we can rewrite it as

$$\begin{cases} u_t + av_x = 0 \\ v_t + au_x = 0 \end{cases}.$$

To solve this, let U_j^n be defined on the grid points and $V_{j+\frac{1}{2}}^{n+\frac{1}{2}}$ be defined on the center of the grid squares. With this, we can apply the leap frog scheme:

$$\begin{aligned} \frac{U_j^{n+1} - U_j^n}{\Delta t} + a \frac{V_{j+\frac{1}{2}}^{n+\frac{1}{2}} - V_{j-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} &= 0. \\ \frac{V_{j+\frac{1}{2}}^{n+\frac{3}{2}} - V_{j+\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta t} + a \frac{U_{j+1}^{n+1} - U_j^{n+1}}{\Delta x} &= 0. \end{aligned}$$

Remark 10.7 — This also has second order time and space accuracy, since it is using central difference.

For the stability, let us construct a Fourier solution:

$$(U^n, V^{n+\frac{1}{2}}) = \lambda^n e^{ik\Delta x} (\hat{u}, \hat{v}).$$

where \hat{u}, \hat{v} are constants. This gives us:

$$\begin{aligned} \frac{\hat{u}(\lambda^{n+1} e^{ikj\Delta x} - \lambda^n e^{ikj\Delta x})}{\Delta t} + a \frac{\hat{v}(\lambda^n e^{ik(j+\frac{1}{2})\Delta x} - \lambda^n e^{ik(j-\frac{1}{2})\Delta x})}{\Delta x} &= 0. \\ \frac{\hat{v}(\lambda^{n+1} e^{ik(j+\frac{1}{2})\Delta x} - \lambda^n e^{ik(j+\frac{1}{2})\Delta x})}{\Delta t} + a \frac{\hat{u}(\lambda^{n+1} e^{ik(j+1)\Delta x} - \lambda^{n+1} e^{ikj\Delta x})}{\Delta x} &= 0. \\ \implies \begin{cases} \hat{u}(\lambda - 1) + \nu \hat{v}(2i \sin k \frac{\Delta x}{2}) = 0 \\ \hat{v}(\lambda - 1) e^{ik \frac{\Delta x}{2}} + \nu \hat{u} \lambda (e^{ik\Delta x} - 1) = 0 \end{cases} &. \\ \implies \begin{bmatrix} \lambda - 1 & 2i \sin \frac{k\Delta x}{2} \\ 2i\lambda \nu \sin \frac{k\Delta x}{2} & \lambda - 1 \end{bmatrix} \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} &= 0. \end{aligned}$$

for nontrivial solutions, this requires

$$\begin{aligned} (\lambda - 1)^2 + \lambda 4\nu^2 \sin^2 \frac{k\Delta x}{2} &= 0. \\ \implies \lambda^2 - 2(1 - 2\nu \sin^2 \frac{k\Delta x}{2})\lambda + 1 &= 0. \end{aligned}$$

solving this would give us $|\lambda| = 1$ which is stable.

Remark 10.8 — We can also do direct discretization of the scheme which is second order in time and space and is stable.

11 October 20th, 2020

11.1 Elliptic Equation

Let us consider a simple elliptic equation:

$$\begin{cases} -u'' = f \\ u(0) = u(1) = 0 \end{cases}.$$

This can be approximated by central difference

$$-\frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} = f_j.$$

where h is the step size this can be expressed in the form of a tridiagonal linear system. In 2D, we have a similar construction with:

$$\frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{(\Delta x)^2} + \frac{U_{i,j+1} - 2U_{i,j} + U_{i,j-1}}{(\Delta y)^2} = f_{i,j}.$$

This would give us a system of equation $AU = F$, with $(M-1) \times (N-1)$ unknowns and A being a matrix with 5 diagonals, similar to the 2D hyperbolic case.

Remark 11.1 — How to solve this linear system efficiently will be discussed in next semester.

11.2 Finite Element Method

Finite different methods need the domain to be a rectangular mesh, making it not effective for arbitrary domains. This is where **finite element method** comes in. The approach of this finite element method is very different from finite difference. In finite difference, we are discretizing the derivative. However, in finite element, we are discretizing the solution space.

Let us consider the same equation as before:

$$\begin{cases} \Delta u = f \\ u|_{\partial\Omega} = 0 \end{cases}.$$

Let us define a **functional**:

$$I(v) = \int_{\Omega} \left(\frac{1}{2} |\nabla v|^2 + f v \right) dx dy.$$

where ∇v is the **gradient** of v , i.e. $\nabla v = (\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y})$.

Remark 11.2 — This is a functional because the parameter v is a function.

We define this over the function space:

$$H_0^1(\Omega) = \{v | v, \nabla v \in L^2(\Omega), v|_{\partial\Omega} = 0\}.$$

Remark 11.3 — $H_0^1(\Omega)$ is set of functions that are square integrable (in L^2) and satisfy the boundary condition. Superscript 1 means that the first derivative is square integrable, and subscript 0 means it is zero at the boundary.

We would like to find the minimum over the function space:

$$\min_{v \in H_0^1(\Omega)} I(v).$$

In order to find the minimum, we want to find the derivative and set it to zero. This is a necessary condition.

Let's assume that function v_0 is a minimum of $I(v)$, meaning that:

$$I(v_0 + \epsilon u) \geq I(v_0), \quad \forall \epsilon, u \in H_0^1(\Omega).$$

Now we create a function:

$$J(\epsilon) = I(v_0 + \epsilon u).$$

where $\epsilon = 0$ is a minimum of $J(\epsilon)$, i.e. $J(\epsilon) \geq J(0)$ for all ϵ . This means that:

$$\left. \frac{dJ}{d\epsilon} \right|_{\epsilon=0} = 0.$$

From the definition of I , we have:

$$J(\epsilon) = I(v_0 + \epsilon u) = \int_{\Omega} \frac{1}{2} |\nabla v_0 + \epsilon \nabla u|^2 + f \cdot (v_0 + \epsilon u) dx dy.$$

Expanding, we have:

$$\int_{\Omega} \frac{1}{2} |\nabla v_0|^2 + \epsilon \nabla v_0 \nabla u + \frac{1}{2} \epsilon |\nabla u|^2 + \int_{\Omega} f(v_0) + \epsilon \int_{\Omega} f u.$$

Taking the derivative and evaluating at zero, we have:

$$\left. \frac{dJ}{d\epsilon} \right|_{\epsilon=0} = \int_{\Omega} \nabla v_0 \nabla u + \int_{\Omega} f u = 0.$$

From the divergence theorem:

$$\int_{\Omega} \operatorname{div} \vec{F}(x) dx = \int_{\partial\Omega} \vec{F} \cdot \vec{n} ds.$$

Taking $\vec{F} = v \nabla u$, we have:

$$\operatorname{div} \vec{F} = \nabla v \nabla u + v \Delta u.$$

Thus, we have:

$$\int_{\Omega} \operatorname{div} \vec{F}(x) dx = \int_{\Omega} \nabla v \nabla u + \int_{\Omega} v \Delta u = \int_{\partial\Omega} \vec{F} \cdot \vec{n} ds = 0.$$

because we are integrating over the boundary which is zero. Thus, we have:

$$\begin{aligned}\left.\frac{dJ}{d\epsilon}\right|_{\epsilon=0} &= \int_{\Omega} \nabla v_0 \nabla u + \int_{\Omega} f u = 0 \\ &= - \int_{\Omega} \Delta v_0 u + \int_{\Omega} f u = 0 \\ &= - \int_{\Omega} (\Delta v_0 - f) u \, dx dy = 0.\end{aligned}$$

Since u is arbitrary, we must have:

$$\Delta v_0 - f = 0.$$

As such:

$$\begin{cases} \Delta v_0 = f \\ v_0|_{\partial\Omega} = 0 \end{cases} \iff \int_{\Omega} \nabla v_0 \nabla u + \int_{\Omega} f u = 0 \quad \text{for all } u \in H_0^1(\Omega).$$

Definition 11.4. The right hand side is a **weak formulation** of the original PDE.

Remark 11.5 — If we have a minimum of the functional, then we have a solution to desired equation.

Remark 11.6 — This is called the weak formulation because we only involve the first derivative.

Let us assume there is a basis $\{\phi_1, \phi_2, \dots, \phi_N\}$ for $H_0^1(\Omega)$. Let us consider a finite dimensional subspace of $H_0^1(\Omega)$ (thus discretize the function space):

$$S_N = \{\phi_1, \phi_2, \dots, \phi_N\} \subset H_0^1(\Omega).$$

For any $v_0 \in S_N$, we can express it in terms of the basis functions:

$$v_0 = \sum_{i=1}^N v_i \phi_i.$$

Since the weak formulation must hold true for any $u \in S_N$, we can choose $u = \phi_j$:

$$\int_{\Omega} \nabla v_0 \nabla \phi_j + \int_{\Omega} f \phi_j = 0, \quad \text{for } j = 1, 2, \dots, N.$$

Expanding v_0 , we have:

$$\sum_i \int_{\Omega} v_i \nabla \phi_i \nabla \phi_j + \int_{\Omega} f \phi_j = 0.$$

Giving us a linear system $A\vec{v} = \vec{b}$ where:

$$A = (a_{ij}), \quad a_{ij} = \int_{\Omega} \nabla \phi_i \nabla \phi_j.$$

$$\vec{b} = (b_j), \quad b_j = \int_{\Omega} f \phi_j.$$

$$\vec{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix}.$$

12 October 22th, 2020

12.1 Finite Element Method Cont.

Recall that the finite element method transform the PDE into a equivalent weak formulation:

$$\int_{\Omega} \nabla v_0 \nabla u + \int_{\Omega} f u = 0 \quad \text{for all } u \in H_0^1(\Omega).$$

With this, we can discretize the function space $H_0^1(\Omega)$ by expressing it in terms of its basis and truncating it to only N basis functions. If we find a solution $v_0 \in S_N$ which satisfies this weak formulation, we would have:

$$\sum_i \int_{\Omega} v_i \nabla \phi_i \nabla \phi_j + \int_{\Omega} f \phi_j = 0.$$

Giving us a linear system $A\vec{v} = \vec{b}$ where:

$$A = (a_{ij}), \quad a_{ij} = \int_{\Omega} \nabla \phi_i \nabla \phi_j.$$

$$\vec{b} = (b_j), \quad b_j = \int_{\Omega} f \phi_j.$$

$$\vec{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix}.$$

Remark 12.1 — The resulting linear system of equations depends on the choice of the basis functions and the subspace S_N .

Remark 12.2 — Note that the solution is not a discretized solution since we are solving for the coefficients of the basis set. Also, note that this is an approximate solution because we did truncation on the number of basis functions.

With this, we would like to investigate the convergence of this method and the construction of basis functions.

Let us consider the 1D elliptical PDE:

$$\begin{cases} -u'' = f & 0 \leq x \leq 1 \\ u(0) = u(1) = 0 \end{cases}.$$

This has the corresponding weak formulation:

$$\text{Find } u \in H_0^1(\Omega) \text{ such that } \int_0^1 u' v' dx = \int_0^1 f v dx, \quad \forall v \in H_0^1(\Omega).$$

We have:

$$H_0^1(\Omega) = \{v | v, v' \in L^2, v(0) = v(1) = 0\}.$$

Now we would like to find a N dimensional subspace of this function space. To do this, let us consider S_N to be the set of continuous piecewise linear functions that vanishes at $x = 0, 1$.

Remark 12.3 — Piecewise linear means that the function is linear within the subinterval $[x_i, x_{i+1}]$.

Let us define a "hat" function at each grid point x_j :

$$\phi_j(x) = \begin{cases} 0 & x < x_{j-1} \vee x > x_{j+1} \\ \frac{x-x_{j-1}}{x_j-x_{j-1}} & x \in [x_{j-1}, x_j] \\ \frac{x_{j+1}-x}{x_j-x_{j+1}} & x \in [x_j, x_{j+1}] \end{cases}.$$

Claim 12.4. $\{\phi_1, \phi_2, \dots, \phi_N\}$ forms a basis of S_N .

Proof. To show that this is a basis, we need to show that they are linearly independent and that they are complete.

Independence: Suppose $c_1\phi_1 + c_2\phi_2 + \dots + c_N\phi_N = 0$. Consider at point x_j :

$$c_1\phi_1 + c_2\phi_2 + \dots + c_N\phi_N = c_j\phi_j(x_j) = 0 \implies c_j = 0.$$

this is true for all $j = 1, 2, \dots, N$, meaning that $\{\phi_j\}$ is linearly independent.

Complete: Let $v \in S_N$. Take $c_j = v(x_j)$:

$$\sum c_j\phi_j = \sum v(x_j)\phi_j(x) = v(x).$$

Since they are equal at all x_i and linear in $[x_i, x_{i+1}]$, they are equal everywhere. □

With these hat functions, we can write down the linear system:

$$A\vec{u} = \vec{b}.$$

where

$$a_{ij} = \int_0^1 \phi'_i \phi'_j dx, \quad \vec{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix}, \quad \vec{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}, \quad b_j = \int_0^1 f \phi_j dx.$$

Let us assume that we have a uniform grid $\{x_i\}$, where $x_i = i \cdot h$ and $h = \frac{1}{N}$. With this, we have:

$$\phi_i(x) = \begin{cases} 0 & x \leq x_{i-1} \\ \frac{x-x_{i-1}}{h} & x_{i-1} \leq x \leq x_i \\ -\frac{x-x_{i+1}}{h} & x_i \leq x \leq x_{i+1} \\ 0 & x \geq x_{i+1} \end{cases} \implies \phi'_i(x) = \begin{cases} 0 & x \leq x_{i-1} \\ \frac{1}{h} & x_{i-1} \leq x \leq x_i \\ -\frac{1}{h} & x_i \leq x \leq x_{i+1} \\ 0 & x \geq x_{i+1} \end{cases}.$$

Thus, we have:

$$a_{ij} = \int_0^1 \phi'_i \phi'_j dx.$$

Note that this is non-zero only when there is overlap between ϕ_i and ϕ_j . This only happens between ϕ_i and $\phi_i + 1$, giving us:

$$\begin{aligned} a_{i,i+1} &= \int_0^1 \phi_1' \phi_{i+1}' dx = \int_{x_i}^{x_{i+1}} \phi_i' \phi_{i+1}' dx = \int_{x_i}^{x_{i+1}} \left(-\frac{1}{h}\right) \left(\frac{1}{h}\right) dx = -\frac{1}{h^2} \cdot h = -\frac{1}{h} \\ a_{i,i-1} &= \int_0^1 \phi_1' \phi_{i-1}' dx = \int_{x_{i-1}}^{x_i} \phi_i' \phi_{i-1}' dx = \int_{x_{i-1}}^{x_i} \left(\frac{1}{h}\right) \left(-\frac{1}{h}\right) dx = -\frac{1}{h^2} \cdot h = -\frac{1}{h} \\ a_{i,i} &= \int_0^1 \phi_1' \phi_i' dx = \int_{x_{i-1}}^{x_{i+1}} \phi_i' \phi_i' dx = \int_{x_{i-1}}^{x_{i+1}} \left(\frac{1}{h}\right) \left(\frac{1}{h}\right) dx = \frac{1}{h^2} \cdot 2h = \frac{2}{h} \end{aligned}$$

Thus, we have:

$$A = \frac{1}{h} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 2 \end{bmatrix}.$$

For \vec{b} , we have $f_i \sim f(x_i)$:

$$b_i = \int_0^1 f \phi_i dx = \int_{x_{i-1}}^{x_{i+1}} f_i \phi_i dx = f_i \int_{x_{i-1}}^{x_{i+1}} \phi_i dx = f_i \frac{1}{2}(2h) = f_i h.$$

Thus, we have:

$$-\frac{1}{h^2} \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & & 1 & -2 \\ & & & & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_N \end{bmatrix}.$$

Remark 12.5 — This is very similar to the finite different scheme we saw before because we assume a uniform grid and use the hat functions for the basis functions. Since this is the same as the finite different method, it converges and is second order in both time and space.

Remark 12.6 — The order of accuracy depends on S_N , as if we instead used a quadratic piecewise function, we would have a different order error term.

Note that our finite element solution is not differentiable in the classical sense because there are corners. This leads to the idea of a weak derivative which will be introduced in the next class.