MA615 Numerical Methods for PDEs
Spring 2024 Lecture Notes

Xiangxiong Zhang
Department of Mathematics, Purdue University

Contents

Preface

These notes have been and will be evolving. A considerable amount of content consists of original discussions thus it is less likely to be flawless. I will correct them whenever possible. Even for the typo free part, please use it with caution.

1

Introduction

There are many different types of partial differential equations. A good choice of numerical schemes is often dependent on the type of equations, which is the key difficulty of studying numerical methods. For instance, successful and popular schemes for solving compressible flows are fundamentally different from the ones for solving incompressible flows in fluid dynamics.

1.1 Partial differential equations

Most of classical PDEs originate from modeling physical phenomenon, used in science and engineering problems. One thing we should always keep in mind is that these equations are chosen *models*, which are supposed to be valid, suitable or acceptable only under certain assumptions or only within certain context. For instance, compressible Navier-Stokes equations is a good continuum description of gas dynamics, if gas is not as rarefied as in a space shuttle entering the outer atmosphere.

In many applications, a PDE is a simplified approximated continuum modeling, as opposed to alternative particle models, e.g., the Boltzmann equation describes the statistical behaviour of a thermodynamic system, which can also be descirbed via molecular dynamics. PDEs have also been used for an efficient surrogate modeling of pedestrian flows or a flock of birds for which a particle model might seem more reasonable at least intuitively.

For beginners, equations can be assumed as given and *well-posed*, which roughly means that the equation has a unique nice solution. For a better understanding of the numerical methods, eventually one must understand the origin of the equation, which often plays a critical role in designing numerical schemes. Classical equations were mostly derived from *physical principles* (e.g., compressible Euler equations were derived from conservation of mass, momentum and energy) along with some empirical formula (e.g., equation of state for descrbing pressure dependence on mass, momentum

and energy). On the other hand, in practical applications, many *ad hoc* equations have been proposed and used. For example, if we know $u_t = u_x$ represents *convection*, $u_t = u_{xx}$ represents *dissipation* and $u_t = u_{xxx}$ represents *dispersion*, then it makes sense, at least seemingly, to use $u_t = au_x + bu_{xx} + cu_{xxx}$ as a model equation for modeling a system of convection-dissipation-dispersion. Nonetheless, a common practice does not necessarily mean that it is the right way.

1.2 Numerical schemes

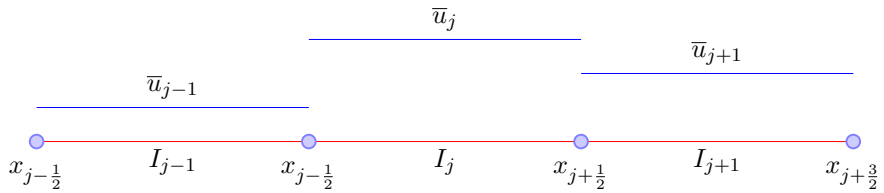
For PDEs, usually there are no exact solution formulae, and even if there is one, the formula can be demanding or dfficult to compute. One practial goal of numerical methods for PDEs is of course to provide a computationally *tractable* way for generating some kind of accurate approximations of the solution. Be aware that not all computational methods are *tractable* with given computational resources.

There are many popular numerical methods, which one may not have used but likely have heard of, such as *finite difference*, *finite element*, *finite volume* and *spectral methods*. As shown in Figure 1.1, approximations are obviously quite different in different numerical methods, which is however only a superficial way of understanding numerical schemes for PDEs. As a matter of fact, many of these different numerical methods can sometimes be regarded equivalent, especially for solving a one-dimensional problem.

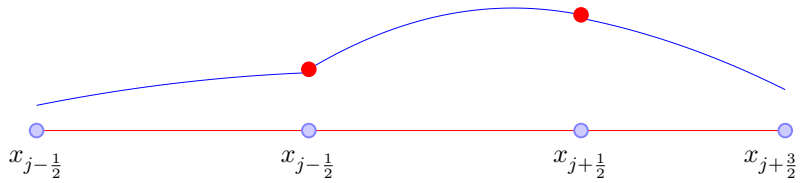
The key is not the difference in the choice of approximation methods, but rather the PDEs that one needs to solve. For certain types of PDEs such as wave equations $u_{tt} = \Delta u$, almost all kinds of numerical methods can be used to obtain a useful numerical scheme. For many other types of PDEs, it can be hard to use even a very popular numerical method. Even though the popular finite element methods are equipped with various software packages and the most complete and beautiful mathematical theory, there are equations and problems that they cannot handle. There is no single numerical method to serve as a silver bullet, unless one is content with solving only particular kinds of PDEs.

For example, finite volume schemes are successful for solving hyperbolic conservation laws and they are derived by discretizing the integral form of the conservation laws, and it is a perfectly natural thing to do because those PDEs are derived from the integral equations in the first place. On the other hand, it is very challenging to construct a scheme for hyperbolic conservation laws using spectral methods and continuous finite element methods. For conservation laws, there are other popular and useful schemes such finite difference WENO (weighted essentially non-oscillatory) methods and discontinuous Galerkin methods, all of which can be interpreted as some kind of finite volume scheme.

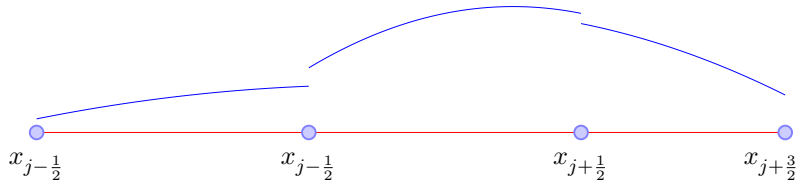
(a) Finite Difference



(b) Finite Volumne



(c) Continuous Finite Element



(d) Discontinuous Galerkin

Figure 1.1: An illustration of a few popular methods.

1.3 Computational tools

One particular emphasis of this lecture notes is the breadth of the scope. We will discuss a few different types of equations, thus different chapters might seem irrelevant to one another. The variety of different equations and different methods might seem overwhelming thus pose challenges, which however can become opportunities later because various methods provide ample inspiring perspectives of computational philosophy.

Many numerical methods go beyond solving PDEs. The simplest centered difference for Poisson's equation naturally extends to graph Laplacian on a graph. Numerical schemes for differential equations and numerical optimization algorithms are closely related. Many classical algorithms find roots in both territories. To name a few, the proximal point method for solving convex optimization, is nothing but backward Euler time discretization for numerical ODE. The most popular splitting method for convex composite optimization is called Douglas-Rachford and Peaceman-Rachford splitting, proposed by Lions and Mercier in 970s, which is also called ADI (alternating direction implicit) method for solving PDEs, originally designed for efficiently solving two-dimensionally heat equation in 1950s.

Put simply, methods in numerical PDEs are also useful tools for other modern computational tasks.

2

Finite difference methods for the Poisson's equation

We start with the Poisson's equation, one of the most popular linear PDEs, to understand basic concepts for numerical methods. The numerical method introduced in this chaper is the traditional or classical way of constructing a finite difference scheme, which thrived since 1950s (e.g., the book of Kantorovich and Krylov in Russian [5] and Collatz's book in German [2], originally published before 1960), and still a popular method nowadays [7].

The traditional finite difference method is easy to pick up for beginners, because only simple tools like calculus and linear algebra are needed. However, substantial difficulty will emerge if one tries to use such a numerical method for solving the Poisson's equation in a generic context, even on a rectangular domain, such as constructing a high order accurate scheme for solving a variable coefficient problem with Neumann boundary conditions.

Most of the numerical schemes in this chapter can be derived from the finite element methods on structured meshes with suitable numerical integration in Chapter 3. It has been well known that a finite element method can be equivalent to the traditional finite differen scheme since the finite element theory was born, e.g, , Kang Feng's first paper in Chinese in 1965 on finite element method was titled *Finite Difference Method Based on Variation Principles*. It has also been an effective approach to derive various finite difference schemes from different fintie element methods. On the other hand, such an equivalence is often overlooked in textbooks, resulting in a superficial impression that finite difference and finite element methods are completely different. To be precise, the traditional finite difference approach simply approximates the Poisson's equation directly, because of which most of its difficulties persist. Instead of approximating PDEs, the finite element approach approximates their equivalent variational formulation, which is the real and key difference here. However, the prerequites for fully understanding the finite element theory include functional analysis, which is probably

the reason why the traditional finite difference approach is still useful and interesting for rudimentary tasks, e.g., for teaching beginners without functional analysis background, or solving $-\Delta u = f$ on simple domains with only Dirichlet boundary conditions.

2.1 Finite difference approximations

For a smooth function $u(x)$, define the following finite difference operators approximating $u'(x)$ at the point $\bar{x}$:

- Forward Difference: $D_+u(\bar{x}) = \frac{u(\bar{x}+h)-u(\bar{x})}{h}$.
- Backward Difference: $D_-u(\bar{x}) = \frac{u(\bar{x})-u(\bar{x}-h)}{h}$.
- Centered Difference: $D_0u(\bar{x}) = \frac{u(\bar{x}+h)-u(\bar{x}-h)}{2h}$.

By Taylor expansion, the truncation errors of these operators are

$$D_\pm u(\bar{x}) = u'(\bar{x}) + \mathcal{O}(h), \quad D_0u(\bar{x}) = u'(\bar{x}) + \mathcal{O}(h^2).$$

Define $\hat{D}_0u(\bar{x}) = \frac{u(\bar{x}+h/2)-u(\bar{x}-h/2)}{h}$, then a classial second order finite difference approximation to $u''(x)$ at $\bar{x}$ is given by (denoted by D^2):

$$D^2u(\bar{x}) = D_+D_-u(\bar{x}) = \hat{D}_0\hat{D}_0u(\bar{x}) = \frac{u(\bar{x}+h) - 2u(\bar{x}) + u(\bar{x}-h)}{h^2} = u''(\bar{x}) + \mathcal{O}(h^2).$$

2.2 Poisson's equation

The Poisson's equations are

- 1D: $u''(x) = f(x)$
- 2D: $\Delta u(x, y) = u_{xx} + u_{yy} = f(x, y)$.
- 3D: $\Delta u(x, y, z) = f(x, y, z)$.

They are used and involved in many different contexts. To name a few,

- It is the steady-state equation of the heat equation $u_t = u_{xx} - f$.
- It is often involved in solving a time-dependent problem with a divergence free constraint. For example, in the incompressible Navier-Stokes equations

  $$\mathbf{u}_t + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \nu\Delta\mathbf{u} + \mathbf{f}, \quad \nabla \cdot \mathbf{u} = 0,$$

  take the divergence of the both sides in the momentum conservation equation, we get $\Delta p = \nabla \cdot (\nu\Delta\mathbf{u} + \mathbf{f} - (\mathbf{u} \cdot \nabla)\mathbf{u})$. By solving this equation, we get the pressure p from the velocity $\mathbf{u}$.

2.3 1D BVP: Dirichlet b.c.

Consider solving the 1D Poisson's equation with homogeneous Dirichlet boundary conditions:

$$\begin{cases} -u''(x) = f(x), & x \in (0,1), \\ u(0) = 0,\ u(1) = 0. \end{cases} \tag{2.1}$$

Discretize the domain $[0,1]$ by a uniform grid with spacing $h = \frac{1}{n+1}$ and n interior nodes: $x_j = jh$, $j = 1, 2, \cdots, n$. See Figure 2.1. Let $u(x)$ denote the true solution and $f_j = f(x_j)$. For convenience, define two ghost points $x_0 = 0$ and $x_{n+1} = 1$. Let u_j be the value of the numerical solution at x_j. Since two end values are given as $u(0) = 0, u(1) = 0$, only the interior point values $u_j (j = 1, \cdots, n)$ are unknowns. After approximating $\frac{d^2}{dx^2}$ by D^2, we get a finite difference scheme

$$-D^2 u_j = \frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = f_j, \quad j = 1, 2, \cdots, n \tag{2.2}$$

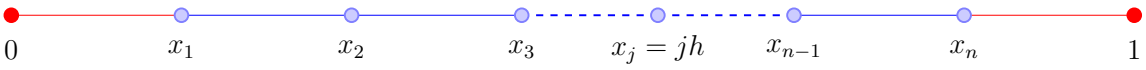


Figure 2.1: An illustration of the discretized domain.

Define

$$U_h = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \quad F = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}, \quad \widehat{U} = \begin{bmatrix} u(x_1) \\ u(x_2) \\ \vdots \\ u(x_n) \end{bmatrix}, \quad K = \begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}.$$

With the boundary values $u_0 = 0$ and $u_{n+1} = 0$ from the boundary condition, we can rewrite the finite difference scheme in the matrix vector form:

$$\frac{1}{h^2} K U_h = F.$$

2.3.1 Consistency, stability and convergence

- Local truncation error (LTE): the LTE is defined as the residue after replacing the numerical solution by the true soluion in the numerical scheme. For instance, the scheme (2.2) can be written as

$\frac{-u_{j-1}+2u_j-u_{j+1}}{h^2} - f_j = 0$. By the Taylor expansion of the exact solution $u(x)$ a x_j, the LTE of this scheme is

$$\begin{aligned}\tau_j &= \frac{-u(x_{j-1}) + 2u(x_j) - u(x_{j+1})}{h^2} - f_j \\ &= -u''(x_j) - \frac{1}{12}h^2 u''''(x_j) + \mathcal{O}(h^4) - f(x_j) \\ &= -\frac{1}{12}h^2 u''''(x_j) + \mathcal{O}(h^4) = \mathcal{O}(h^2),\end{aligned}$$

where $-u''(x_j) = f(x_j)$ is used. Denote $\tau_h = [\tau_1, \tau_2, \cdots, \tau_n]^T$.

- Consistency: if $\tau_h \to \mathbf{0}$ when $h \to 0$, we say the scheme is consistent.
- Global error: $E_h = \widehat{U} - U_h$ is the actual error of the scheme, defined as the global error. Let $A_h = \frac{1}{h^2}K$, then $\tau_h = A_h\widehat{U} - F = A_h\widehat{U} - A_hU_h = A_hE_h$ thus $E_h = A_h^{-1}\tau_h$ if A_h is invertible.
- Stability: we say the scheme $A_hU_h = F$ is stable if $\|A_h^{-1}\| \le C$ for small h, where $\|A\|$ denotes the spectral norm of the matrix A (i.e., the largest singular value of A, or equivalently $\|A\| = \max_{\mathbf{x}\in\mathbb{R}^n} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$ with $\|\mathbf{x}\|$ denoting the norm for vectors). If $\|A_h^{-1}\| \le C$, then $\|E_h\| = \|A_h^{-1}\tau_h\| \le \|A_h^{-1}\|\|\tau_h\| \le C\|\tau_h\|$.
- Convergence: if $\|E_h\| \to 0$ when $h \to 0$, we say the scheme is convergent. For this simple linear problem, we have

$$\text{Consistency} + \text{Stability} \to \text{Convergence}.$$

Remark 2.1. *For a normal matrix (normal means $A^*A = AA^*$ with A^* denoting conjugate transpose, e.g., real symmetric matrices, complex Hermitian matrices) A, $\|A\| = \max_i |\lambda_i|$ where λ_i are eigenvalues of A.*

2.3.2 Eigenvalues of K and stability in 2-norm

Since the eigenfunctions of $-u'' = \lambda u, u(0) = u(1) = 0$ are $\sin(m\pi x)$ for integers m, we expect that the eigenvectors of K would look like $\sin(m\pi x)$ for small h. With the following trigonometric formulas,

$$\sin(m\pi x_{j+1}) = \sin(m\pi(x_j+h)) = \sin(m\pi x_j)\cos(m\pi h) + \cos(m\pi x_j)\sin(m\pi h),$$

$$\sin(m\pi x_{j-1}) = \sin(m\pi(x_j-h)) = \sin(m\pi x_j)\cos(m\pi h) - \cos(m\pi x_j)\sin(m\pi h),$$

thus,

$$-\sin(m\pi x_{j-1}) + 2\sin(m\pi x_j) - \sin(m\pi x_{j+1}) = (2 - 2\cos(m\pi h))\sin(m\pi x_j).$$

Notice the facts that $\sin(m\pi x_0) = 0$ and $\sin(m\pi x_{n+1}) = 0$, we also have

$$2\sin(m\pi x_1) - \sin(m\pi x_2) = (2 - 2\cos(m\pi h))\sin(m\pi x_1),$$

$$-\sin(m\pi x_{n-1}) + 2\sin(m\pi x_n) = (2 - 2\cos(m\pi h))\sin(m\pi x_n).$$

Let $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$, then the eigenvectors of K are $\sin(m\pi\mathbf{x})$:

$$K\sin(m\pi\mathbf{x}) = (2 - 2\cos(m\pi h))\sin(m\pi\mathbf{x}), \quad m = 1, 2, \cdots, n.$$

Define the eigenvector matrix as $S = [\sin(\pi\mathbf{x}) \quad \sin(2\pi\mathbf{x}) \quad \cdots \quad \sin(n\pi\mathbf{x})]$ and consider the diagonal matrix Λ with diagonal entries $2-2\cos(m\pi h), m = 1, \cdots, n$. Then $K = S\Lambda S^{-1}$ and $A_h^{-1} = h^2 S\Lambda^{-1}S^{-1}$. By L'Hospital's rule we have $\frac{h^2}{2-2\cos(m\pi h)} \to \frac{1}{m^2\pi^2}, h \to 0$. Notice that $\frac{h^2}{2-2\cos(m\pi h)}$ is a monotonically increasing function of h. Therefore, we can conclude that $\|A_h^{-1}\| \le C$.

Problem 2.1. *Show that $\|A_h^{-1}\| \le C$ for any h, where C is a constant independent of h.*

So we obtain the global error E_h in vector 2-norm:

$$\|E_h\| \le \|A_h^{-1}\|\|\tau_h\| \le C\|\tau_h\|.$$

On the other hand, the standard vector norm $\|E_h\|$ is a not a good choice of measuring errors. For instance, assume $E_j = h^2$ for any h, then

$$\|E_h\| = \sqrt{\sum_{j=1}^{n} E_j^2} = \sqrt{\sum_{j=1}^{n} h^4} = \mathcal{O}(h^{1.5})$$

because of $h = \frac{1}{n+1}$. To this end, we define the 2-norm for errors:

$$\|E_h\|_2 = \sqrt{h\sum_{j=1}^{n} E_j^2}.$$

Remark 2.2. *The new 2-norm is a natural discrete version of the function L^2-norm. For instance, the L^2-norm of a single variable function $f(x)$ on an interval $x \in [0, 1]$ is defined as*

$$\|f\|_{L^2} = \sqrt{\int_0^1 f(x)^2 dx}.$$

If we discretize the inverval $x \in [0, 1]$ by grid points as in Figure 2.1, and use the simple approximation of integral (numerical integration, a.k.a, quadrature) hf_i for each small interval, then for a function $f(x)$ satisfying $f(0) = f(1) = 0$, we get $\sqrt{h\sum_{j=1}^{n} f(x_j)^2}$.

Notice that we do have

$$\|E_h\| \leq \|A_h^{-1}\| \cdot \|\tau_h\| \Rightarrow \|E_h\|_2 \leq \|A_h^{-1}\| \cdot \|\tau_h\|_2.$$

Assume that the one-dimensional boundary value problem (2.1) has a solution $u(x)$ with high order derivatives, e.g., if $f(x) = -g''(x)$ where $g(x)$ is a smooth function satisfying $g(0) = g(1) = 0$. Then we have the truncation error $\tau_j = \frac{1}{12}u''''(x_j)h^2 + \mathcal{O}(h^4) = \frac{1}{12}f''(x_j)h^2 + \mathcal{O}(h^4)$. Assume $f''(x)$ is bounded, then we simply have

$$\|\tau_h\|_2 = \mathcal{O}(h^2) \Rightarrow \|E_h\|_2 = \mathcal{O}(h^2).$$

Problem 2.2. *Consider the $n \times n$ matrix*

$$T(\alpha) = \begin{pmatrix} \alpha & -1 & & & & \\ -1 & \alpha & -1 & & & \\ & -1 & \alpha & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & \alpha & -1 \\ & & & & -1 & \alpha \end{pmatrix}.$$

Find its eigenvalues and eigenvectors. For what kind of α, this matrix will become positive definite?

2.3.3 Poisson's solver by eigenvectors

The matrix $S = [\sin(\pi \mathbf{x}) \quad \sin(2\pi \mathbf{x}) \quad \cdots \quad \sin(n\pi \mathbf{x})]$ is the Sine Matrix $S_{jk} = \sin(\frac{jk\pi}{n+1})$ thus mutliplying S to a vector $\mathbf{v}$ is equivalent to the discrete sine transform of $\mathbf{v}$ which can be computed with $\mathcal{O}(n \log_2 n)$ complexity. Mutliplying S^{-1} would be the inverse DST. For instance, the DST can be computed by the dst function in MATLAB:

```
% the grid points vector x has length n
dx=1/(n+1);
x=[0:dx:1];x=x(2:end-1);lambda=2*ones(n,1)-2*cos([1:n]*pi*dx);
v=sin(x*pi*[1:n]);
% verify the eigenvectors and eigenvalues of K
norm(K*v-v*diag(lambda)); % it should be machine zero,
                          % i.e., around 10^{-16}

% verify that multiplying the eigenvector matrix is
% equivalent to DST on a random vector b
a=randn(n,1);
b=K*a;
norm(a-dst(idst(b)./lambda)) % it should be machine zero,
                             % i.e., around 10^{-16}
```

Since K is a tridiagonal matrix, Gaussian elimination costs only $\mathcal{O}(n)$ which is faster than the DST. But in 2D and 3D, the eigenvector method would be advantageous.

2.3.4 Nonhomegeous Dirichlet b.c.

Consider solving the 1D Poisson's equation with nonhomogeneous Dirichlet boundary conditions:

$$\begin{cases} -u''(x) = f(x), & x \in (0,1), \\ u(0) = a,\ u(1) = b. \end{cases}$$

Then the scheme (2.2) can be changed as

$$\begin{aligned} \frac{2u_1 - u_2}{h^2} &= f_1 + a/h^2, \\ \frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} &= f_j, \quad j = 2, \cdots, n-1, \\ \frac{-u_{n-1} + 2u_n}{h^2} &= f_n + b/h^2. \end{aligned}$$

In other words, we can use the same coefficient matrix in the scheme for the nonhomogeneous b.c. with modified right hand side data compensating the nonzero boundary conditions. So from now on we just focus on the homogeneous case.

2.4 1D BVP: Dirichlet and Neumann b.c.

Consider solving the 1D Poisson's equation with homogeneous Dirichlet and Neumann boundary conditions:

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

Discretize the domain $[0,1]$ by a uniform grid with spacing h. For the interior nodes, we can use the same finite difference scheme (2.2).

2.4.1 A symmetric matrix T

Discretize the domain $[0,1]$ by a uniform grid with spacing $h = \frac{1}{n+1}$ and n interior nodes: $x_j = jh$, $j = 1, 2, \cdots, n$. See Figure 2.1. Let $x_0 = 0$ and $x_{n+1} = 1$.

For the boundary condition $u'(0) = 0$, we can first consider a simple first order approximation by forward difference:

$$\frac{u_1 - u_0}{h} \approx u'(x_0) \Rightarrow \frac{u_1 - u_0}{h} = 0 \Rightarrow u_0 = u_1,$$

thus the scheme (2.2) at x_1 becomes:

$$\frac{u_1 - u_2}{h^2} = f_1.$$

Then our scheme can be written as $\frac{1}{h^2}TU_h = F$ with

$$T = \begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}.$$

This scheme can be at most first order accurate in the local truncation error since forward difference is used for the boundary condition $u'(0) = 0$. Recall that the local truncation error is not the true error. Even though intuitively we expect first order accuracy for the convergence as well due to the first order local truncation error, it is possible to have a higher order convergence than the order of truncation error in boundary treatment.

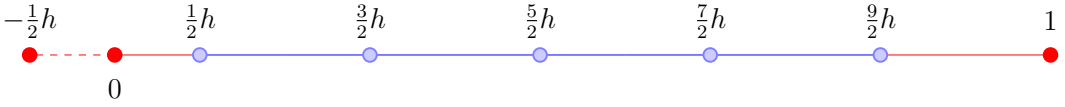


Figure 2.2: The grid.

By considering a new grid $x_j = (j - \frac{1}{2})h$ $(j = 1, 2, \cdots, n)$ with $h = \frac{1}{n+\frac{1}{2}}$, the $\frac{1}{h^2}TU_h = F$ becomes second order accurate. Let $x_0 = -\frac{1}{2}h$ then $\frac{u_1 - u_0}{h}$ is the centered difference approximating $u'(0)$, i.e.,

$$\frac{u_1 - u_0}{h} = u'(0) + \mathcal{O}(h^2) \Rightarrow \frac{u_1 - u_0}{h} = 0 \Rightarrow u_0 = u_1.$$

Let $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$ in this new grid, then the eigenvectors of T are $\cos[(m - \frac{1}{2})\pi\mathbf{x}]$ with eigenvalues $2 - 2\cos[(m - \frac{1}{2})\pi h]$, $m = 1, \cdots, n$.

2.4.2 Nonsymmetric matrix T_2

We can also construct a second order scheme on the integer grid $x_j = (j-1)h$ with $h = \frac{1}{n}$. Let $x_0 = -h$ then $\frac{u_2 - u_0}{2h}$ is the second order centered difference

approximation to $u'(0)$. We get a scheme $\frac{1}{h^2}T_2U_h = F$ with

$$T_2 = \begin{pmatrix} 2 & -2 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}.$$

Problem 2.3. *Find the eigenvalues and eigenvectors for T_2. Implement an eigenvector method for the scheme $\frac{1}{h^2}T_2U_h = F$ by using dct and idct functions in MATLAB.*

Unfortunately, T_2 is no longer symmetric, not even a normal matrix. But we can convert it to T by deviding the first equation by 2, then the scheme becomes

$$\frac{1}{h^2}TU_h = \begin{pmatrix} f_1/2 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} \tag{2.3}$$

However, the matrix $\frac{1}{h^2}T$ in the scheme (2.3) is not exactly the approximation to $-\frac{d^2}{dx^2}$ any more.

Problem 2.4. *Find the local truncation error of the scheme* (2.3) *at the left boundary.*

Problem 2.5. *Prove the stability and convergence of the scheme $\frac{1}{h^2}T_2U_h = F$. Notice that T_2 is not normal thus eigenvalues of T_2 do not give its spectral norm. Instead, try to eigenvalues of T and the inequality $\|AB\| \le \|A\|\|B\|$ to estimate $\|T_2^{-1}\|$.*

If we perform local truncation error at the first grid point, then we will find out that it is only first order even though it is obtained by a second order approximation to the Neumann boundary condition. So only first order convergence can be proven by estimating $\|T_2^{-1}\|$. On the other hand, the scheme is actually second order accurate not only in the discrete 2-norm but also in the maximum norm. See next subsection for explanation.

2.5 Convergence in maximum norm

For a vector $\mathbf{x} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$, its l^1-norm and l^∞-norm are defined as

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|, \quad \|\mathbf{x}\|_\infty = \max_i |x_i|.$$

For a matrix $A = [a_{ij}]$, its corresponding induced matrix norms are

$$\|A\|_1 = \max_{\mathbf{x}} \frac{\|A\mathbf{x}\|_1}{\|\mathbf{x}\|_1} = \max_j \sum_{i=1}^{n} |a_{ij}|,$$

$$\|A\|_\infty = \max_{\mathbf{x}} \frac{\|A\mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} = \max_i \sum_{j=1}^{n} |a_{ij}|.$$

For a numerical scheme in the form of $AU = F$ with local truncation error $\tau = A\hat{U} - F$, the true error is given as $U - \hat{U} = A^{-1}\tau$, thus $\|A^{-1}\|_\infty$ is needed if one needs the convergence in the maximum norm.

Of course, in general it is even harder to find what $\|A^{-1}\|_\infty$ is than finding eigenvalues of A. However, the matrices we have seen so far, e.g., K, T and T_2 are very special: they are **M-matrices**. The sharp characterization of an M-matrix is complicated and we only need a convenient sufficient condition here:

Theorem 2.1. *For a real square matrix A with positive diagonal entries and non-positive off-diagonal entries, A is a nonsingular M-matrix if all the row sums of A are non-negative and at least one row sum is positive.*

We need the following well-known but nontrival result to proceed:

Theorem 2.2. *Nonsingular M-matrices are monotone. Namely, the inverse matrix of a nonsingular M-matrix has non-negative entries.*

By this theorem, we have $K^{-1} \geq 0, T^{-1} \geq 0, T_2^{-1} \geq 0$, where the inequalities are entry-wise inequalities, i.e., these matrices have non-negative entries. If a matrix has an inverse with non-negative entries, then we call it monotone. So K, T and T_2 are monotone matrices.

Usually monotonicity of A can help to establish the estimate on ∞-norm of A^{-1}. If we can find a vector $\mathbf{v}$ such that $A\mathbf{v} = \mathbf{1}$ where $\mathbf{1}$ is a vector with each entry being 1, then $\|A^{-1}\|_\infty = \|A^{-1}\mathbf{1}\|_\infty = \|\mathbf{v}\|_\infty$.

2.5.1 Dirichlet boundary conditions

For the K matrix, in order to find $\mathbf{v}$ such that $K\mathbf{v} = \mathbf{1}$, first think about the exact solution to the problem $-u'' = 1, u(0) = u(1) = 0$, which is $v(x) = \frac{1}{2}x(1-x)$.

Let $\mathbf{v} = v(\mathbf{x})$ where $\mathbf{x}$ is the grid points for the corresponding scheme, i.e., $\mathbf{x} = \begin{bmatrix} h & 2h & \cdots & nh \end{bmatrix}^T$ with $h = \frac{1}{n+1}$. It is straightforward to verify that $\frac{1}{h^2}K\mathbf{v} = \mathbf{1}$. On the other hand, since $0 \leq v(x) \leq \frac{1}{8}$ for $x \in (0,1)$, we have $\|\mathbf{v}\|_\infty \leq \frac{1}{8}$, thus $\|(\frac{1}{h^2}K)^{-1}\|_\infty = \|\mathbf{v}\|_\infty \leq \frac{1}{8}$, with which we can easily establish the second order convergence in maximum norm.

2.5.2 Dirichlet and Neumann b.c.

The exact solution to the problem $-u'' = 1, u'(0) = u(1) = 0$, which is $v(x) = \frac{1}{2} - \frac{1}{2}x^2$. Let $\mathbf{v} = v(\mathbf{x})$ where $\mathbf{x}$ is the grid points for the corresponding scheme to T_2, i.e., $\mathbf{x} = \begin{bmatrix} 0 & h & 2h & \cdots & (n-1)h \end{bmatrix}^T$ with $h = \frac{1}{n}$. It is straightforward to verify that $\frac{1}{h^2}T_2\mathbf{v} = \mathbf{1}$. Thus similarly as in previous subsection, $\|(\frac{1}{h^2}T_2)^{-1}\|_\infty = \|\mathbf{v}\|_\infty \leq \frac{1}{2}$.

Let $D = diag\{\frac{1}{2}, 1, \cdots, 1\}$ be a diagonal matrix.

Recall that the local truncation error is only first order at the left boundary. In order to establish second order convergence, we need the estimate for the first column of T_2^{-1}. Let $A = \frac{1}{h^2}T_2$, if we can find a vector $\mathbf{w}$ such that $A\mathbf{w} \geq \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T$. Then the first column of A^{-1} is given by $A^{-1}\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \leq \mathbf{w}$ (inequality holds because A^{-1} has non-negative entries).

Let $w(x) = h(\frac{1}{2} - \frac{1}{2}x)$ and $\mathbf{w} = w(\mathbf{x})$, then it is straightforward to verify $\frac{1}{h^2}T_2\mathbf{w} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T$. (Obviously $w(x)$ is not the exact solution to $-u'' = \delta(0), u'(0) = u(1) = 0$. So how would you find $\mathbf{w}$?)

Let $A = [a_{ij}]$ and $A^{-1} = [a^{ij}]$. Then $\mathbf{w} = A^{-1}\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T$ implies that $\max_i |a_{i1}| = \max_i |w_i| \leq \frac{1}{2}h$.

So the true error is $(U - \hat{U})_i = (A^{-1}\tau)_i = a_{i1}\tau_1 + \sum_{j=2}^n a_{ij}\tau_j$ where $\tau_1 = \mathcal{O}(h)$ and $\tau_j = \mathcal{O}(h^2), j = 2, 3, \cdots, n$ (see previous section for the local truncation errors).

$$|(U - \hat{U})_i| \leq \frac{1}{2}h\tau_1 + \max_{j\geq 2}\tau_j \sum_{j=2}^n a_{ij}$$

$$\leq \frac{1}{2}h\tau_1 + \max_{j\geq 2}\tau_j \sum_{j=1}^n a_{ij} \leq \frac{1}{2}h\tau_1 + \max_{j\geq 2}\tau_j \|A^{-1}\|_\infty = \mathcal{O}(h^2).$$

2.6 1D BVP: Neumann b.c.

2.6.1 The matrix B on the one-half grid

Consider solving the 1D Poisson's equation with homogeneous Neumann boundary conditions:

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

Following Section 2.4.1, we use the grid $x_j = (j - \frac{1}{2})h$, $j = 1, \cdots, n$ with

$h = \frac{1}{n}$ and obtain a second order scheme $\frac{1}{h^2}BU_h = F$ with

$$B = \begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{pmatrix}.$$

The eigenvectors of B are $\cos(m\pi\mathbf{x})$ with eigenvalues $2 - 2\cos(m\pi h)$, $m = 0, \cdots, n-1$.

Problem 2.6. *Why do we choose eigenvectors as* $\cos(m\pi\mathbf{x})$, $m = 0, \cdots, n-1$ *rather than* $\cos(m\pi\mathbf{x})$, $m = 1, \cdots, n$*?*

2.6.2 An alternative point of view for one-half grid

For the boundary condition $u'(0) = 0$, instead of considering the ghost point, we can obtain the scheme in Section 2.6.1 by approximating the second order derivative $u''(\frac{1}{2}h)$ by the centered difference of the first order derivative:

$$u''(\frac{1}{2}h) \approx \frac{u'(h) - u'(0)}{h} = \frac{u'(h)}{h} \approx \frac{u(\frac{3}{2}h) - u(\frac{1}{2}h)}{h^2}.$$

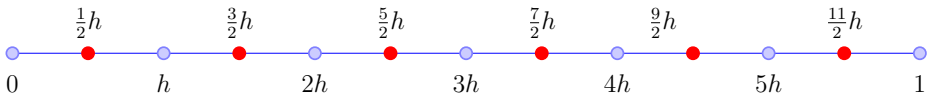


Figure 2.3: The integer grid and the one-half grid.

2.6.3 The matrix B on the integer grid

Following Section 2.4.2, on the grid $x_j = (j-1)h$, $j = 1, \cdots, n$ with $h = \frac{1}{n-1}$, we have a second order scheme $\frac{1}{h^2}BU_h = \begin{pmatrix} f_1/2 \\ f_2 \\ \vdots \\ f_n/2 \end{pmatrix}$ with

$$B = \begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{pmatrix}.$$

The eigenvectors and eigenvalues of B are the same as before. The advantage of B is its symmetry.

2.6.4 The matrix B_2 on the integer grid

On the grid $x_j = (j-1)h$, $j = 1, \cdots, n$ with $h = \frac{1}{n-1}$, the scheme in Section 2.6.3 is equivalent to $\frac{1}{h^2} B_2 U_h = F$ with

$$B_2 = \begin{pmatrix} 2 & -2 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -2 & 2 \end{pmatrix}.$$

The eigenvectors and eigenvalues of this matrix are: $\cos(\mathbf{x}m\pi)$, $2-2\cos(m\pi h), m = 0, 1, 2, ..., n-1$. Notice that the eigenvectors are not the same as those for the B matrix.

2.6.5 Compatibility Condition of Neumann b.c.

Consider a general purely Neumann boundary value problem

$$\begin{cases} -u''(x) = f(x), & x \in (0,1), \\ u'(0) = \sigma_0, \ u'(1) = \sigma_1. \end{cases}$$

Integrating the equation, we get

$$\int_0^1 f(x)dx = \int_0^1 -u''(x)dx = -u'(1) + u'(0) = \sigma_0 - \sigma_1. \qquad (2.4)$$

Thus the Poisson equation with a purely Neumann boundary condition has no solutions unless the compatbility condition (2.4) is satisfied. When (2.4) is satisfied, the solution is unique up to any constant.

Next we consider the scheme in Section 2.6.4:

$$\frac{1}{h^2} \begin{pmatrix} 2 & -2 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -2 & 2 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix} = \begin{pmatrix} f_1 - 2\sigma_0/h \\ f_2 \\ f_3 \\ \vdots \\ f_{n-1} \\ f_n + 2\sigma_1/h \end{pmatrix},$$

which can be denoted as

$$\frac{1}{h^2} B_2 U = F.$$

Obviously B_2 is not invertible since each row sums to zero. The linear system $\frac{1}{h^2}B_2U = F$ has a solution only if F is in the column space of B_2, which is not necessarily true even if (2.4) is satisfied. In other words, we still need a discrete compatbility condition so that the numerical scheme has a solution.

Notice that $F \in \mathrm{Col}(B_2)$ is equivalent to $F \perp \mathrm{Null}(B_2^T)$, and $\mathrm{Null}(B_2^T)$ is spanned by the vector $\begin{bmatrix}\frac{1}{2} & 1 & \cdots & 1 & \frac{1}{2}\end{bmatrix}^T$ (namely all columns of B_2 are orthogonal to this vector). So we obtain the following discrete compatbility condition

$$\frac{h}{2}f_1 + hf_2 + \cdots + hf_{n-1} + \frac{h}{2}f_n = \sigma_0 - \sigma_1. \tag{2.5}$$

Notice that $\frac{h}{2}f_1 + hf_2 + \cdots + hf_{n-1} + \frac{h}{2}f_n = \int_0^1 f(x)dx + \mathcal{O}(h^2)$ by the trapezoidal quadrature rule.

Thus given a problem satisfying (2.4), we would like to have a second order scheme $\frac{1}{h^2}B_2U = \bar{F}$ by generating a new right hand side vector $\bar{F}$ belonging to the column space of B_2, which can be obtained by projecting F to $\mathrm{Col}(B_2)$. In general, such a projection might be nontrivial to obtain. However, the rank of B_2 is $n-1$ and we know the orthogonal complement of $\mathrm{Col}(B_2)$ is spanned by $\begin{bmatrix}\frac{1}{2} & 1 & \cdots & 1 & \frac{1}{2}\end{bmatrix}^T$, thus it is straightforward to obtain this projection. First we have the compatbility error as

$$c = \frac{h}{2}f_1 + hf_2 + \cdots + hf_{n-1} + \frac{h}{2}f_n - \sigma_0 + \sigma_1 = \mathcal{O}(h^2).$$

Second, the projection $\bar{F}$ can be written as

$$\bar{F} = \begin{pmatrix} f_1 - 2a_0/h \\ f_2 \\ f_3 \\ \vdots \\ f_{n-1} \\ f_n + 2a_1/h \end{pmatrix} + a \begin{pmatrix} \frac{1}{2} \\ 1 \\ 1 \\ \vdots \\ 1 \\ \frac{1}{2} \end{pmatrix},$$

where a is a parameter determined by requiring $\bar{F}$ is orthogonal to $\begin{bmatrix}\frac{1}{2} & 1 & \cdots & 1 & \frac{1}{2}\end{bmatrix}^T$. So we obtain $a = \frac{-c}{h(n-\frac{3}{2})}$, thus

$$\bar{F} = \begin{pmatrix} f_1 - 2\sigma_0/h - c/h/(n-3/2)/2 \\ f_2 - c/h/(n-3/2) \\ f_3 - c/h/(n-3/2) \\ \vdots \\ f_{n-1} - c/h/(n-3/2) \\ f_n + 2\sigma_1/h - c/h/(n-3/2)/2 \end{pmatrix}, \tag{2.6}$$

Finally, if we prefer to solve a symmetric system, we can symmetrize the system by dividing the first and last row by two:

$$\frac{1}{h^2}\begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{pmatrix}\begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix} = \begin{pmatrix} \frac{1}{2}f_1 - \sigma_0/h - c/h/(n-3/2)/4 \\ f_2 - c/h/(n-3/2) \\ f_3 - c/h/(n-3/2) \\ \vdots \\ f_{n-1} - c/h/(n-3/2) \\ \frac{1}{2}f_n + \sigma_1/h - c/h/(n-3/2)/4 \end{pmatrix}.$$

Problem 2.7. *Suppose $\frac{1}{h^2}B_2U = F$ does not satisfy the discrete compatbility condition. Consider the least square solution $\hat{U}$ which is the minimizer for the function $\|\frac{1}{h^2}B_2U - F\|$ for the vector 2-norm. Show that $\hat{U}$ is a solution to $\frac{1}{h^2}B_2U = \bar{F}$.*

2.6.6 Inverting B and B_2

Notice that B is a singular matrix. In other words, there are infinitely many numerical solutions. The true solutions to the Neumann b.c. are not unique (any solution plus a constant is still a solution). To "invert" B, we can do the following by choosing a particular solution for $\frac{1}{h^2}BU = F$:

```
% eigenvectors and eigenvalues of B
h=1/n;
x=[h/2:h:1-h/2];lambda=2*ones(n,1)-2*cos([0:n-1]*pi*h);
S=cos(x*pi*[0:n-1]);
% Multiply the "inverse" of B/h^2 to a vector f
U=(inv(S)*F)./lambda;
U(1)=0;  % special treatment for the zero eigenvalue
U=S*U*h*h;
```

Let $\lambda_i = 2 - 2\cos((i-1)\pi h)$ then $\lambda_1 = 0$. Define Λ^{-1} as a diagonal matrix with diagonal entries $0, \frac{1}{\lambda_2}, \frac{1}{\lambda_3}, \cdots, \frac{1}{\lambda_n}$. Let S denote the eigenvector matrix then we set $U = S\Lambda^{-1}S^{-1}F$. Here we choose to set $\frac{1}{\lambda_1} = 0$, but of course you can choose it to be any other number. Since B is a positive definite matrix, thus if each column of S is normalized to be unit vector, its eigen-decomposition $B = S\Lambda S^{-1}$ is also its singular value decomposition (SVD). By setting $\frac{1}{\lambda_1} = 0$, the numerical solution has zero component along the first eigenvector which is $[1 \quad 1 \quad \cdots \quad 1]^T$. This means that the numerical solution U is perpendicular to $[1 \quad 1 \quad \cdots \quad 1]^T$ thus $\sum_j U_j = 0$.

Problem 2.8. *Use SVD to show that setting $\frac{1}{\lambda_1} = 0$ gives the least square solution for $\frac{1}{h^2}BU = F$.*

Problem 2.9. *For the B_2 matrix, we can use the same procedure as above by setting $\frac{1}{\lambda_1} = 0$. However, it no longer gives a solution satisfying $\sum_j U_j = 0$. Why is this?*

Remark 2.3. *Since the exact solution is not unique, we should compare our numerical solution satisfying $\sum_j U_j = 0$ with a shifted exact solution: suppose $\widehat{U} = [u(x_1), \cdots, u(x_n)]^T$ where $u(x)$ is any exact solution, then we should compute the error as $E_h = U - \widetilde{U}$ with $\widetilde{U} = [u(x_1) - \frac{1}{n}\bar{U}, \cdots, u(x_n) - \frac{1}{n}\bar{U}]^T$ and $\bar{U} = \sum_j u(x_j)$.*

For the B_2 matrix, let $B_2 = S\Lambda S^{-1}$ be its eigendecomposition with $\lambda_1 = 0$. Recall that the eigenvectors and eigenvalues of B_2 are: $\cos(\mathbf{x}m\pi)$, $2 - 2\cos(m\pi\frac{1}{n-1}), m = 0, 1, 2, ..., n-1$, where $\mathbf{x} = \begin{bmatrix} 0 & \frac{1}{n-1} & \frac{2}{n-1} & \cdots & 1 \end{bmatrix}$. Let $\mathbf{w} = \begin{bmatrix} \frac{1}{2} & 1 & \cdots & 1 & \frac{1}{2} \end{bmatrix}^T$. Let $\mathbf{v}_i$ be the eigenvector for λ_i. Then $0 = \mathbf{w}^T B_2 \mathbf{v}_i = \lambda_i \mathbf{w}^T \mathbf{v}_i$ implies $\mathbf{w}^T \mathbf{v}_i = 0$ if $\lambda_i \neq 0$. Thus the eigenvectors associated with nonzero eigenvalues are orthogonal to $\mathbf{w}$, i.e., the orthogonal complement of the column space of B_2.

Problem 2.10. *For any vector $\bar{F}$ satisfying the discrete compatbility condition (2.5), $\bar{F}$ is orthogonal to $\mathbf{w}$. Show that $U = S\Lambda^{-1}S^{-1}\bar{F}$ (with $\frac{1}{\lambda_1}$ being defined as zero) is a solution to $B_2U = \bar{F}$. In other words, show that*

$$S \begin{pmatrix} 0 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & 1 \end{pmatrix} S^{-1}\bar{F} = \bar{F}.$$

Hint: show that $\bar{F}$ lives in the eigenspace for eigenvalues $\lambda_2, \cdots, \lambda_n$ thus

$$\bar{F} = S \begin{pmatrix} 0 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{pmatrix}.$$

For the B_2 matrix, and a vector F which does not satisfy the discrete compatbility condition (2.5), i.e., F is not in the column space of B_2, the vector $U = S\Lambda^{-1}S^{-1}F$ (with $\frac{1}{\lambda_1}$ being defined as zero) is the least square solution to $B_2U = F$. In other words, $B_2U = B_2S\Lambda^{-1}S^{-1}F$ is the projection of F onto the column space of B_2. To see why it is true, assume $\tilde{F}$ is the projection of F onto the column space of B_2, then we know $B_2S\Lambda^{-1}S^{-1}\tilde{F} =$

$\tilde{F}$, therefore we get

$$B_2U = B_2S\Lambda^{-1}S^{-1}(F-\tilde{F})+S\begin{pmatrix}0 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & 1\end{pmatrix}S^{-1}\tilde{F} = B_2S\Lambda^{-1}S^{-1}(F-\tilde{F})+\tilde{F}.$$

So we only need to show $B_2S\Lambda^{-1}S^{-1}(F-\tilde{F}) = 0$. And we know $F-\tilde{F}$ is orthogonal to the column space of B_2. Notice that $F-\tilde{F}$ should be exactly the shift we added to generate $\bar{F}$ in (2.6).

Problem 2.11. *Show that* $B_2S\Lambda^{-1}S^{-1}(F-\tilde{F}) = 0$.

2.7 1D BVP: periodic b.c.

Consider solving the 1D Poisson's equation with periodic boundary conditions:

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

We use the grid $x_j = (j-1)h$, $j = 1, \cdots, n$ with $h = \frac{1}{n}$ and obtain a second order scheme $\frac{1}{h^2}CU_h = F$ with

$$C = \begin{pmatrix} 2 & -1 & & & & -1 \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ -1 & & & & -1 & 2 \end{pmatrix}.$$

The eigenvectors of C are $\exp(-\mathbf{i}m2\pi\mathbf{x})$ with eigenvalues $2-2\cos(m2\pi h)$, $m = 0, \cdots, n-1$. By choosing these eigenvectors, the eigenvector matrix is precisely the discrete Fourier transform matrix, i.e., dftmtx(n) in MATLAB. Multiplying the DFT matrix is equivalent to FFT.

Remark 2.4. *The matrix C is circulant. The columns of DFT matrix are eigenvectors to any circulant matrix thus the DFT matrix can diagonalize any circulant matrix.*

The matrix C is singular. To "invert" the matrix C, we can do the same thing as previously for the matrix B: set the component along the zero eigenvector to be zero, which gives a numerical solution summing to zero.

2.8 2D BVP: Dirichlet b.c.

Consider solving the 2D Poisson's equation with homogeneous Dirichlet boundary conditions:

$$\begin{cases} -u_{xx}(x,y) - u_{yy}(x,y) = f(x,y), & (x,y) \in (0,1) \times (0,2), \\ u(x,y)|_\Gamma = 0, \end{cases}$$

where Γ denotes the boundary of the rectangular domain.

We use the grid $x_i = i\Delta x$, $i = 1, \cdots, Nx$ with $\Delta x = \frac{1}{Nx+1}$ and $y_j = j\Delta y$, $j = 1, \cdots, Ny$ with $\Delta y = \frac{2}{Ny+1}$. Let U_{ij} be the numerical solution at (x_i, y_j). Let U be a $Ny \times Nx$ matrix such that $U(j,i) = U_{ij}$.

We will use two operators:

- Kronecker product of two matrices: if A is $m \times n$ and B is $p \times q$, then $A \otimes B$ is $mp \times nq$ give by

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \vdots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}.$$

- For a $m \times n$ matrix X, $vec(X)$ denotes the vectorization of the matrix X, i.e., rearranging X into a vector column by column. In MATLAB the *reshape* function can act as the inverse of the *vec* operator: if $v = vec(X)$, then $reshape(v, m, n)$ recovers X.

The following properties will be used:

1. $(A \otimes B)(C \otimes D) = AC \otimes BD$.
2. $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$.
3. $(B^T \otimes A)vec(X) = vec(AXB)$.
4. $(A \otimes B)^T = A^T \otimes B^T$.

The second order finite difference scheme for the interior nodes is given by

$$\frac{-U_{i-1,j} + 2U_{ij} - U_{i+1,j}}{\Delta x^2} + \frac{-U_{i,j-1} + 2U_{ij} - U_{i,j+1}}{\Delta y^2} = f_{ij}. \tag{2.7}$$

Let F be a $Ny \times Nx$ matrix with entries $F(j,i) = f(x_i, y_j)$. The matrix vector form of this scheme can be written as

$$\left(\frac{1}{\Delta x^2} K_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2} K_y\right) vec(U) = vec(F),$$

where K_x is the K matrix of size $Nx \times Nx$, I_x is the identity matrix of size $Nx \times Nx$, K_y is the K matrix of size $Ny \times Ny$, and I_y is the identity matrix of size $Ny \times Ny$.

Problem 2.12. *Find S_y and Λ_y.*

Let $K2D$ denote the matrix $\frac{1}{\Delta x^2}K_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}K_y$. Since we know the eigenvectors of K_x and K_y, we can find the inverse of the matrix $K2D$ by the following *eigenvector method*. Suppose the eigendecompositions of K_x and K_y are $K_x = S_x\Lambda_x S_x^{-1}$ and $K_y = S_y\Lambda_y S_y^{-1}$, then

$$\begin{aligned}\frac{1}{\Delta x^2}K_x \otimes I_y &= S_x\frac{1}{\Delta x^2}\Lambda_x S_x^{-1} \otimes I_y = S_x\frac{1}{\Delta x^2}\Lambda_x S_x^{-1} \otimes S_y I_y S_y^{-1} \\ &= (S_x \otimes S_y)(\frac{1}{\Delta x^2}\Lambda_x S_x^{-1} \otimes I_y S_y^{-1}) = (S_x \otimes S_y)(\frac{1}{\Delta x^2}\Lambda_x \otimes I_y)(S_x^{-1} \otimes S_y^{-1}),\end{aligned}$$

where the first property of the kronecker product is used twice. Similarly we get

$$I_x \otimes \frac{1}{\Delta y^2}K_y = (S_x \otimes S_y)(I_x \otimes \frac{1}{\Delta y^2}\Lambda_y)(S_x^{-1} \otimes S_y^{-1}).$$

Thus we get the eigenvectors and eigenvalues of $K2D$:

$$K2D = (S_x \otimes S_y)(\frac{1}{\Delta x^2}\Lambda_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}\Lambda_y)(S_x^{-1} \otimes S_y^{-1}).$$

Implementation:

1. $(S_x \otimes S_y)^{-1}vec(F) = (S_x^{-1} \otimes S_y^{-1})vec(F) = vec(S_y^{-1}FS_x^{-1})$ where $S_x^T = S_x$ is used.
2. Let $\Lambda = reshape(diag(\frac{1}{\Delta x^2}\Lambda_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}\Lambda_y), Ny, Nx)$. Then $\Lambda(j,i) = \frac{2-2\cos(j\frac{\pi}{2}\Delta y)}{\Delta y^2} + \frac{2-2\cos(i\pi\Delta x)}{\Delta x^2}$. Because $\frac{1}{\Delta x^2}\Lambda_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}\Lambda_y$ is a diagonal matrix, we have

   $$(\frac{1}{\Delta x^2}\Lambda_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}\Lambda_y)^{-1}vec(S_y^{-1}FS_x^{-1}) = vec(S_y^{-1}FS_x^{-1}./\Lambda),$$

   where ./ denotes the component-wise division.
3. $(S_x^{-1} \otimes S_y^{-1})^{-1}vec(S_y^{-1}FS_x^{-1}./\Lambda) = (S_x \otimes S_y)vec(S_y^{-1}FS_x^{-1}./\Lambda) = vec(S_y(S_y^{-1}FS_x^{-1}./\Lambda)S_x)$ where $S_x^T = S_x$ is again used.

To summarize, we simply have $U = S_y(S_y^{-1}FS_x^{-1}./\Lambda)S_x$, which has $\mathcal{O}(N^3)$ complexity if $N = Nx = Ny$. If DST is used for multiplying S_x and S_x^{-1}, it reduces to $\mathcal{O}(N^2 \log_2 N)$. The Gaussian elimination of $K2D$ costs $\mathcal{O}(N^4)$ ($\mathcal{O}(N^7)$ in 3D) because the bandwidth is N (N^2 in 3D).

Problem 2.13. *Show that $K2D$ is symmetric.*

Problem 2.14. *Consider a matrix given in the form of $A \otimes B + B \otimes A$ where A and B are matrices of size $n \times n$. Can it be inverted using a similar procedure as described in this subsection through eigen-decomposition of small matrices of size $n \times n$? Find reasonable assumptions to make the answer to be yes.*

2.9 2D BVP: Neumann b.c.

Consider solving the 2D Poisson's equation with homogeneous Neumann boundary conditions:

$$\begin{cases} -u_{xx}(x,y) - u_{yy}(x,y) = f(x,y), & (x,y) \in (0,1) \times (0,2), \\ \frac{\partial}{\partial \mathbf{n}} u(x,y)|_\Gamma = 0, \end{cases}$$

where Γ denotes the boundary of the rectangular domain and $\frac{\partial}{\partial \mathbf{n}} u(x,y)|_\Gamma$ denotes the directional derivative of u along the direction normal to Γ.

2.9.1 The one-half grid

We first use the grid in Section 2.6.1. Let $x_i = (i - \frac{1}{2})\Delta x$, $i = 1, \cdots, Nx$ with $\Delta x = \frac{1}{Nx}$ and $y_j = (j - \frac{1}{2})\Delta y$, $j = 1, \cdots, Ny$ with $\Delta y = \frac{2}{Ny}$. Then we get the scheme

$$\left(\frac{1}{\Delta x^2} B_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2} B_y \right) vec(U) = vec(F),$$

where the matrix B in Section 2.6.1 is used for B_x and B_y.

The solution can be found by $U = S_y(S_y^{-1} F S_x^{-1}./\Lambda)S_x$ where S_x and S_y are the eigenvector matrices to B_x and B_y. Here $\Lambda(1,1) = 0$ so we set the $(1,1)$ entry in $S_y^{-1} F S_x^{-1}./\Lambda$ to be zero, which returns a soluition with zero sum.

2.9.2 The integer grid: matrix B

If we choose to use the method in Section 2.6.3 to contruct the scheme in 2D on an integer grid $x_i = (i-1)\Delta x$, $i = 1, \cdots, Nx$ with $\Delta x = \frac{1}{Nx-1}$ and $y_j = (j-1)\Delta y$, $j = 1, \cdots, Ny$ with $\Delta y = \frac{2}{Ny-1}$, then the scheme becomes

$$\left(\frac{1}{\Delta x^2} B_x \otimes E_y + E_x \otimes \frac{1}{\Delta y^2} B_y \right) vec(U) = vec(\widehat{F}),$$

where $\widehat{F}$ is the modified right hand side data, and the diagonal matrix E replaces the identity matrix

$$E = \begin{pmatrix} \frac{1}{2} & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & \frac{1}{2} \end{pmatrix}.$$

To see why we should have the extra $\frac{1}{2}$ factor in the $2D$ scheme, consider the scheme at the point (x_m, y_1) for some $2 < m < Nx$ on the left boundary of the domain. Then the second FD approximation at (x_m, y_1) is

$$\frac{2U_{m,1} - 2U_{m,2}}{\Delta x^2} + \frac{-U_{m-1,1} + 2U_{m,1} - U_{m+1,2}}{\Delta y^2} = f_{ij}.$$

But in order to have the matrix B as in Section 2.6.3, we need to convert it to

$$\frac{U_{m,1} - U_{m,2}}{\Delta x^2} + \frac{1}{2}\frac{-U_{m-1,1} + 2U_{m,1} - U_{m+1,2}}{\Delta y^2} = \frac{1}{2}f_{ij}.$$

Due to the extra $\frac{1}{2}$ in the D matrix, the eigenvectors of $\frac{1}{\Delta x^2}B_x \otimes E_y + E_x \otimes \frac{1}{\Delta y^2}B_y$ is no longer $S_x \otimes S_y$.

2.9.3 The integer grid: matrix B_2

The scheme in the previous subsection is equivalent to

$$\left(\frac{1}{\Delta x^2}B_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}B_y\right) vec(U) = vec(F),$$

where the matrix B_x and B_y are the matrix B_2 in Section 2.6.4.

The solution can be found by $U = S_y(S_y^{-1}FS_x^{-1}./\Lambda)S_x$ where S_x and S_y are the eigenvector matrices to B_x and B_y. Here $\Lambda(1,1) = 0$ so we set the $(1,1)$ entry in $S_y^{-1}FS_x^{-1}./\Lambda$ to be zero, which returns a soluition with zero sum.

The normal derivatives at four corner points are not really well defined though.

2.10 The 9-point Laplacian

The 5-point stencil scheme (2.7) for the solving the 2D Poisson equation is second order accurate. If we use Δ_5 to denote the 5-point discrete Laplacian, then

$$-\Delta_5 U_{i,j} = \frac{-U_{i-1,j} + 2U_{ij} - U_{i+1,j}}{\Delta x^2} + \frac{-U_{i,j-1} + 2U_{ij} - U_{i,j+1}}{\Delta y^2},$$

and the matrix representation of the operator $-\Delta_5$ for the homogeneous Dirichlet boundary condition is

$$K2D = \frac{1}{\Delta x^2}K_x \otimes I_y + I_x \otimes \frac{1}{\Delta y^2}K_y.$$

Now consider the following 9-point Laplacian for $h = \Delta x = \Delta y$:

$$\begin{aligned}\Delta_9 U_{i,j} =& \frac{1}{6h^2}(4U_{i-1,j} + 4U_{i+1,j} + 4U_{i,j-1} + 4U_{i,j+1} \\ &+ U_{i-1,j-1} + U_{i+1,j-1} + U_{i+1,j+1} + U_{i+1,j+1} - 20U_{i,j}).\end{aligned}$$

For convenience, let $u_{i,j}$ denote the value of a smooth function $u(x,y)$ at the point (x_i, y_j), then we perform the Taylor expansion for the smooth function around the point (x_i, y_j),

$$\begin{aligned} u_{i,j\pm1} = u_{i,j} &\pm \Delta y(u_y)_{i,j} + \frac{1}{2}\Delta y^2(u_{yy})_{i,j} \pm \frac{1}{6}\Delta y^3(u_{yyy})_{i,j} + \frac{1}{24}\Delta y^4(u_{yyyy})_{i,j} \\ &\pm \frac{1}{120}\Delta y^5(u_{yyyyy})_{i,j} + \frac{1}{6!}\Delta y^6(\partial_y^6 u)_{i,j} \pm \frac{1}{7!}\Delta y^7(\partial_y^7 u)_{i,j} + \mathcal{O}(\Delta y^8), \end{aligned}$$

thus

$$u_{i,j+1}+u_{i,j-1} = 2u_{i,j}+\Delta y^2(u_{yy})_{i,j}+\frac{1}{12}\Delta y^4(u_{yyyy})_{i,j}+\frac{2}{6!}\Delta y^6(\partial_y^6 u)_{i,j}+\mathcal{O}(\Delta y^8).$$

Similarly,

$$u_{i+1,j}+u_{i-1,j} = 2u_{i,j}+\Delta x^2(u_{xx})_{i,j}+\frac{1}{12}\Delta x^4(u_{xxxx})_{i,j}+\frac{2}{6!}\Delta x^6(\partial_x^6 u)_{i,j}+\mathcal{O}(\Delta x^8).$$

Next we first perform the Taylor expansion around the point (x_{i+1}, y_j):

$$u_{i+1,j+1}+u_{i+1,j-1} = 2u_{i+1,j}+\Delta y^2(u_{yy})_{i+1,j}+\frac{1}{12}\Delta y^4(u_{yyyy})_{i+1,j}+\frac{2}{6!}\Delta y^6(\partial_y^6 u)_{i+1,j}+\mathcal{O}(\Delta y^8),$$

$$u_{i-1,j+1}+u_{i-1,j-1} = 2u_{i-1,j}+\Delta y^2(u_{yy})_{i-1,j}+\frac{1}{12}\Delta y^4(u_{yyyy})_{i-1,j}+\frac{2}{6!}\Delta y^6(\partial_y^6 u)_{i-1,j}+\mathcal{O}(\Delta y^8).$$

Then we perform the Taylor expansion around the point (x_i, y_j),

$$\begin{aligned} &u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j+1} + u_{i-1,j-1} \\ =&2[2u_{i,j} + \Delta x^2(u_{xx})_{i,j} + \frac{1}{12}\Delta x^4(u_{xxxx})_{i,j} + \frac{2}{6!}\Delta x^6(\partial_x^6 u)_{i,j} + \mathcal{O}(\Delta x^8)] \\ &+ \Delta y^2[(u_{yy})_{i-1,j} + (u_{yy})_{i+1,j}] + \frac{1}{12}\Delta y^4[(u_{yyyy})_{i-1,j} + (u_{yyyy})_{i+1,j}] \\ &+ \frac{2}{6!}\Delta y^6[(\partial_y^6 u)_{i-1,j} + (\partial_y^6 u)_{i+1,j}] + \mathcal{O}(\Delta y^8) \\ =&4u_{i,j} + 2\Delta x^2(u_{xx})_{i,j} + \frac{1}{6}\Delta x^4(u_{xxxx})_{i,j} + \frac{4}{6!}\Delta x^6(\partial_x^6 u)_{i,j} + \mathcal{O}(\Delta x^8) \\ &+ \Delta y^2[2(u_{yy})_{i,j} + \Delta x^2(u_{yyxx})_{i,j} + \frac{1}{12}\Delta x^4(u_{yyxxxx})_{i,j} + \mathcal{O}(\Delta x^6)] \\ &+ \frac{1}{12}\Delta y^4[2(u_{yyyy})_{i,j} + \Delta x^2(u_{yyyyxx})_{i,j} + \mathcal{O}(\Delta x^4)] + \frac{2}{6!}\Delta y^6[2(\partial_y^6 u)_{i,j} + \mathcal{O}(\Delta x^2)] + \mathcal{O}(\Delta y^8) \\ =&4u_{i,j} + 2\Delta x^2(u_{xx})_{i,j} + 2\Delta y^2(u_{yy})_{i,j} + \frac{1}{6}\Delta x^4(u_{xxxx})_{i,j} + \frac{1}{6}\Delta x^4(u_{yyyy})_{i,j} + \Delta x^2\Delta y^2(u_{yyxx})_{i,j} \\ &+ \frac{4}{6!}\Delta x^6(\partial_x^6 u)_{i,j} + \frac{4}{6!}\Delta y^6(\partial_y^6 u)_{i,j} + \frac{1}{12}\Delta y^4\Delta x^2(\partial_y^4\partial_x^2 u)_{i,j} + \frac{1}{12}\Delta x^4(\Delta y^2\partial_x^4\partial_y^2 u)_{i,j} \\ &+ \mathcal{O}(\Delta x^8) + \mathcal{O}(\Delta x^6\Delta y^2) + \mathcal{O}(\Delta x^2\Delta y^6) + \mathcal{O}(\Delta y^8) \end{aligned}$$

Since we have assumed $h = \Delta x = \Delta y$, we get

$$\Delta_9 u(x_i, y_j) = u_{xx}+u_{yy}+\frac{1}{12}h^2(u_{xxxx}+2u_{xxyy}+u_{yyyy})+\frac{1}{360}h^4(\partial_x^2+\partial_y^2)(u_{xxxx}+4u_{xxyy}+u_{yyyy})+\mathcal{O}($$

which implies the 9-point discrete Laplacian is only second order accurate. However the second order error term is precisely $\Delta^2 u = \Delta(\Delta u) = u_{xxxx} + 2u_{xxyyy} + u_{yyyy}$, where Δ^2 is the biharmonic operator. If u is the solution to $-\Delta u = -u_{xx} - u_{yy} = f$, then $\Delta^2 u = -\Delta f$. Therefore, the following scheme is fourth order accurate for solving $-u_{xx} - u_{yy} = f$:

$$-\Delta_9 U_{i,j} = F_{i,j},$$

where

$$F_{i,j} = f(x_i, y_j) + \frac{h^2}{12}\Delta_5 f(x_i, y_j).$$

Let $K2D_9$ be the matrix representing the 9-point Laplacian operator $-\Delta_9$ for the homogeneous Dirichlet boundary condition, then it can be written as

$$K2D_9 = -\frac{1}{6h^2}[H_x \otimes H_y - 36 I_x \otimes I_y],$$

where H is a symmetric tridiagonal matrix

$$H = \begin{pmatrix} 4 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & & 1 & 4 \end{pmatrix}.$$

Problem 2.15. *Explain why the matrix representation of the 9-point Laplacian operator is given as the matrix $K2D_9$.*

Problem 2.16. *Find the eigenvectors and eigenvalues of $K2D_9$ and use them to establish the stability thus convergence of the scheme.*

For the 9-point Laplacian, the size of the matrices H_x and H_y do not need to be the same, and we do not need $\Delta x = \Delta y$. As a matter of fact the 9-point discrete Laplacian for the Poisson equation can be also written as

$$\frac{1}{12}\left[\frac{1}{\Delta x^2}\begin{pmatrix} 1 & -2 & 1 \\ 10 & -20 & 10 \\ 1 & -2 & 1 \end{pmatrix} + \frac{1}{\Delta y^2}\begin{pmatrix} 1 & 10 & 1 \\ -2 & -20 & -2 \\ 1 & 10 & 1 \end{pmatrix}\right] \circ \bar{U} = \frac{1}{12}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 8 & 1 \\ 0 & 1 & 0 \end{pmatrix} \circ \bar{F},$$

which reduces to the following under the assumption $\Delta x = \Delta y = h$,

$$\frac{1}{6h^2}\begin{pmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{pmatrix} \circ \bar{U} = \frac{1}{12}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 8 & 1 \\ 0 & 1 & 0 \end{pmatrix} \circ \bar{F}.$$

Here $\circ$ denotes the Hadamard product of matrices, and $\bar{U}$ and $\bar{F}$ are matrices:

$$\bar{U} = \begin{pmatrix} U_{i-1,j+1} & U_{i,j+1} & U_{i+1,j+1} \\ U_{i-1,j} & U_{i,j} & U_{i+1,j} \\ U_{i-1,j-1} & U_{i,j-1} & U_{i+1,j-1} \end{pmatrix}, \quad \bar{F} = \begin{pmatrix} F_{i-1,j+1} & F_{i,j+1} & F_{i+1,j+1} \\ F_{i-1,j} & F_{i,j} & F_{i+1,j} \\ F_{i-1,j-1} & F_{i,j-1} & F_{i+1,j-1} \end{pmatrix}.$$

Problem 2.17. *Verify the fourth order local truncation error in the 9-point scheme for $\Delta x \neq \Delta y$.*

Problem 2.18. *Prove that the accuracy of the 9-point scheme becomes sixth order for solving the Laplace equation $\Delta u = 0$ with Dirichlet boundary conditions.*

2.11 Variable coefficient problems

2.11.1 1D Dirichlet b.c.

We first consider a 1D variable coefficient problem:

$$-(a(x)u'(x))' = f(x), \quad x \in [0,1],$$

with homogeneous Dirichlet boundary conditions. A conservative discretization should be used:

$$\frac{1}{\Delta x^2}[-a_{j-\frac{1}{2}}u_{j-1} + (a_{j-\frac{1}{2}} + a_{j+\frac{1}{2}})u_j - a_{j+\frac{1}{2}}u_{j+1}] = f_j,$$

where $a_{j-\frac{1}{2}} = a(x_j - \frac{1}{2}\Delta x)$. The matrix vector form of this scheme is $Bu = f$ where B is a real symmetric tridiagonal matrix:

$$B = \frac{1}{\Delta x^2}\begin{pmatrix} a_{\frac{1}{2}} + a_{\frac{3}{2}} & -a_{\frac{3}{2}} & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}} + a_{\frac{5}{2}} & -a_{\frac{5}{2}} & \\ & \ddots & \ddots & \ddots \end{pmatrix}. \tag{2.8}$$

Notice that $x_j = j\frac{1}{n+1}$ for $j = 1, \cdots, n$ are the n grid points and the coefficient function $a(x)$ is sampled at $n+1$ points $x_{j-\frac{1}{2}} = (j - \frac{1}{2})\frac{1}{n+1}$ for $j = 1, \cdots, n+1$. Let D denote the $(n+1) \times n$ matrix:

$$D = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \\ & & & -1 \end{bmatrix},$$

and A be a diagonal matrix with diagonal entries $a_{\frac{1}{2}}, \cdots, a_{n+\frac{1}{2}}$. Then we have $D^T D = K$ and $B = \frac{1}{\Delta x^2} D^T A D$.

Obviously it is impossible to compute eigenvalues of the matrix B anymore. However, it is still possible to have a useful estimate of the smallest eigenvalues of the matrix B as will be shown in Chapter 3 (Section 3.6.4), so that we can establish the stability thus the convergence of the scheme.

2.11.2 2D Dirichlet b.c.

Now consider the following 2D problem

$$\nabla(a(x,y)\nabla u) = f(x,y),$$

with homogeneous Dirichlet b.c. on a rectangular domain, where $a(x,y) > 0$ is some known coefficient function. If we use the same conservative centered difference discretization as above, then we obtain

$$\left[\frac{1}{\Delta x^2}(D_x^T \otimes I_y)A_1(D_x \otimes I_y) + \frac{1}{\Delta y^2}(I_x \otimes D_y^T)A_2(I_x \otimes D_y)\right] vec(U) = vec(F),$$

where A_1 and A_2 are two diagonal matrices defined as follows.

Let a_1 be a 2D array of size $Ny \times (Nx+1)$ satisfying $a_2(j,i) = a(x_{i-\frac{1}{2}}, y_j)$ and a_2 be a 2D array of size $(Ny+1) \times Nx$ satisfying $a_1(j,i) = a(x_i, y_{j-\frac{1}{2}})$. The diagonal entries of A_1 are

$$reshape(a_1, (Nx+1)Ny, 1),$$

and the diagonal entries of A_2 are

$$reshape(a_2, (Ny+1)Nx, 1).$$

Problem 2.19. *Show that the matrix* $\frac{1}{\Delta x^2}(D_x^T \otimes I_y)A_1(D_x \otimes I_y) + \frac{1}{\Delta y^2}(I_x \otimes D_y^T)A_2(I_x \otimes D_y)$ *is symmetric.*

2.11.3 1D Neumann b.c.

Next we consider the Neumann boundary conditions:

$$-(a(x)u'(x))' = f(x), \quad x \in [0,1], \quad u'(0) = \sigma_0, u'(1) = \sigma_1.$$

The compatbility condition for this problem is

$$\int_0^1 f(x)dx = -a(1)\sigma_1 + a(0)\sigma_0.$$

In this section we use the grid points $x_j = j\frac{1}{n+1}$ for $j = 0, \cdots, n+1$. The half grid points are $x_{j+\frac{1}{2}} = (j+\frac{1}{2})\frac{1}{n+1}$. For approximating the boundary conditions, we can use a second order one-sided difference:

$$\frac{1}{\Delta x}\left(-\frac{3}{2}u(0) + 2u(\Delta x) - \frac{1}{2}u(2\Delta x)\right) = u'(0) + \mathcal{O}(\Delta x^2),$$

$$\frac{1}{\Delta x}\left(\frac{1}{2}u(1-2\Delta x) - 2u(1-\Delta x) + \frac{3}{2}u(1)\right) = u'(1) + \mathcal{O}(\Delta x^2),$$

thus

$$-\frac{3}{2}U_0 + 2U_1 - \frac{1}{2}U_2 = h\sigma_0,$$

$$\frac{1}{2}U_{n-1} - 2U_n + \frac{3}{2}U_{n+1} = h\sigma_1.$$

Therefore, we get

$$U_0 = \frac{4}{3}U_1 - \frac{1}{3}U_2 - \frac{2}{3}h\sigma_0,$$

$$U_{n+1} = -\frac{1}{3}U_{n-1} + \frac{4}{3}U_n + \frac{2}{3}h\sigma_1.$$

So we get a second order approximation to the first order derivative:

$$\begin{pmatrix} u'(x_{\frac{1}{2}}) \\ u'(x_{\frac{3}{2}}) \\ \vdots \\ u'(x_{\frac{n+1}{2}}) \end{pmatrix} \approx \frac{1}{h}\begin{bmatrix} -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}\begin{pmatrix} U_0 \\ U_1 \\ \vdots \\ U_{n-1} \\ U_{n+1} \end{pmatrix} = \frac{1}{h}\begin{bmatrix} -\frac{1}{3} & \frac{1}{3} & & \\ -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \\ & & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}\begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix} + \begin{pmatrix} \frac{2}{3}\sigma_0 \\ 0 \\ \vdots \\ 0 \\ \frac{2}{3}\sigma_1 \end{pmatrix}.$$

Thus we get a second order approximation for $(a(x)u')'$:

$$\frac{1}{h}\begin{bmatrix} -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}\begin{bmatrix} a_{\frac{1}{2}} & & & \\ & a_{\frac{3}{2}} & & \\ & & \ddots & \\ & & & a_{n+\frac{1}{2}} \end{bmatrix}\left(\frac{1}{h}\begin{bmatrix} -\frac{1}{3} & \frac{1}{3} & & \\ -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \\ & & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}\begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix} + \begin{pmatrix} \frac{2}{3}\sigma_0 \\ 0 \\ \vdots \\ 0 \\ \frac{2}{3}\sigma_1 \end{pmatrix}\right).$$

A second order scheme for $-(a(x)u')' = f$ can be written as

$$\frac{1}{h^2}\begin{pmatrix} a_{\frac{3}{2}} - \frac{1}{3}a_{\frac{1}{2}} & -a_{\frac{3}{2}} + \frac{1}{3}a_{\frac{1}{2}} & & & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}} + a_{\frac{5}{2}} & -a_{\frac{5}{2}} & & & \\ & \ddots & \ddots & \ddots & & \\ & & -a_{n-\frac{3}{2}} & a_{n-\frac{3}{2}} + a_{n-\frac{1}{2}} & -a_{n-\frac{1}{2}} & \\ & & & \frac{1}{3}a_{n+\frac{1}{2}} - a_{n-\frac{1}{2}} & a_{n-\frac{1}{2}} - \frac{1}{3}a_{n+\frac{1}{2}} \end{pmatrix}\begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix}$$

$$= \begin{pmatrix} f_1 - \frac{1}{h}a_{\frac{1}{2}}\frac{2}{3}\sigma_0 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n + \frac{1}{h}a_{n+\frac{1}{2}}\frac{2}{3}\sigma_1 \end{pmatrix},$$

which can be denoted as

$$\frac{1}{h^2}AU = F.$$

The discrete compatbility condition is

$$\frac{a_{\frac{3}{2}}}{a_{\frac{3}{2}}-\frac{1}{3}a_{\frac{1}{2}}}(hf_1-a_{\frac{1}{2}}\frac{2}{3}\sigma_0)+hf_2+\cdots+hf_{n-1}+\frac{a_{n-\frac{1}{2}}}{a_{n-\frac{1}{2}}-\frac{1}{3}a_{n+\frac{1}{2}}}(hf_n+a_{n+\frac{1}{2}}\frac{2}{3}\sigma_1)=0. \tag{2.9}$$

Problem 2.20. *Show that the discrete compatbility condition is indeed* (2.9) *and show it is a second order accurate approximation to the compatbility condition for* $a(x)$ *being a constant.*

To obtain a modified right hand side $\bar{F}$ so that $\frac{1}{h^2}AU=F$ has a solution if F does not satisfy (2.9), we can project F to the column space of A. Similar to the discussions in Section 2.6.5, we have

$$\bar{F}=\begin{pmatrix} f_1-2a_0/h \\ f_2 \\ f_3 \\ \vdots \\ f_{n-1} \\ f_n+2a_1/h \end{pmatrix}+b\begin{pmatrix} a_{\frac{3}{2}}/(a_{\frac{3}{2}}-\frac{1}{3}a_{\frac{1}{2}}) \\ 1 \\ 1 \\ \vdots \\ 1 \\ a_{n-\frac{1}{2}}/(a_{n-\frac{1}{2}}-\frac{1}{3}a_{n+\frac{1}{2}}) \end{pmatrix},$$

where b is obtained by requiring $\bar{F}$ is orthogonal to the vector $\begin{pmatrix} a_{\frac{3}{2}}/(a_{\frac{3}{2}}-\frac{1}{3}a_{\frac{1}{2}}) \\ 1 \\ 1 \\ \vdots \\ 1 \\ a_{n-\frac{1}{2}}/(a_{n-\frac{1}{2}}-\frac{1}{3}a_{n+\frac{1}{2}}) \end{pmatrix}$.

Sometimes we may prefer to solve a symmetric system. For this matter, we can symmetrize the system by dividing proper numbers in the first and the last row of the system to obtain an equivalent symmetric one:

$$\frac{1}{h^2}\begin{pmatrix} a_{\frac{3}{2}} & -a_{\frac{3}{2}} & & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}}+a_{\frac{5}{2}} & -a_{\frac{5}{2}} & & \\ & \ddots & \ddots & \ddots & \\ & & -a_{n-\frac{3}{2}} & a_{n-\frac{3}{2}}+a_{n-\frac{1}{2}} & -a_{n-\frac{1}{2}} \\ & & & -a_{n-\frac{1}{2}} & a_{n-\frac{1}{2}} \end{pmatrix}\begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_{n-1} \\ U_n \end{pmatrix}$$

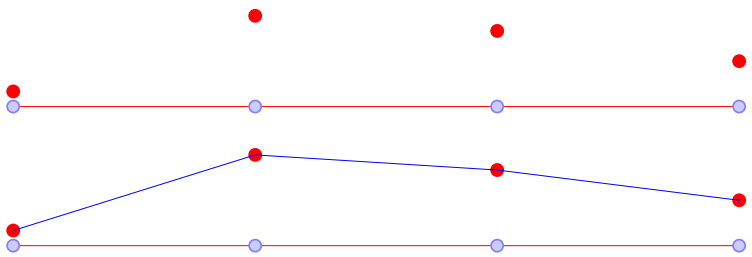
$$=\begin{pmatrix} \frac{a_{\frac{3}{2}}}{a_{\frac{3}{2}}-\frac{1}{3}a_{\frac{1}{2}}}(f_1-\frac{1}{h}a_{\frac{1}{2}}\frac{2}{3}\sigma_0)+\left(\frac{a_{\frac{3}{2}}}{a_{\frac{3}{2}}-\frac{1}{3}a_{\frac{1}{2}}}\right)^2 b \\ f_2+b \\ \vdots \\ f_{n-1}+b \\ \frac{a_{n-\frac{1}{2}}}{a_{n-\frac{1}{2}}-\frac{1}{3}a_{n+\frac{1}{2}}}(f_n+\frac{1}{h}a_{\frac{n+1}{2}}\frac{2}{3}\sigma_1)+\left(\frac{a_{n-\frac{1}{2}}}{a_{n-\frac{1}{2}}-\frac{1}{3}a_{n+\frac{1}{2}}}\right)^2 b \end{pmatrix}.$$

3

A brief introduction of finite element methods

3.1 Motivation and plans

Finite difference method can only be used on a rectangular domain or a domain which can be transformed to a rectangle such as a disk via polar coordinates. For complicated problems in real applications, finite element method is the most successful approach due to its rich theories and flexibility with geometries. In this notes, we mainly focus on rectangular domains, on which finite difference method is the most convenient choice. On the other hand, the finite element method on a rectangular domain can be implemented as a finite difference method when integrals are replaced with quadrature. For instance, there is no essential difference between grid point values and piece-wise linear polynomials represented by its grid point values. Even on unstructured triangular meshes, a linear polynomial on a triangle can be represented by its point values on three vertices of the triangle, which is often called *nodal representation*.



The conventional approach of constructing a finite difference method that we have seen in Chapter 2 includes two crucial steps: first, develop a consistent discretization or approximation to the differential operator and the boundary conditions then try to establish the stability, i.e., try to show $\|A^{-1}\| \leq C$ if the matrix-vector form of the scheme is $A\mathbf{u} = \mathbf{f}$. While the

first step is a relative easy task, the second step seems to be fine with the second order centered difference because we have eigenvalues and eigenvectors, which is however nearly impossible to find for higher order accurate schemes and more general problems such as variable coefficient problems. In general there are quite a few drawbacks and challenges in such a traditional approach. The key issues include:

- It is not elegant or convenient to design a high order accurate scheme to use Taylor expansion (for instance, do you actually enjoy solving Problem 2.17). It also becomes harder to deal with boundary conditions in high order schemes.

- Stability is hard to establish in general: estimating the inverse of a matrix is always hard. If using only linear algebra, singular values and eigenvalues are impossible to estimate for more general schemes (think about a high order accurate scheme for $-(a(x)u')' = f$).

Remark 3.1. *9-point discrete Laplacian is successful, but only for Laplacian operator on uniform meshes with Dirichlet boundary conditions.*

Moreover, there are practical concerns:

- Loss of accuracy on non-uniform meshes: if the local truncation error is obtained by Taylor expansion on uniform grids, the proof of order of accuracy breaks down regardless of whether the actual scheme is still as accurate as on uniform grids or not.

- Loss of symmetry in the matrix A: the matrix in general is not symmetric and hard to symmetrize (think about $-\nabla(a(x,y)\nabla u) = f$ with Neumann boundary conditions).

Remark 3.2. *One of the main reasons why a symmetric A is much better is for purely Neumann boundary conditions. The exact solution is not unique for purely Neumann b.c.. So A in the numerical scheme $A\mathbf{u} = \mathbf{f}$ is not invertible thus the linear system $A\mathbf{u} = \mathbf{f}$ may not have a solution (unless $\mathbf{f}$ happens to lie in the column space of A, which is usually not true). So as we have seen in Section 2.11.3, one would have to instead consider $A\mathbf{u} = \bar{\mathbf{f}}$ where $\bar{\mathbf{f}}$ is the projection of $\mathbf{f}$ onto the column space of A. The left null vector $\mathbf{v}$ (i.e.. $\mathbf{v}^T A = \mathbf{0}$) is needed for such a projection. If A is symmetric, then the left null vector is also the right null vector, which is usually $\begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ since A approximates a differential operator. If A is not symmetric, then one would have to solve an eigenvalue problem $A^T\mathbf{v} = 0 * \mathbf{v}$ which is even more expensive (at least 2-3 times more expensive) than solving $A\mathbf{u} = \bar{\mathbf{f}}$. The difficulty of using a non-symmetric matrix for purely Neumann boundary conditions will also be explained in Section 3.9.*

Remark 3.3. *For solving $A\mathbf{u} = \bar{\mathbf{f}}$ as above, it is mathematically equivalent to solve the least square solution by solving $A^T A\mathbf{u} = A^T\mathbf{f}$, which is however a lot harder to solve numerically, because the condition number of $A^T A$ will be nearly the square of the condition number of A.*

All these concerns and difficulties can be solved by using finite element method! What is even better is that finite element method on rectangular meshes (or regular triangular meshes) looks like exactly a finite difference method. In this chapter, we will first see how a finite element method is defined then implement it as a finite difference method.

Caution to readers: this is a very brief introduction to the finite element method because

- We will give up certain math rigor such as complete definition of distribution and Sobolev spaces, proof of existence and uniqueness of variational formulation and important estimates. Instead they will be given and stated as facts.
- We focus mainly on rectangular domains and rectangular meshes.

Despite of these simplifications in mind, you will still learn and understand the key ingredients of the finite element method.

3.2 Preliminaries

3.2.1 Weak derivatives and Sobolev spaces

Let $C_0^\infty(\mathbb{R})$ be the set of all infinitely differentiable functions which are nonzero only on a finite interval.

If a function $f(x)$ is differentible, then after integration by parts, for any smooth function $v(x) \in C_0^\infty(\mathbb{R})$, we have

$$\int_{-\infty}^{+\infty} f(x)v'(x)dx = -\int_{-\infty}^{+\infty} f'(x)v(x)dx. \tag{3.1}$$

The function $f(x) = |x|$ is not differentiable but we can define its *weak or generalized derivative* as the step function $g(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$ in the sense of (3.1):

Definition 3.1. *A function $g(x)$ is defined to be the weak or generalized derivative of $f(x)$ if it satisfies*

$$\int_{-\infty}^{+\infty} f(x)v'(x)dx = -\int_{-\infty}^{+\infty} g(x)v(x)dx, \quad \forall v(x) \in C_0^\infty(\mathbb{R}).$$

Example 3.1. *It is straightforward to verify that the step function is the weak derivative of the absolute value function.*

If a function is differentible, then its weak derivative is simply the derivative. **From now on, in this Chapter, derivatives are understood as generalized derivatives.**

Next we need to define a few spaces:

- $$L^2([0,1]) = \left\{ f(x) : \int_0^1 f(x)^2 dx < \infty \right\}.$$

  For a general domain Ω, $L^2(\Omega)$ is similarly defined. Here integral is the Lebesgue integral if you know what it means. The $L^2(\Omega)$-norm will be denoted as

  $$\|f\|_{0,\Omega} = \|f\|_{L^2(\Omega)} = \left(\int_\Omega f(x)^2 dx \right)^{\frac{1}{2}}.$$

  When there is no confusion, we will drop Ω in the subscript, e.g., $\|f\|_0$ simply denotes the L^2-norm.

- $$H^1([0,1]) := \left\{ f(x), f'(x) \in L^2 : \int_0^1 [f(x)^2 + f'(x)^2] dx < \infty \right\}.$$

  The $H^1(\Omega)$-norm will be denoted as

  $$\|f\|_{1,\Omega} = \|f\|_{H^1(\Omega)} = \left(\int_\Omega [f(x)^2 + f'(x)^2] dx \right)^{\frac{1}{2}}.$$

  We also define a seminorm:

  $$|f|_{1,\Omega} = |f|_{H^1(\Omega)} = \left(\int_\Omega f'(x)^2 dx \right)^{\frac{1}{2}}.$$

  When there is no confusion, we will drop Ω in the subscript, e.g., $\|f\|_1$ simply denotes the H^1-norm.

  Fact: in one dimension, $H^1([0,1]) \subset C([0,1])$.

- $H_0^1([0,1])$ is the subset $H^1([0,1])$ with the property of vanishing at the boundary.

- H^2 space is similarly defined:

  $$H^2([0,1]) = \left\{ f(x), f'(x), f''(x) \in L^2 : \int_0^1 [f(x)^2 + f'(x)^2 + f''(x)^2] dx < \infty \right\}.$$

  Norm and semi-norm are

  $$\|f\|_{2,\Omega} = \|f\|_{H^2(\Omega)} = \left(\int_\Omega [f(x)^2 + f'(x)^2 + f''(x)^2] dx \right)^{\frac{1}{2}},$$

  $$|f|_{2,\Omega} = |f|_{H^2(\Omega)} = \left(\int_\Omega f''(x)^2 dx \right)^{\frac{1}{2}}.$$

- H^3 space and its norm are also similarly defined: just add $f'''(x)$.

More about continuity:

- The most general statement is from general Sobolev inequalities [3], which imply for a bounded open set $\Omega \subset \mathbb{R}^n$ with a C^1 boundary:

$$k > \frac{n}{2}, f(\mathbf{x}) \in H^k(\Omega) \Longrightarrow f(\mathbf{x}) \in C(\bar{\Omega}).$$

- The special case for one dimension: $f(x) \in H^1(-1,1) \Longrightarrow f(x) \in C[-1,1]$.
- Two dimensions: $f(x,y) \in H^2(\Omega) \Longrightarrow f(x,y) \in C(\bar{\Omega})$.
- Three dimensions: $f(x,y,z) \in H^2(\Omega) \Longrightarrow f(x,y,z) \in C(\bar{\Omega})$.
- In two dimensions, H^1 is not enough for continuity: consider Ω as a disk centered at the origin with radius $R = \frac{1}{2}$, then the following function cannot be made continous or even bounded by changing any point values:

$$f(x,y) = \left(-\log(x^2+y^2)\right)^{\alpha} \in H^1(\Omega), f(x,y) \notin C(\Omega),$$

where $\alpha \in (0, \frac{1}{2})$ is a constant. Let $r = \sqrt{x^2+y^2}$, we first have

$$\iint_{\Omega} |f|^2 dxdy = \int_{r=0}^{\frac{1}{2}} \int_{\theta=0}^{2\pi} [-\log r^2]^{2\alpha} r dr d\theta \leq C$$

because $[-\log r^2]^{2\alpha} r$ is bounded and continuous on $r \in [0, \frac{1}{2}]$. Then

$$|\nabla f| = \alpha \left(-\log r^2\right)^{\alpha-1} \frac{1}{r} = C(-\log r)^{\alpha-1} r^{-1}.$$

$$\iint_{\Omega} |\nabla f|^2 dxdy = \int_{r=0}^{\frac{1}{2}} \int_{\theta=0}^{2\pi} C(-\log r)^{2\alpha-2} r^{-2} r dr d\theta$$

$$= C \int_{r=0}^{\frac{1}{2}} (-\log r)^{2\alpha-2} r^{-1} dr (t = -\log r) = -C \int_{t=-\log\frac{1}{2}}^{+\infty} t^{2\alpha-2} dt < +\infty.$$

3.2.2 Interpolation and quadrature

Finite element methods are built upon basic tools including interpolation and quadrature (numerical integration).

- Lagrange interpolation is a convenient polynomial approximation to a function through its point values: given $k+1$ point values of $f(x)$ at $k+1$ grid points x_i $(i = 1, 2, \cdots, k+1)$, there is a unique polynomial $p(x)$ of degree k to satisfy $p(x_i) = f(x_i)$ $(i = 1, 2, \cdots, k+1)$.

The linear Lagrange interpolation at x_i, x_{i+1} for a function $f(x)$ is given by

$$\frac{x - x_{i+1}}{x_i - x_{i+1}} f_i + \frac{x - x_i}{x_{i+1} - x_i} f_{i+1}.$$

The quadratic Lagrange interpolation at x_{i-1}, x_i, x_{i+1} for a function $f(x)$ is given by

$$\frac{(x - x_i)(x - x_{i+1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} f_{i-1} + \frac{(x - x_{i-1})(x - x_{i+1})}{(x_i - x_{i-1})(x_i - x_{i+1})} f_i + \frac{(x - x_i)(x - x_{i-1})}{(x_{i+1} - x_i)(x_{i+1} - x_{i-1})} f_{i+1}.$$

- Quadrature means numerical integration, which is to approximate integrals on computer.

$$\text{Trapezoidal rule} : \int_{-1}^{1} f(x)dx \approx f(-1) + f(1)$$
$$\text{Simpson's rule} : \int_{-1}^{1} f(x)dx \approx \frac{1}{3} f(-1) + \frac{4}{3} f(0) + \frac{1}{3} f(1)$$

Trapezoidal rule is exact if $f(x)$ is a linear polynomial. Simpson's rule is also 3-point Gauss-Lobatto rule or 3-point Newton-Cotes rule, which is exact if $f(x)$ is a cubic polynomial.

Consider an uniform mesh with grids $0 = x_0 < x_1 < \cdots < x_N < x_{N+1} = 1$ with spacing $h = \frac{1}{N+1}$ for the interval $[0, 1]$, which consists of $N + 1$ intervals $I_k = [x_{k-1}, x_k]$ $(k = 1, \cdots, N + 1)$. Then for each interval we can use a linear polynomial to approximate $f(x)$ if given $f_i = f(x_i)$. Let $\Pi_1 f(x)$ denote such a piecewise linear polynomial function.

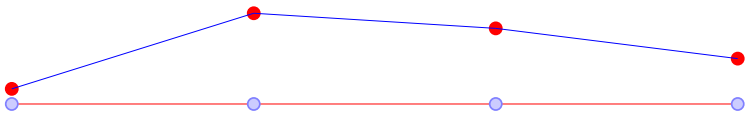


Figure 3.1: Four grid points and three intervals. For each interval, a linear polynomial is interpolated.

Next consider an uniform mesh with grids $0 = x_0 < x_1 < \cdots < x_N < x_{N+1} = 1$ with spacing $h = \frac{1}{N+1}$ for the interval $[0, 1]$. And this time we assume $N = 2n-1$ is odd. Then there are n small intervals $I_k = [x_{2k-2}, x_{2k}]$ $(k = 1, \cdots, n)$, on which we can define a piecewise quadratic interpolation polynomial, denoted by $\Pi_2 f(x)$.

Here are the facts that we will use without any proof first: for a smooth enought function $f(x)$, the interpolation error and quadrature error are given as

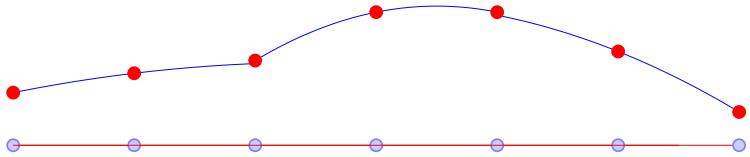


Figure 3.2: Seven grid points and three intervals. For each interval, a quadratic polynomial is interpolated.

- L^2 and H^1 errors of piecewise linear interpolation:

$$\|f - \Pi_1 f\|_0 \leq Ch^2|f|_2, \quad \|f - \Pi_1 f\|_1 \leq Ch|f|_2. \tag{3.2}$$

- L^2 and H^1 errors of piecewise quadratic interpolation:

$$\|f - \Pi_2 f\|_0 \leq Ch^3|f|_3, \quad \|f - \Pi_2 f\|_1 \leq Ch^2|f|_3. \tag{3.3}$$

- Quadrature error of trapezoidal rule for each small interval in Figure 3.1 :

$$\left| \int_0^1 f(x)dx - \sum_{k=1}^{N+1} \frac{1}{2} h[f(x_{k-1}) + f(x_k)] \right| \leq Ch^2|f|_2.$$

- Quadrature error of Simpson's rule for each small interval in Figure 3.2:

$$\left| \int_0^1 f(x)dx - \sum_{k=1}^{n} h[\frac{1}{3}f(x_{2k-1}) + \frac{4}{3}f(x_{2k}) + \frac{1}{3}f(x_{2k+1})] \right| \leq Ch^4|f|_4.$$

Notice that the estimate above only needs the minimal assumption on the function, e.g., for (3.2) we only need to assume $f(x) \in H^2(\Omega)$ (the second order derivative exists in the weak sense). The same order can be obtained by Taylor expansion, but obviously we need the derivatives to exist in the classical sense. All these estimates above can be easily derived from the Bramble-Hilbert Lemma in Section 3.7.2. On the other hand, you can simply assume these estimates are true for now.

3.3 1D BVP: homogeneous Dirichlet b.c.

3.3.1 Variational formulation

Given a function $f(x) \in L^2(0,1)$, consider solving

$$-u'' = f, \quad x \in (0,1),$$

with boundary conditions

$$u(0) = 0, \quad u(1) = 0.$$

Multiplying a test function $v \in H_0^1(0,1)$, after integration by parts, we get

$$\int_0^1 u'(x)v'(x)dx = \int_0^1 f(x)v(x)dx$$

which can be denoted as

$$(u', v') = (f, v),$$

if we define

$$(f, g) := \int_0^1 f(x)g(x)dx.$$

It can be shown that the solution to the PDE is equivalent to the solution to the following variational formulation

$$\text{seek} \quad u \in H_0^1(0,1), \quad \text{satisfying} \quad (u', v') = (f, v), \forall v \in H_0^1(0,1). \tag{3.4}$$

Theorem 3.1. *Assume* $f(x) \in C([0,1])$ *and* $u(x) \in C^2([0,1])$ *satisfies* (3.4), *then* $-u''(x) = f(x)$.

Proof. After integration by parts in (3.4), we get

$$0 = (f, v) - (u', v') = (f, v) + (u'', v) = (f + u'', v) = \int_0^1 [u''(x) + f(x)]v(x)dx.$$

If $u''(x) + f(x) \neq 0$, then due to continuity, $u''(x) + f(x)$ is either positive or negative on an interval $[x_0, x_1] \subset [0,1]$. Without loss of generality, assume $u''(x) + f(x) > 0$ on $[x_0, x_1] \subset [0,1]$. Consider a test function

$$v(x) = \begin{cases} 0, & x < x_0 \\ (x - x_0)^2(x - x_1)^2, & x \in [x_0, x_1] \\ 0, & x > x_1 \end{cases},$$

and we have

$$\int_0^1 [u''(x) + f(x)]v(x)dx = \int_{x_0}^{x_1} [u''(x) + f(x)]v(x)dx > 0,$$

which is a contradiction. □

Why the variational formulation implies the PDE is one big step that we choose to skip. If this is your first time to learn finite element method, it is the best to accept this fact without spending time pursuing why. But if this is your fifth or even tenth time to read this chapter, it might be a good time to start to learn why it should be true, in a different book! Of course the solution of (3.4) is also the solution to the PDE only when it is a solution with a second order derivative at least in the weak sense. It can be shown that the solution of (3.4) has weak second order derivative, which is called *elliptic regularity theorem*.

Consider the 1D variable coefficient problem

$$-(a(x)u')' = f, \quad x \in (0,1), \quad u(0) = u(1) = 0, \tag{3.5}$$

where $a(x) > 0$ is a smooth coefficient, with boundary conditions.

We can introduce a new notation called bilinear form

$$\mathcal{A}(u,v) := \int_0^1 au'v'dx,$$

then the equivalent variational formulation is

$$\text{seek} \quad u \in H_0^1(0,1), \quad \text{satisfying} \quad \mathcal{A}(u,v) = (f,v), \forall v \in H_0^1(0,1). \tag{3.6}$$

3.3.2 The abstract finite element method

Given a mesh with n intervals I_j ($j = 1, \cdots, n$), let V^h denote the continuous piecewise polynomial of degree k approximation to the space $H^1(0,1)$:

$$V^h := \{v_h(x) \in C(0,1) : v_h(x) \text{ is polynomial of degree } k \text{ on each interval } I_j\}.$$

We will only consider $k = 1$ or $k = 2$, i.e., linear or quadratic polynomial approximation. In general, these small intervals I_j do not have to be uniform. But for convenience and also for the sake of constructing a finite difference scheme on a uniform mesh, let us assume they have an uniform interval size. Then Figure 3.1 and Figure 3.2 are illustrations of elements in V^h.

The space V_0^h is similarly defined as an approximation to $H_0^1(0,1)$:

$$V_0^h := \{v_h(x) \in C(0,1) : v_h(0) = v_h(1) = 0, v_h(x) \in P^k(I_j), \forall j\}.$$

A continous piecewise polynomial can have a weak derivative as defined by Definition 3.1, which is the piecewise derivative inside each interval, just like that the weak derivative of $f(x) = |x|$ is the step function. Thus we have the following fact:

$$V^h \subset H^1(0,1), \quad V_0^h \subset H_0^1(0,1).$$

Given V_0^h, the abstract finite element method for (3.6) is defined as

$$\text{seek} \quad u_h \in V_0^h, \quad \text{satisfying} \quad \mathcal{A}(u_h, v_h) = (f, v_h), \forall v_h \in V_0^h. \tag{3.7}$$

We call (3.7) the abstract finite element method because it can never be exactly implemented. For example, the right hand side integral (f, v_h) can never be computed exactly, unless $f(x)$ is a very simple function.

3.3.3 The abstract implementation

Assume we know how to compute all integrals in (3.7), e.g., if the coefficient $a(x) \equiv 1$ and $f(x)$ is a polynomial in (3.6), then all integrands are polynomials. Then let us think about how the scheme (3.7) should be implemented, e.g., in the scheme (3.7) what does arbitrariness of the test function v_h mean?

First of all, once the mesh is fixed and polynomial degree is fixed, the piecewise polynomial space V_0^h is a finite dimensional vector space. Assume it is N-dimensional with basis functions $\{\phi_i(x) : i = 1, \cdots, N\}$.

Second, in the scheme (3.7), both the left hand side and the right hand side are linear operators with respect to the test function v_h. Therefore, $A(u_h, v_h) = (f, v_h)$ for arbitrary test function v_h in an N-dimensional vector space V_0^h is equivalent to $A(u_h, v_h) = (f, v_h)$ for v_h being all basis functions $\phi_i(x)$. Namely, (3.7) is equivalent to

$$\mathcal{A}(u_h, \phi_i) = (f, \phi_i), \quad i = 1, \cdots, N.$$

Third, $u_h \in V_0^h$ implies that u_h is a linear combination of the basis functions:

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

Next, plugging in $u_h(x) = \sum_{j=1}^{N} u_j \phi_j(x)$ and using the linearity of the bilinear form $\mathcal{A}$, we get that

$$\sum_{j=1}^{N} u_j \mathcal{A}(\phi_j, \phi_i) = (f, \phi_i), \quad i = 1, \cdots, N,$$

which is a system of N linear equations.

The last step is to solve a linear system $S\mathbf{u} = \mathbf{f}$ where the *stiffness* matrix S has entries $S_{ij} = \mathcal{A}(\phi_j, \phi_i)$, and

$$\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} (f, \phi_1) \\ (f, \phi_2) \\ \vdots \\ (f, \phi_N) \end{pmatrix}.$$

3.3.4 The simple practical implementation on uniform meshes

To implement the scheme (3.7), one needs to address the issue of how to compute integrals. One convenient choice is to use quadrature. Let us use trapezoidal rule for P^1 method (and Simpson's rule for P^2 method). Let $\mathcal{A}_h(\cdot, \cdot)$ and $\langle f, v_h \rangle_h$ denote the quadrature approximation to $\mathcal{A}(\cdot, \cdot)$ and (f, v_h) respectively. Then we get a new scheme

$$\text{seek} \quad u_h \in V_0^h, \quad \text{satisfying} \quad \mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \forall v_h \in V_0^h. \tag{3.8}$$

Recall that for both P^1 mesh Figure 3.1 and P^2 mesh 3.2, there are N interior grid points. Let $\phi_i(x)$ $(i = 1, \cdots, N)$ denote the basis functions in V_0^h satisfying

$$\phi_i(x_j) = \delta_{ij} = \begin{cases} 1 & i = j, \\ 0 & i \neq j, \end{cases}, \forall j = 1, \cdots, N.$$

This kind of basis is often called *Lagrangian basis* or *nodal basis*. For instance, $\phi_i(x)$ for P^1 method is given as

$$\phi_i(x) = \begin{cases} \frac{1}{h}(x - x_{i-1}), & x \in [x_{i-1}, x_i], \\ \frac{1}{h}(x_{i+1} - x), & x \in [x_i, x_{i+1}], \\ 0, & \text{otherwise}, \end{cases}$$

and its weak derivative is

$$\phi_i'(x) = \begin{cases} \frac{1}{h}, & x \in [x_{i-1}, x_i], \\ -\frac{1}{h}, & x \in [x_i, x_{i+1}], \\ 0, & \text{otherwise}. \end{cases}$$

For Lagrangian basis $\phi_i(x)$, if we set $u_j = u_h(x_j)$, then

$$u_h(x) = \sum_{i=1}^{N} u_j \phi_j(x),$$

thus the numerical solution u_h can also be denoted as a vector of point values

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix}.$$

Plugging $\sum_{j=1}^{N} u_j \phi_j(x, y)$ into the bilinear form, we get

$$\mathcal{A}_h(u_h, v_h) = \sum_{j=1}^{N} u_j \mathcal{A}_h(\phi_i(x), v_h).$$

Since it suffices to ask $\mathcal{A}_h(u_h, v_h)$ to hold for $v_h = \phi_i$ for all i, the scheme (3.8) is equivalent to

$$\text{seek } \mathbf{u} \in \mathbb{R}^N, \quad \text{satisfying } \sum_{j=1}^{N} \mathcal{A}_h(\phi_j(x), \phi_i(x)) u_j = \langle f, \phi_i(x) \rangle_h, \forall i = 1, \cdots, N. \tag{3.9}$$

The right hand side can be explicitly written as

$$\langle f, \phi_i(x)\rangle_h = \sum_{k=0}^{N} \frac{1}{2} h[f(x_k)\phi_i(x_k) + f(x_{k+1})\phi_i(x_{k+1})] = f_i h.$$

So the matrix vector form of (3.9) is $S\mathbf{u} = h\mathbf{f}$ where the stiffness matrix S has its (i, j)-th entry as

$$S_{ij} = \mathcal{A}_h(\phi_j(x), \phi_i(x)).$$

Consider the simplest Laplacian case $a(x) \equiv 1$, then

$$S_{ij} = \mathcal{A}_h(\phi_j(x), \phi_i(x)) = \langle \phi_j'(x), \phi_i'(x)\rangle_h = \begin{cases} \frac{2}{h} & i = j \\ -\frac{1}{h} & i = j \pm 1 \\ 0 & \text{otherwise.} \end{cases}.$$

In other words, for solving $-u'' = f, u(0) = u(1) = 0$, the matrix vector form of the P^1 finite element method with trapezoidal quadrature is precisely the second order centered difference:

$$\frac{1}{h}\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = h \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{N-1} \\ f_N \end{bmatrix}.$$

For the variable coefficient problem $-(au')' = f, u(0) = u(1) = 0$, similarly we can derive the matrix vector form for the scheme (3.9) with piecewise linear basis:

$$\frac{1}{h}\frac{1}{2}\begin{pmatrix} a_0 + 2a_1 + a_2 & -a_1 - a_2 & & & \\ -a_1 - a_2 & a_1 + 2a_2 + a_3 & -a_2 - a_3 & & \\ & & & & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \\ \vdots \end{bmatrix} = h \begin{bmatrix} f_1 \\ f_2 \\ \\ \vdots \end{bmatrix} \tag{3.10}$$

Recall that the traditional finite difference scheme (2.8) in Chapter 2 is given as

$$\frac{1}{\Delta x^2}\begin{pmatrix} a_{\frac{1}{2}} + a_{\frac{3}{2}} & -a_{\frac{3}{2}} & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}} + a_{\frac{5}{2}} & -a_{\frac{5}{2}} & \\ & \ddots & \ddots & \ddots \end{pmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix},$$

and the matrix can be easily written as $B = \frac{1}{\Delta x^2} D^T A D$, where A be a diagonal matrix with diagonal entries $a_{\frac{1}{2}}, \cdots, a_{n+\frac{1}{2}}$ and

$$D = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ 0 & \ddots & \ddots & \\ & & -1 & 1 \\ & & & -1 \end{bmatrix}_{(n+1)\times n}.$$

Notice that the two schemes (2.8) and (3.10) would be exactly the same if we use an approximation $a_{j+\frac{1}{2}} \approx \frac{a_j + a_{j+1}}{2}$ for the mid point values of $a(x)$ in (2.8). For smooth $a(x)$, the approximation $a_{j+\frac{1}{2}} \approx \frac{a_j + a_{j+1}}{2}$ is second order accurate by Taylor expansion. Because of this, the stiffness matrix S in the finite element method (3.10) can be easily written as

$$S = \frac{1}{2}\frac{1}{h} D^T A D$$

where A is a diagonal matrix with diagonal entries $a_0+a_1, a_1+a_2, a_2+a_3, \cdots$.

Problem 3.1. *Are there any alternatives to compute or approximate integrals in* (3.7) *if we do not use quadrature?*

Problem 3.2. *For the variable coefficient problem* $-(au')' = f, u(0) = u(1) = 0$, *derive the equivalent matrix vector form* (3.10) *for the scheme* (3.9) *with piecewise linear basis.*

Problem 3.3. *Derive the basis functions* $\phi_i(x)$ *for the* P^2 *method and find the explicit matrix vector form of the scheme* (3.9) *for the* $-u'' = f, u(0) = u(1) = 0$.

Problem 3.4. *Implement both schemes* (2.8) *and* (3.10), *and compare their errors for a problem with a smooth solution* u *for a smooth coefficient* $a(x)$.

Problem 3.5. *For a rectangular domain* Ω, *consider a 2D variable coefficient problem*

$$-\nabla \cdot (a(\mathbf{x})\nabla u(\mathbf{x})) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega$$

with homogeneous Dirichlet boundary condition, where $a(\mathbf{x}) > 0$ *is a scalar coefficient. Consider a uniform rectangular mesh and using* Q^1 *finite element method with trapezoidal quadrature for both* x *and* y *variables. The finite element method is to seek* $u_h \in V_0^h$ *satisfying*

$$\mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle.$$

Using notation in Chapter 2, the scheme can be written as

$$\left[\frac{1}{\Delta x^2}(D_x^T \otimes I_y)A_1(D_x \otimes I_y) + \frac{1}{\Delta y^2}(I_x \otimes D_y^T)A_2(I_x \otimes D_y)\right] vec(U) = vec(F),$$

where A_1 and A_2 are two diagonal matrices defined as follows.

Let a_1 be a 2D array of size $Ny \times (Nx+1)$ satisfying $a_1(j,i) = \frac{1}{2}a(x_i, y_j) + \frac{1}{2}a(x_{i-1}, y_j)$ and a_2 be a 2D array of size $(Ny+1) \times Nx$ satisfying $a_2(j,i) = \frac{1}{2}a(x_i, y_j) + \frac{1}{2}a(x_i, y_{j-1})$. Then A_1 and A_2 can be easily generated in MATLAB as sparse diagonal matrices:

```
A1=sparse(diag(a1(:)));
A2=sparse(diag(a2(:)));
```

Implement this scheme and test the accuracy for a smooth solution.

3.4 Basic properties of the bilinear form

3.4.1 Coercivity

We consider the bilinear form $\mathcal{A}(u,v) = \int_0^1 au'v'dx$ with the smooth coefficient $a(x)$ satisfying $0 \le \min_{x\in[0,1]} a(x) \le a(x) \le \max_{x\in[0,1]} a(x) < +\infty$ for any x. The first useful concept is called *coercivity* of the bilinear form:

$$\forall v \in H_0^1(\Omega), \quad \mathcal{A}(v,v) = \int_0^1 a(x)v'v'dx \ge \min_x a(x)|v|_1^2 \ge C\|v\|_1^2,$$

where $C > 0$ is a constant.

To establish the coercivity, we have used the fact that the H^1-seminorm $|\cdot|_{1,\Omega}$ and the H^1-norm $\|\cdot\|_{1,\Omega}$ are equivalent in $H_0^1(\Omega)$, i.e., there is a constant $C > 0$ depending only on Ω s.t. for any

$$\forall v \in H_0^1(\Omega), \quad C\|v\|_{1,\Omega}^2 \le |v|_{1,\Omega}^2 \le \|v\|_{1,\Omega}^2. \tag{3.11}$$

The second inequality in (3.11) is trivial. The first inequalilty in (3.11) simply says that the function value can be controlled by the derivatives, which is in general not true. For example, if $v(x) \equiv 1$ on $\Omega = [0,1]$, then $\|v\|_0 = 1$ and $|v|_1 = \|v'(x)\|_0 = 0$ thus the first inequalilty cannot hold for $v \notin H_0^1(\Omega)$.

For $\Omega = (0,1)$, here are some quick arguments to see why it is even possible to control function values by derivatives for $v(x) \in H_0^1(0,1)$. If $v(0) = 0$ and $v'(x)$ exists everywhere in the classical sense, then by the Mean Value Theorem we have $\frac{v(x)-v(0)}{x-0} = v'(y)$ for some $y \in (0,x)$, thus $v(x) = xv'(y)$ and $|v(x)| \le |v'(y)|$ for any $x \in [0,1]$. You can simply assume (3.11) is true for now, and read *Poincaré inequalilty* in the Appendix for a rigorous statement.

Remark 3.4. *The estimates in* (3.11) *hold even for a function $v(x)$ which vanishes only along a part or a very small part of the boundary of Ω.*

3.4.2 Continuity

The *continuity* of the bilinear form is simple implication of Cauchy Schwartz inequality:

$$\forall u, v \in H^1(\Omega), \mathcal{A}(u,v) = \int_0^1 au'v'dx \leq \max_x a(x) \int_0^1 u'v'dx$$

$$\leq \max_x a(x) \sqrt{\int_0^1 [u']^2 dx} \sqrt{\int_0^1 [v']^2 dx} \leq C\|u\|_1 \|v\|_1.$$

3.4.3 Coercivity is stability

Recall that the abstract finite element method can be casted as a linear system:

$$\sum_{j=1}^{N} u_j \mathcal{A}(\phi_j, \phi_i) = (f, \phi_i), \quad i = 1, \cdots, N.$$

Whenever a scheme is given as a linear system for an elliptic equation, it must be addressed whether the linear system has a solution. In other words, we need to show the solvability, i.e., the invertability of the *stiffness* matrix S with entries $S_{ij} = \mathcal{A}(\phi_j(x), \phi_i)$. In Chapter 2, we computed eigenvalues of K matrix for constant coefficient problems so that we can show the nonsingularity of the matrix K.

Obviously, for a variable coefficient problem, e.g., the scheme (3.10), the eigenvalues of the stiffness matrix should be estimated rather than computed because it is nearly impossible to compute. Such an eigenvalue estimate can be done by the coercivity. In this section we only focus on the bilinear form $\mathcal{A}$ and we will discuss $\mathcal{A}_h$ later. For any $v_h \in V_0^h$, we have

$$\mathcal{A}(v_h, v_h) = \mathcal{A}(\sum_{i=1}^{n} v_i\phi_i, \sum_{j=1}^{n} v_j\phi_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} \mathcal{A}(\phi_j, \phi_i) v_j v_i = \mathbf{v}^T S \mathbf{v}.$$

The coercivity says that

$$\mathcal{A}(v,v) \geq C_1\|v\|_1^2 \geq C_1\|v\|_0^2, \quad \forall v(x) \in H_0^1(\Omega),$$

where the constant C_1 only depends on the domain Ω. Thus

$$v_h(x) \in V_0^h \subset H_0^1(\Omega) \Rightarrow \mathcal{A}(v_h, v_h) \geq C_1\|v_h\|_0^2.$$

Notice that $\|v_h\|_0$ and $\|\mathbf{v}\|$ are both norms of the same finite dimensional vector space V_0^h, thus they are equivalent:

$$C_2\|\mathbf{v}\|^2 \leq \|v_h\|_0^2 \leq C_3\|\mathbf{v}\|^2,$$

where the constants C_2, C_3 depends on the dimension N of the vector space V_0^h.

Thus coercivity gives us

$$\mathbf{v}^T S\mathbf{v} = \mathcal{A}(v_h, v_h) \geq C_1\|v_h\|_0^2 \geq C_1C_2\|\mathbf{v}\|^2 \Rightarrow \frac{\mathbf{v}^T S\mathbf{v}}{\mathbf{v}^T\mathbf{v}} \geq C_1C_2.$$

Recall that $\mathcal{A}(u,v) = \int_0^1 a(x)u'(x)v'(x)dx$, thus $\mathcal{A}(u,v) = \mathcal{A}(v,u)$ implies that S is real symmetric. By the Courant-Fisher-Weyl min-max principle (see Appendix A.1), $\frac{\mathbf{v}^T S\mathbf{v}}{\mathbf{v}^T\mathbf{v}} \geq C_1C_2$ implies that the smallest eigenvalue of S is greater than or equal to $C_1C_2 > 0$. Therefore S is invertible.

In Section 3.6.4, we will further show that we can get a useful estimate for $\|S^{-1}\|$ from the coercivity.

3.5 Error estimates of the abstract finite element method

We have just seen that the finite element method with linear basis and quadrature recovers the K matrix for approximating second order derivative. Even though a finite element method with quadrature becomes a finite difference scheme, it is much more useful to understand the same scheme from the finite element perspective. In Chapter 2, we had to find eigenvalues of K to discuss the stability thus convergence rate for the FD scheme $\frac{1}{h^2}K\mathbf{u} = \mathbf{f}$. Obviously, eigenvalues of a matrix are difficult to find in general, e,g., the matrix (3.10).

We first prove the error estimates for (3.7), i.e., the finite element method without quadrature. We will focus on the analysis for P^1 finite element method for the one-dimensional problem with homogeneous Dirichlet boundary conditions to understand the key components in the finite element method anaylysis, but keep in mind that all discussions in this section apply to much more general cases such as solving variable coefficient elliptic equations by P^k polynomial finite element method with Neumann boundary conditions on unstructured meshes for a curved domain in multiple dimensions.

3.5.1 H^1-norm estimate: stability and consistency imply convergence

The continuous piecewise polynomial has weak derivatives, thus we have $V_0^h \subset H_0^1(\Omega)$, and such a finite element method is called *conforming*. Here is one simple example of nonconforming finite element space for the Poisson equation: the discontinuous piecewise polynomial space is not a subspace of the $H_0^1(\Omega)$ function space.

Theorem 3.2 (Galerkin Orthogonality)**.** *Let u be the solution to* (3.6)*. The solution u_h to the conforming finite element method* (3.7) *satisfies:*

$$\mathcal{A}(u - u_h, w_h) = 0, \forall w_h \in V_0^h.$$

Proof. The exact solution u satisfies

$$\mathcal{A}(u, w) = (f, w), \forall w \in H_0^1(\Omega)$$

thus

$$\mathcal{A}(u, w_h) = (f, w_h), \forall w_h \in V_0^h \subset H_0^1(\Omega).$$

The numerical solution u_h satisfies

$$\mathcal{A}(u_h, w_h) = (f, w_h), \forall w_h \in V_0^h.$$

Subtracting these two equations, we get Galerkin Orthogonality, which is a straightforward implication from the choice of approximation space $V_0^h \subset H_0^1(\Omega)$. □

Galerkin Orthogonality simply says that the true error $u - u_h$ is somehow "orthogonal" to any test function w_h in V_0^h through the bilinear form $\mathcal{A}$.

Remark 3.5. *Galerkin Orthogonality is the analog of the consistency or truncation error in Chapter 2. Since* $\mathcal{A}(u_h, w_h) = (f, w_h)$*,*

$$\mathcal{A}(u - u_h, w_h) = \mathcal{A}(u, w_h) - \mathcal{A}(u_h, w_h) = \mathcal{A}(u, w_h) - (f, w_h).$$

So Galerkin Orthogonality is the same as

$$\mathcal{A}(u, w_h) - (f, w_h) = 0, \forall w_h \in V_0^h,$$

which is nothing but replacing u_h *by* u *in the numerical scheme. On the other hand, it seems that the "truncation error" is zero here, which is due to the direct approximation of the variational form. Notice that the "truncation error"* $A(u, w_h) - (f, w_h)$ *is zero only when the test functions are in the approximated space* V_0^h*.*

Theorem 3.3 (Céa's Lemma)**.** *Let* u *be the solution to* (3.6)*. The solution* u_h *to the conforming finite element method* (3.7) *satisfies:*

$$\|u - u_h\|_1 \leq C \inf_{w_h \in V_0^h} \|u - w_h\|_1.$$

Proof. First of all, we have $u_h \in V_0^h \subset H_0^1(\Omega)$ thus $u - u_h \in H_0^1(\Omega)$. The coercivity implies

$$C\|u - u_h\|_1^2 \leq \mathcal{A}(u - u_h, u - u_h).$$

Next, we have

$$\mathcal{A}(u - u_h, u - u_h) = \mathcal{A}(u - u_h, w_h - u_h) + \mathcal{A}(u - u_h, u - w_h).$$

Galerkin orthogonality implies $\mathcal{A}(u-u_h, w_h-u_h)=0$. So we get

$$\mathcal{A}(u-u_h, u-u_h)=\mathcal{A}(u-u_h, u-w_h)\le C\|u-u_h\|_1\|u-w_h\|_1,$$

where continuity is used.

Finally, we have

$$C\|u-u_h\|_1^2\le \mathcal{A}(u-u_h, u-u_h)=\mathcal{A}(u-u_h, u-w_h)\le C\|u-u_h\|_1\|u-w_h\|_1,$$

thus

$$\|u-u_h\|_1^2\le C\|u-u_h\|_1\|u-w_h\|_1.$$

So we have $\|u-u_h\|_1\le C\|u-w_h\|_1$ for any $w_h\in V_0^h$.

□

Céa's Lemma says the finite element solution error is controlled by the best piecewise polynomial approximation error, which we do not know. On the other hand, we do know polynomial interpolation error estimates (3.2) and (3.3). Assuming $u\in H^2(\Omega)$ or $u\in H^3(\Omega)$ (i.e., assuming u is smooth enough), we easily obtain error estimate in H^1-norm:

$$\|u-u_h\|_1\le C\inf_{w_h\in V_0^h}\|u-w_h\|_1\le C\|u-\Pi_k u\|_1=\begin{cases}Ch|u|_2, & k=1\\ Ch^2|u|_3, & k=2\end{cases}. \quad (3.12)$$

Céa's Lemma gives (3.12), which is the *convergence* of the finite element method. On the other hand, Céa's Lemma is implied by both Galerkin Orthogonality (consistency) and Coercivity (stability). So we get the same conclusion as in Chapter 2 and Chapter 7 for linear schemes solving linear PDEs:

$$\text{consistency}+\text{stability}\longrightarrow\text{convergence}.$$

Recall that $u_h\in V_0^h\subset H_0^1(\Omega)$ and $u\in H_0^1(\Omega)$, thus the error function $u-u_h$ is an element in the $H_0^1(\Omega)$ space, in which H^1-norm is equivalent to the H^1-seminorm. So the H^1-norm estimate above simply implies that P^1 finite element method generates a numerical solution satisfying that

$$\sqrt{\int_0^1|u'(x)-u_h'(x)^2|dx}=\mathcal{O}(h).$$

For function values, we will get one order higher, explained in the next subsection.

3.5.2 L^2-norm estimate: elliptic regularity and duality arguments

Notice that H^1 estimate cannot explain why P^1 method gives a second order accurate scheme. Recall that H^1-norm, i.e., $\|u-u_h\|_1$, measures the error

both in the function value and the first order derivative. The L^2-norm, i.e., $\|u-u_h\|_0$ measures the error in the function value. For example, we already know that the P^1 finite element method on a uniform mesh gives exactly the standard centered finite difference, which is second order accurate for the function value.

The L^2 estimate will be one order higher than H^1 estimate:

Theorem 3.4 (Aubin-Nitsche Lemma)**.** *Let u be the solution to* (3.6)*. The solution u_h to the conforming finite element method* (3.7) *satisfies:*

$$\|u - u_h\|_0 \leq Ch\|u - u_h\|_1,$$

where h is the mesh size.

For proving the Aubin-Nitsche Lemma, we need a basic fact about the Poisson equation, which is called *elliptic regularity*: the solution to (3.6) satisfies $\|u\|_2 \leq C\|f\|_0$, which simply says that the second order derivative of u and lower order ones are controlled by function value of $f(x)$.

Even though we only seek $u(x) \in H_0^1(\Omega)$ in (3.6), the elliptic regularity theorem guarantees that $f(x) \in L^2(\Omega) \Rightarrow u(x) \in H^2(\Omega)$. In particular, if $f(x)$ is infinitely differentiable, then so is $u(x)$. The elliptic regularity can be proven under certain assumptions for the domain Ω.

We also need a *dual* problem to help us here. A dual problem of (3.6) is to find $w \in H_0^1(\Omega)$ satisfying

$$\mathcal{A}(w, v) = (u - u_h, v), \quad \forall v \in H_0^1(\Omega). \tag{3.13}$$

The equivalent PDE form of the dual problem above is

$$-w''(x) = u(x) - u_h(x),$$

if the original PDE we want to solve is $-u''(x) = f(x)$.

The elliptic regularity theorem on the dual problem gives the following

$$\|w\|_2 \leq \|u - u_h\|_0.$$

For the dual problem, its finite element solution for finding $w_h \in V_0^h$ satisfying

$$\mathcal{A}(v_h, w_h) = (u - u_h, v_h), \quad \forall v_h \in V_0^h,$$

where we have used the symmetry of the bilinear form $\mathcal{A}(w, v) = \mathcal{A}(v, w)$. By Céa's Lemma and H^1-estimate applied to the finite element solution w_h, we have

$$\|w - w_h\|_1 \leq Ch|w|_2 \leq Ch\|w\|_2.$$

where we have used the interpolation error estimate.

So with elliptic regularity for the dual problem $\|w\|_2 \leq C\|u - u_h\|_0$, we get

$$\|w - w_h\|_1 \leq Ch\|w\|_2 \leq Ch\|u - u_h\|_0.$$

Proof. Let w_h be the finite element solution for the dual problem. Then Galerkin orthogonality implies $\mathcal{A}(u-u_h, w_h) = 0$, thus

$$\mathcal{A}(u-u_h, w) = \mathcal{A}(u-u_h, w-w_h) + \mathcal{A}(u-u_h, w_h) = \mathcal{A}(u-u_h, w-w_h).$$

Continuity implies

$$\mathcal{A}(u-u_h, w-w_h) \leq C\|u-u_h\|_1\|w-w_h\|_1.$$

Recall that w is the solution to the dual problem thus plugging in $v = u - u_h \in H_0^1(\Omega)$ in (3.13), we get

$$\mathcal{A}(u-u_h, w) = \mathcal{A}(w, u-u_h) = (u-u_h, u-u_h) = \|u-u_h\|_0^2$$

Finally, putting everything together

$$\|u-u_h\|_0^2 = \mathcal{A}(u-u_h, w) \leq C\|u-u_h\|_1\|w-w_h\|_1 \leq C\|u-u_h\|_1 h\|u-u_h\|_0,$$

which gives

$$\|u-u_h\|_0 \leq Ch\|u-u_h\|_1$$

□

With the H^1-norm estimate (3.12), the Aubin-Nitsche Lemma gives us the L^2-norm error estimates:

$$\|u-u_h\|_0 \leq Ch\|u-u_h\|_1 = \begin{cases} Ch^2|u|_2, & k=1 \\ Ch^3|u|_3, & k=2 \end{cases}. \tag{3.14}$$

This is consistent with what we already know: P^1 finite element method gives a second order accurate scheme for function values. The P^2 finite element method gives a third order accurate scheme for function values, which is consistent with the interpolation error order (3.3). However, if we implement P^2 finite element method as a finite difference scheme, we can actually get a fourth order accurate finite difference scheme, which is called *superconvergence*. It will be explained in the rest of the chapter.

Remark 3.6. *In estimates like* (3.12) *and* (3.14)*, it is already assumed that* u *should be smooth enough such that either* $u \in H^2(\Omega)$ *or* $u \in H^3(\Omega)$*. The elliptic regularity theorem implies that* $f(x) \in L^2(\Omega) \Rightarrow u \in H^2(\Omega)$ *and* $f(x) \in H^1(\Omega) \Rightarrow u \in H^3(\Omega)$*.*

3.5.3 Summarization and comparison

Now let us just focus on the P^1 finite element method and think about how the second order accuracy is proven differently from the one we did in Chapter 2. In Chapter 2, we computed the eigenvalues of the K matrix for proving stability $\|A^{-1}\| \leq C$ in a matrix-vector form of the scheme $A\mathbf{u} = \mathbf{f}$.

On the one hand, it only requires simpler knowledge of linear algebra. On the other hand, it is highly restrictive because we cannot even compute eigenvalues for a one-dimensional variable coefficient problem.

The discussion in this section obviously applies to the variable coefficient problem, but we need so many much more advanced tools such as Solobev spaces and elliptic regularity. Recall how exactly we can prove the second order error in P^1 finite element method for function values:

1. The homogeneous Dirichlet boundary condition is built into the function space $H_0^1(\Omega)$, which in return gives the *Poincaré inequalilty*:

$$\int_0^1 |v'(x)|^2 dx \geq C \left[\int_0^1 |v'(x)|^2 dx + \int_0^1 |v(x)|^2 dx \right], \quad \forall v(x) \in H_0^1(\Omega).$$

2. The *Poincaié inequalilty* gives the *coercivity*

$$\mathcal{A}(v_h, v_h) \geq C \|v_h\|_0^2,$$

   which is the *stability*.

3. From the fact that it is conforming $V_0^h \subset H_0^1(\Omega)$, *Galerkin orthogonality* is easily obtained:

$$\mathcal{A}(u - u_h, v_h) = 0, \quad \forall v_h \in V_0^h.$$

   Galerkin orthogonality is the *consistency*.

4. With *Galerkin orthogonality* and *coercivity*, we get Céa's Lemma, which says the finite element solution error is controlled by the best piecewise polynomial approximation error:

$$\|u - u_h\|_1 \leq C \inf_{w_h \in V_0^h} \|u - w_h\|_1.$$

   This is step is nothing but saything that consistency and stability imply convergence.

5. We know the interpolation error using P^k polynomials, so the H^1-estimate is simply by Céa's Lemma:

$$\|u - u_h\|_1 \leq C \inf_{v_h \in V_0^h} \|u - v_h\|_1 \leq C \|u - \Pi_k u\|_1 \leq \begin{cases} Ch|u|_2, & k = 1 \\ Ch^2|u|_3, & k = 2 \end{cases}.$$

6. Finally, with the elliptic regularity on a dual problem and almost everything above, we get the Aubin-Nitsche Lemma

$$\|u - u_h\|_0 \leq Ch \|u - u_h\|_1 \leq \begin{cases} Ch^2|u|_2, & k = 1 \\ Ch^3|u|_3, & k = 2 \end{cases}.$$

3.6 V^h-ellipticity: properties of the bilinear form with quadrature

Since in practice quadrature is used to implement the finite element method, we also need to know whether coercivity and continuity hold for $\mathcal{A}_h$. Usually the discrete continuity can be easily derived from Cauchy-Schwartz inequality. The discrete coercivity is called V^h-ellipticity.

We only consider the continuous piecewise linear space V_0^h as an example in this section. Let x_i $(i = 0, 1, \cdots, N+1)$ be an uniform mesh for the whole interval $[0, 1]$, where $x_0 = 0$ and $x_{N+1} = 1$ are boundary points. The grid spacing is $h = \frac{1}{N+1}$.

3.6.1 Equivalent norms of the piecewise linear polynomial space

Everything in this subsection can be derived by abstract arguments. But instead we use some explicit elementary tools to derive what we need for coercivity.

For any $v_h \in V_0^h$, let $v_i = v_h(x_i)$ and $\mathbf{v} = \begin{bmatrix} v_1 & \cdots & v_N \end{bmatrix}^T$. So $\|v_h\|_0$ and $\|\mathbf{v}\|$ are both norms of the same finite dimensional vector space V_0^h, thus they are equivalent:

$$C_2\|\mathbf{v}\|^2 \le \|v_h\|_0^2 \le C_3\|\mathbf{v}\|^2,$$

where the constants C_2, C_3 depends on the dimension N of the vector space V_0^h.

It is useful to figure out the exact dependence of of these constants on the dimension N or the mesh size h. For the one-dimensional problem continuous piecewise linear polynomial space V_0^h on a uniform mesh with mesh size h, we have

$$\|v_h\|_0^2 = \sum_{j=0}^{N} \int_{x_j}^{x_{j+1}} |v_h(x)|^2 dx = \sum_{j=0}^{N} \int_0^h \left[\frac{v_{j+1} - v_j}{h} x + v_j\right]^2 dx$$

$$= h \sum_{j=0}^{N} \left(\frac{1}{3} v_{j+1}^2 + \frac{5}{6} v_j^2 - \frac{1}{6} v_{j+1} v_j\right).$$

Recall that $v_h(x) \in V_0^h \Rightarrow v_0 = v_{N+1} = 0$. With two simple inequalilties

$$-\frac{1}{2} v_{j+1}^2 - \frac{1}{2} v_j^2 \le -v_{j+1} v_j \le \frac{1}{2} v_{j+1}^2 + \frac{1}{2} v_j^2,$$

we can derive

$$h\|\mathbf{v}\|^2 \le \|v_h\|_0^2 \le \frac{4}{3} h \|\mathbf{v}\|^2. \tag{3.15}$$

Let us consider $v_h'(x)$, which is only piecewise constant. Recall that $v_h(x) \in V_0^h$ is weakly differentiable thus $v_h'(x_j)$ is double valued unless $j = 0, N+1$. Let $v_h'(x_j)^-$ and $v_h'(x_j)^+$ denote two values obtained by taking derivatives in the intervals $[x_{j-1}, x_j]$ and $[x_j, x_{j+1}]$ respectively. For convenience, we will also abuse the notation by denoting

$$v_h'(x_j) := \frac{v_h'(x_j)^- + v_h'(x_j)^+}{2}, \quad [v_h'(x_j)]^2 := \frac{[v_h'(x_j)^-]^2 + [v_h'(x_j)^+]^2}{2}.$$

Recall that $v_0 = v_{N+1} = 0$. Let $\mathbf{v}'$ denote the following vector

$$\mathbf{v}' = \begin{pmatrix} v_h'(x_0) \\ v_h'(x_1) \\ v_h'(x_2) \\ \vdots \\ v_h'(x_{N+1}) \end{pmatrix} = \frac{1}{h} \begin{pmatrix} v_1 - v_0 \\ \frac{v_1 - v_0 + v_2 - v_1}{2} \\ \frac{v_2 - v_1 + v_3 - v_2}{2} \\ \vdots \\ v_{N+1} - v_N \end{pmatrix} = \frac{1}{h} \begin{pmatrix} v_1 \\ \frac{v_2}{2} \\ \frac{v_3 - v_1}{2} \\ \vdots \\ -v_N \end{pmatrix},$$

Remark 3.7. *Here for $j = 1, \cdots, N$, we have $[v_h'(x_j)]^2 := \frac{[v_h'(x_j)^-]^2 + [v_h'(x_j)^+]^2}{2} = \frac{v_{j+1} - v_{j-1}}{2h}$, which of course can be regarded as the centered finite difference approximation to the first order derivative at x_j.*

Remark 3.8. *From these happy coincidences with the second order centered difference, we should see that the piecewise linear space V_0^h is the better way to understand or derive the centered difference.*

Let $\bar{V}^h$ denote the vector space of piecewise constant on the intervals I_j. Then $v_h'(x)$ corresponds to an element in $\bar{V}^h$, and obviously $\|\mathbf{v}'\|$ and $\|v_h'\|_0$ can be regarded as two norms for measuring this element in the finite dimensional vector space $\bar{V}^h$, thus they should be equivalent. However, to derive the coercivity of $\mathcal{A}_h(v_h, v_h)$, we need to be careful with the dependence of constants on the dimension N or mesh size h. Similar to (3.15), we can derive

$$\frac{1}{2} h \|\mathbf{v}'\|^2 \leq \|v_h'\|_0^2 \leq 2h \|\mathbf{v}'\|^2. \tag{3.16}$$

Problem 3.6. *Derive* (3.16)*, where $\|v_h'\|_0$ is the L^2-norm for the function $v_h'(x)$. Hint: let $\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{N+1} \end{pmatrix}$ denote the constants that $v_h'(x)$ corresponds to. Notice that the boundary condition $v_0 = v_{N+1} = 0$ implies that $\sum_{j=1}^{N+1} c_j = 0$. Then $\|v_h'\|_0^2 = h \sum_{j=1}^{N+1} c_j^2$. Derive what $\|\mathbf{v}'\|^2$ should be in terms of c_j.*

3.6.2 Coercivity

If using trapezoidal rule for P^1 finite element method in each cell $I_j = [x_j, x_{j+1}]$ in Figure 3.1, then for any $v_h(x) \in V_0^h$ we have

$$\begin{aligned}
\mathcal{A}_h(v_h, v_h) &= \sum_{j=0}^{N} \frac{h}{2} \left(a(x_j)[v_h'(x_j)^+]^2 + a(x_{j+1})[v_h'(x_{j+1})^-]^2 \right) \\
&\geq \min_j a(x_j) \sum_{j=0}^{N} \frac{h}{2} \left([v_h'(x_j)^+]^2 + [v_h'(x_{j+1})^-]^2 \right) \\
&= \min_j a(x_j) \left(\frac{h}{2}[v_h'(x_0)]^2 + h\sum_{j=1}^{N}[v_h'(x_j)]^2 + \frac{h}{2}[v_h'(x_{N+1})]^2 \right) \\
&\geq \min_j a(x_j) \frac{h}{2} \|\mathbf{v}'\|^2 \geq \min_j a(x_j) \frac{1}{4} \|v_h'(x)\|_0^2 \\
&= \min_j a(x_j) \frac{1}{4} |v_h(x)|_1^2 \geq C \min_x a(x) \frac{1}{4} \|v_h(x)\|_1^2,
\end{aligned}$$

where we have used (3.16) and (3.11) in the last two lines, and the constant C is independent of h or N.

3.6.3 Continuity

The continuity for $\mathcal{A}_h$ is straightforward: for any $w_h, v_h \in V_0^h$, we have

$$\begin{aligned}
&\mathcal{A}_h(w_h, v_h) = \sum_{j=0}^{N} \frac{h}{2} \left(a(x_j)[w_h'(x_j)^+][v_h'(x_j)^+] + a(x_{j+1})[w_h'(x_{j+1})^-][v_h'(x_{j+1})^-] \right) \\
\leq &\max_j a(x_j) \frac{h}{2} \sum_{j=0}^{N} \left(\left| [w_h'(x_j)^+][v_h'(x_j)^+] \right| + \left| [w_h'(x_{j+1})^-][v_h'(x_{j+1})^-] \right| \right) \\
\leq &\max_j a(x_j) \frac{h}{2} \sqrt{\sum_{j=0}^{N} \left([w_h'(x_j)^+]^2 + [w_h'(x_{j+1})^-]^2 \right)} \sqrt{\sum_{j=0}^{N} \left([v_h'(x_j)^+]^2 + [v_h'(x_{j+1})^-]^2 \right)},
\end{aligned}$$

where we have used the Cauchy Schwartz inequality for vectors

$$\sum_i a_i b_i \leq \sqrt{\sum_i a_i^2} \sqrt{\sum_i b_i^2}.$$

Recall that we have defined $[v_h'(x_j)]^2 := \frac{[v_h'(x_j)^-]^2 + [v_h'(x_j)^+]^2}{2}$, thus

$$[v_h'(x_j)^-]^2 + [v_h'(x_j)^+]^2 = 2[v_h'(x_j)]^2$$

and

$$\sqrt{\sum_{j=0}^{N} \left([v_h'(x_j)^+]^2 + [v_h'(x_{j+1})^-]^2 \right)} \leq \sqrt{\sum_{j=0}^{N} 2[v_h'(x_j)]^2} = \sqrt{2} \|\mathbf{v}'\|.$$

With (3.16), we get the continuity

$$\mathcal{A}_h(w_h, v_h) \leq \max_x a(x) \frac{h}{2} \sqrt{2}\|\mathbf{w}'\| \sqrt{2}\|\mathbf{v}'\| \leq 2 \max_x a(x) ||w_h'||_0 ||v_h'||_0$$

$$\leq 2 \max_x a(x) ||w_h||_1 ||v_h||_1.$$

3.6.4 Coercivity implies stability of the finite difference scheme

Recall that in Section 3.4.3 we have shown the nonsingularity of the *stiffness* matrix for the abstract finite element method without any quadrature.

Now we are ready to discuss how the V^h-ellipticity or the discrete coercivity can imply nonsingularity of the *stiffness* matrix S with entries $S_{ij} = \mathcal{A}_h(\phi_j, \phi_i)$ for the finite element method with quadrature. In particular, for P^1 finite element method with trapezoidal rule solving a variable coefficient problem, from (3.10) we know the stiffness matrix can be written as

$$S = \frac{1}{h}\frac{1}{2} \begin{pmatrix} a_0 + 2a_1 + a_2 & -a_1 - a_2 & & \\ -a_1 - a_2 & a_1 + 2a_2 + a_3 & -a_2 - a_3 & \\ & & & \\ & \ddots & \ddots & \ddots \end{pmatrix}$$

Since $v_h(x) = \sum_{j=1}^N v_j \phi_j(x)$, we have

$$\mathcal{A}_h(v_h, v_h) = \mathcal{A}_h(\sum_{j=1}^N v_j \phi_j(x), \sum_{i=1}^N v_i \phi_i(x)) = \sum_{i=1}^N \sum_{j=1}^N \mathcal{A}_h(\phi_j(x), \phi_i(x)) v_i v_j = \mathbf{v}^T S \mathbf{v}.$$

With the coercivity in Section 3.6.2 and (3.15), we have

$$\mathcal{A}_h(v_h, v_h) \geq C\|v_h\|_1^2 \geq C\|v_h\|_0^2 \geq Ch\|\mathbf{v}\|^2$$

So we have $\mathbf{v}^T S \mathbf{v} \geq Ch\|\mathbf{v}\|^2$ for any $\mathbf{v} \in \mathbb{R}^N$, which implies S is positive definite. The symmetry of S is implied by $\mathcal{A}_h(w_h, v_h) = \mathcal{A}_h(v_h, w_h)$. So S is invertible. By the Courant-Fisher-Weyl min-max principle (see Appendix A.1), $\frac{\mathbf{v}^T S \mathbf{v}}{\mathbf{v}^T \mathbf{v}} \geq Ch$ implies that the smallest eigenvalue of S is greater than or equal to $Ch > 0$. Therefore S is invertible.

The matrix-vector form of (3.10) is $S\mathbf{u} = h\mathbf{f}$, thus $\mathbf{u} = hS^{-1}\mathbf{f}$. Since we have shown S is real symmetric positive definite, thus singular values are also eigenvalues for both S and S^{-1}. So $\|S^{-1}\|$ is the simply the reciprocal of the smallest eigenvalue of S. Therefore we get $\|hS^{-1}\| = h\|S^{-1}\| \leq C$, which is precisely the stability in the sense of traditional finite difference method in Chapter 2.

Problem 3.7. *Recall that the traditional finite difference scheme* (2.8) *in Chapter 2 is given as*

$$\frac{1}{\Delta x^2}\begin{pmatrix} a_{\frac{1}{2}}+a_{\frac{3}{2}} & -a_{\frac{3}{2}} & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}}+a_{\frac{5}{2}} & -a_{\frac{5}{2}} & \\ & \ddots & \ddots & \ddots \end{pmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix}.$$

Apply the discussion in this section to prove the stability of this scheme. The consistency or the truncation error of the scheme (2.8) *is straightforward to derive. Once we have the stability, we have its convergence following Chapter 2. Hint: it becomes trivial if we can have an equivalent scheme in the following form*

$$\frac{1}{h}\frac{1}{2}\begin{pmatrix} b_0+2b_1+b_2 & -b_1-b_2 & & \\ -b_1-b_2 & b_1+2b_2+b_3 & -b_2-b_3 & \\ & \ddots & \ddots & \ddots \end{pmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = h\begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix}.$$

So how do we define b_i so that they are equivalent?

3.7 Error estimates of the finite element method with quadrature

In order to derive the error estimates of the finite element method with quadrature (3.8), we need to show all the lemmas and theorems in Section 3.5 also hold when $\mathcal{A}(\cdot,\cdot)$ is replaced by $\mathcal{A}_h(\cdot,\cdot)$. If this is your first time to read this chapter, you can assume that this is true and skip this section.

3.7.1 First Strang Lemma

The First Strang Lemma is the Céa's Lemma for the scheme (3.8).

Theorem 3.5. *[First Strang Lemma]*

$$\|u-u_h\|_1 \le C \inf_{v_h\in V_0^h}\left\{\|u-v_h\|_1 + \sup_{w_h\in V_0^h}\frac{|\mathcal{A}(v_h,w_h)-\mathcal{A}_h(v_h,w_h)|}{\|w_h\|_1}\right\}$$
$$+C\sup_{w_h\in V_0^h}\frac{|\langle f,w_h\rangle_h-(f,w_h)|}{\|w_h\|_1}.$$

Remark 3.9. *Compared to Céa's Lemma, the extra terms in the First Strang Lemma is nothing but quadrature error terms.*

Proof. First, we can rewrite the bilinear form

$$\mathcal{A}_h(u_h-v_h,u_h-v_h) = \mathcal{A}_h(u_h,u_h-v_h)-\mathcal{A}_h(v_h,u_h-v_h)+\mathcal{A}(u-v_h,u_h-v_h)-\mathcal{A}(u-v_h,u_h-v_h)$$

$= \mathcal{A}_h(u_h, u_h - v_h) - \mathcal{A}_h(v_h, u_h - v_h) + \mathcal{A}(u - v_h, u_h - v_h) + \mathcal{A}(v_h, u_h - v_h) + \mathcal{A}(u, u_h - v_h).$

By coercivity of $\mathcal{A}_h$, and the facts $\mathcal{A}_h(u_h, u_h - v_h) = \langle f, u_h - v_h \rangle_h$ and $A(u, u_h - v_h) = (f, u_h - v_h)$, we get

$$C\|u_h - v_h\|_1^2 \leq \mathcal{A}_h(u_h - v_h, u_h - v_h) = \mathcal{A}(u - v_h, u_h - v_h) + \mathcal{A}(v_h, u_h - v_h) - \mathcal{A}_h(v_h, u_h - v_h) + \langle f, u_h - v_h \rangle_h - (f, u_h - v_h), \quad \forall v_h \in V_0^h.$$

With $\mathcal{A}(u - v_h, u_h - v_h) \leq C_2 \|u - v_h\|_1 \|u_h - v_h\|_1$, we have

$$C\|u_h - v_h\|_1 \leq C_2\|u - v_h\|_1 + \frac{\mathcal{A}(v_h, u_h - v_h) - \mathcal{A}_h(v_h, u_h - v_h)}{\|u_h - v_h\|_1} + \frac{\langle f, u_h - v_h \rangle_h - (f, u_h - v_h)}{\|u_h - v_h\|_1}$$

thus

$$\|u_h - v_h\|_1 \leq C\|u - v_h\|_1 + C \sup_{w_h \in V_0^h} \frac{|\mathcal{A}(v_h, w_h) - \mathcal{A}_h(v_h, w_h)|}{\|w_h\|_1} + C \sup_{w_h \in V_0^h} \frac{|\langle f, w_h \rangle_h - (f, w_h)|}{\|w_h\|_1}.$$

The proof is done after using the triangle inequality:

$$\|u - u_h\|_1 \leq \|u - v_h\|_1 + \|u_h - v_h\|_1$$

□

3.7.2 Quadrature estimate: Bramble Hilbert Lemma

The first Strang Lemma means that the Céa's Lemma holds up to the quadrature error, which can be estimated by the Bramble Hilbert Lemma:

Theorem 3.6 (Bramble Hilbert Lemma)**.** *For some integer $k \geq 0$, let $\mathcal{L}$ be a continuous linear form on the space $H^{k+1}(0,1)$ with the property that $\forall p(x) \in P^k(\Omega)$ (all polynomials of degree k), $\mathcal{L}[p(x)] = 0$. Then*

$$|\mathcal{L}(f)| \leq C\|\mathcal{L}\|_{k+1}^* |f|_{k+1},$$

where $\|\cdot\|_{k+1}^*$ is the operator norm and $|f|_{k+1} = \sqrt{\int_\Omega |f^{(k+1)}(x)|^2 dx}$ is the H^{k+1}-seminorm.

Remark 3.10. *The notation in the Bramble Hilbert Lemma are abstract but one typical example of such a linear operator is the interpolation error operator. For instance, given point values f_i of some function $f(x)$ on a uniform mesh, we can do a piecewise linear polynomial interpolation as in Section 3.2.2. The interpolation error is a linear operator w.r.t. $f(x)$, and the interpolation error is always zero if $f(x)$ is a linear polynomial. Then the Bramble Hilbert Lemma implies that this piecewise linear interpolation error is controlled by $|f|_2$, which contains the second order derivative (in the weak sense). On the other hand, we can also get similar conclusion that the piecewise linear interpolation error is dominated by or related to $f''(x)$ through Taylor expansion. So if you prefer, you can think of the Bramble Hilbert Lemma as the better alternative as opposed to performing Taylor expansion.*

Remark 3.11. *The power of the abstraction in the Bramble Hilbert Lemma lies in the fact that we easily extend the interpolation and quadrature error estimates in Section 3.2.2 to unstructured meshes on any shape of domain. Recall that the H^1-norm error estimate is built upon the the interpolation error estimate. This is why the arguments for deriving error estimates in this chapter also apply to any general setup such as problems in multiple dimensions.*

Consider the quadrature error operator, which is linear and also zero for polynomials of certain degree. For instance, if considering the trapezoidal rule for each interval in Figure 3.1, then

$$\int_0^1 f(x)dx - \sum_{i=0}^{N} \frac{1}{2}h[f(x_i)+f(x_{i+1})] = \sum_{i=0}^{N} \left(\int_{x_i}^{x_{i+1}} f(x)dx - \frac{1}{2}h[f(x_i)+f(x_{i+1})] \right).$$

Consider a mapping from the small cell $[x_i, x_{i+1}]$ to the reference cell $[0,1]$ by

$$x = h\hat{x} + x_i, \quad \hat{f}(\hat{x}) = f(h\hat{x} + x_i).$$

Let

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

be the quadrature error on a small inverval, and

$$\hat{E}(\hat{f}) = \int_0^1 \hat{f}(\hat{x})d\hat{x} - \frac{1}{2}[\hat{f}(0)+\hat{f}(1)]$$

be the quadrature error on a reference interval $[0,1]$. Then $\hat{E}$ is the linear operator $\mathcal{L}$ in the Bramble Hilbert Lemma on $\Omega = [0,1]$ and we have

$$|E_i(f)| = h|\hat{E}(\hat{f})| \leq hC|\hat{f}|_2 = hC\sqrt{\int_0^1 \hat{[}f''(\hat{x})]^2 d\hat{x}} = h^{2.5}C\sqrt{\int_{x_i}^{x_{i+1}} [f''(x)]^2 dx}.$$

With Cauchy Schwartz inequality for vectors $\sum_i a_i b_i \leq \sqrt{\sum_i a_i^2}\sqrt{\sum_i b_i^2}$, we get the total quadrature error as

$$\sum_{i=0}^{N} |E_i(f)| \leq Ch^2 \sum_{i=0}^{N} \sqrt{h}\sqrt{\int_{x_i}^{x_{i+1}} [f''(x)]^2 dx} \leq Ch^2 \sqrt{\sum_{i=0}^{N} h}\sqrt{\sum_{i=0}^{N} \int_{x_i}^{x_{i+1}} [f''(x)]^2 dx}$$

$$= Ch^2|f|_2.$$

So we have just proven that

$$\left| \int_0^1 f(x)dx - \sum_{i=0}^{N} \frac{1}{2}h[f(x_i)+f(x_{i+1})] \right| \leq Ch^2|f|_2. \tag{3.17}$$

3.7.3 Error estimates

We only demonstrate the main idea why the error estimates for the abstract finite element method can still hold after quadrature error is involved. We focus on the simplest example. Consider the scheme (3.8) for $a(x) \equiv 1$, i.e., the scheme $\frac{1}{h^2}K\mathbf{u} = \mathbf{f}$. The integrand in the bilinear $\mathcal{A}(u_h, v_h)$ is simply piecewise constant because u_h and v_h are piecewise linear. Thus we have $\mathcal{A}(u_h, v_h) = \mathcal{A}_h(u_h, v_h)$ and the first Strang Lemma reduces to

$$\|u - u_h\|_1 \leq C \inf_{v_h \in V_0^h} \|u - v_h\|_1 + C \sup_{w_h \in V_0^h} \frac{|\langle f, w_h\rangle_h - (f, w_h)|}{\|w_h\|_1}.$$

For a piecewise polynomial w_h, its second order derivative only exists on each interval, thus we abuse the notation by letting $|w_h|_2$ denote (this is usually called Broken Sobolev space, i.e., the Sobolev space on each small interval)

$$|w_h|_2^2 = \sum_i \int_{x_i}^{x_{i+1}} [w_h''(x)]^2 dx.$$

With this modification of seminorm (you can verify that (3.17) still holds if replacing f by w_h), by (3.17), we have

$$|\langle f, w_h\rangle_h - (f, w_h)| \leq Ch^2 |fw_h|_2.$$

Notice that in each interval $(fw_h)'' = (f'w_h + fw_h')' = f''w_h + 2f'w_h'$ because $w''(x) \equiv 0$ within each interval. Thus with Cauchy Schwartz inequality, we have

$$\begin{aligned} |\langle f, w_h\rangle_h - (f, w_h)| &\leq Ch^2 |fw_h|_2 = Ch^2 |f''w_h + 2f'w_h'|_0 \\ &\leq Ch^2 (|f''|_0 |w_h|_0 + 2|f'||w_h'|) \leq Ch^2 \|f\|_2 \|w\|_1. \end{aligned}$$

Therefore we obtain the H^1 estimate as

$$\|u - u_h\|_1 \leq Ch|u|_2 + Ch^2 \|f\|_2.$$

Similarly, the Aubin-Nitsche Lemma also holds up to quadrature error.

The conclusion is very simple: the orders in the estimates in (3.12) and (3.14) still hold in the estimates for the scheme with quadrature (3.8).

3.8 Generalization: general domain in two dimensions

We will have a quick glance at how everything can be easily extended to a general setup. Consider solving a two-dimensional Poisson equation:

$$-\nabla \cdot (A(x, y)\nabla u(x, y)) = f(x, y), \quad (x, y) \in \Omega$$

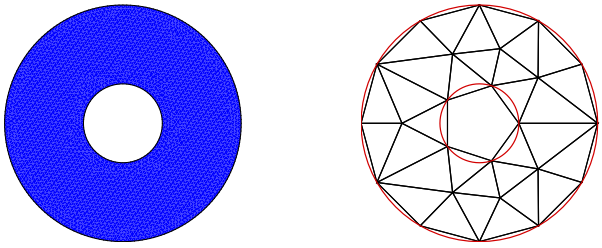


Figure 3.3: Left: the domain Ω. Right: the approximated domain Ω_h via a triangular mesh.

with homogeneous Dirichlet boundary conditions $u(x, y) = 0$ along the domain boundary Ω for a bounded region Ω, where $A(x, y)$ is a 2×2 matrix coefficient.

We just mention some key ingredients in the generalization to see how an easy extension is possible in the first place:

1. Multiplying the test function and integration by parts, we get the equivalent variational formulation for the PDE:

$$\text{seek } u \in H_0^1(\Omega), \quad \iint_\Omega \nabla v^T A \nabla u dxdy = \iint_\Omega f v dxdy, \quad \forall v \in H_0^1(\Omega),$$

   which can be denoted as $\mathcal{A}(u, v) = (f, v)$.

2. Construct an unstructured triangular mesh, which gives an approximated domain Ω_h as shown in Figure 3.3. Notice that the approximated boundary $\partial\Omega_h$ is a piecewise segment approximation to the curved boundary $\partial\Omega$, which induces a second order geometric error thus any finite element method defined on this Ω_h can be at most second order accurate even if using very high order polynomial basis. On the other hand, we can easily fix this issue by using curved triangle along the boundary, but quadrature on curved triangles are more expensive. For simplicity, we just consider the mesh shown in Figure 3.3.

3. We define V_0^h as the continuous piecewise linear polynomial space on the mesh shown in Figure 3.3, with the property of vanishing on $\partial\Omega_h$. An abstract finite element method is natually given as

$$\text{seek } u_h \in V_0^h, \quad \iint_\Omega \nabla v_h^T A \nabla u_h dxdy = \iint_\Omega f v_h dxdy, \quad \forall v_h \in V_0^h,$$

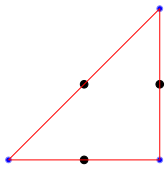
   which can be denoted as $\mathcal{A}(u_h, v_h) = (f, v_h)$.

4. Now let us consider what kind of coefficient $A(x, y)$ can ensure coercivity. For instance, if we assume that A is real symmetric and its smallest eigenvalue has a uniform positive lower bound, i.e., $\lambda(A) \geq C > 0$ for any (x, y), then by the Courant-Fisher-Weyl Min-Max principle,

$$\frac{\nabla v^T A \nabla v}{\nabla v^T \nabla v} \geq C \Rightarrow \mathcal{A}(v, v) \geq C|v|_1^2 \geq C\|v\|_1^2, \quad \forall v \in H_0^1(\Omega).$$

where we have used the fact that H^1-seminorm and H^1-norm are equivalent in $H_0^1(\Omega)$.

5. Assume V_0^h is N-dimensional. We can define *Lagrangian* basis (also called *nodal basis*) functions $\phi_i(x, y)$ on Ω_h just like the one-dimensional case. For instance, a linear polynomial is completely determined by its point values at three vertices on the triangle, and a quadratic polynomial is completely determined by its point values at three vertices and three edge centers on the triangle.



6. Plugging in $u_h(x, y) = \sum_{j=1}^N u_j \phi_j(x)$ we get a linear system

$$\sum_{j=1}^N u_j \mathcal{A}(\phi_j, \phi_i) = (f, \phi_i), \quad i = 1, \cdots, N,$$

and the *stiffness* matrix S has entries $S_{ij} = \mathcal{A}(\phi_j, \phi_i)$.

7. It can be shown that weak partial derivatives of any $v_h \in V_0^h$ exist thus it is conforming: $V_0^h \subset H_0^1(\Omega)$. So the proof of Galerkin Orthogonality holds. Coercivity and Galerkin Orthogonality imply Céa's Lemma. Once we have Céa's Lemma, the H^1-norm error is controlled by the interpolation error, which can be given via Bramble-Hilbert Lemma. Similary, the Aubin-Nitsche Lemma also holds.

8. The quadrature using only three vertices is exact for linear polynomials on a triangle. The quadrature using three vertices and three edge centers is exact for quadratic polynomials on a triangle. With a suitable quadrature, the finite element method can be represented as

$$\text{seek } u_h \in V_0^h, \quad \mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \quad \forall v_h \in V_0^h.$$

9. Finally, if you are curious whether this is still a finite difference scheme if using a structured triangular mesh, e.g., one rectangle is splitted into

two triangles in a rectangular mesh, then the answer is yes! With no surprises, the P^1 finite element method will give the same 5-point discrete Laplacian scheme as in Chapter 2. The P^2 polynomial finite element method for $-\Delta u = f$ gives a fourth order accurate (superconvergence) finite difference scheme with the following stencil:

$$\begin{array}{ccccccccccc} & & & & & & 1 & & & & \\ & & -1 & & & & -4 & & & & \\ \text{edge center} & -1 & 4 & -1 & \text{vertex} & 1 & 12 & -4 & 1 & . \\ & & -1 & & & & -4 & & & & \\ & & & & & & 1 & & & & \end{array}$$

3.9 Generalization: purely Neumann b.c.

Consider a one-dimensional problem

$$-(a(x)u'(x))' = f(x), \quad x \in (0,1)$$

$$u'(0) = \sigma_0, u'(1) = \sigma_1.$$

Recall that $f(x)$ must be compatible with the boundary conditions:

$$\int_0^1 f(x)dx = -a_1\sigma_1 + a_0\sigma_0, \tag{3.18}$$

which is obtained by integrating the PDE.

3.9.1 Quotient space $H^1(\Omega)/P^0(\Omega)$

Recall that this boundary value problem does not have a unique solution: if $u(x)$ is a solution, then so is $u(x) + c$ for any constant c. This non-uniqueness issue must be addressed properly. To this end, it is natural to consider a quotient space in which two functions differing by only a constant are regarded as the same function.

Let $P^0(\Omega)$ be the linear space of all polynomials of degree zero, i.e., all constants. We first introduce an *equivalent* class by

$$\dot{v}(x) := \{w(x) = v(x) + c, \quad c \in P^0(\Omega)\}.$$

In other words, if two functions $v(x)$ and $w(x)$ are different only by a constant c, then we regard them to be in the same equivalent class, which is a set. Any element $w(x)$ in an equivalent class $\dot{v}(x)$ is called a representation of the equivalent class $\dot{v}(x)$. For instance, in this section, $v(x)$ means a representation of the equivalent class $\dot{v}(x)$ which $v(x)$ belongs to.

The quotient space $H^1(\Omega)/P^0(\Omega)$ is defined as

$$H^1(\Omega)/P^0(\Omega) = \{\dot{v}(x) : v(x) \in H^1(\Omega)\}.$$

The norm the quotient space $H^1(\Omega)/P^0(\Omega)$ is defined as

$$\|\dot{v}\|_1 := \inf_{w\in\dot{v}} \|w\|_1,$$

where $\|w\|_1$ is the H^1-norm of the representation $w(x)$. This definition can be explicitly written as

$$\|\dot{v}\|_1 := \inf_{c\in P^0(\Omega)} \|v(x)+c\|_1 = \min_{c\in\mathbb{R}} \sqrt{\int_\Omega |v(x)+c|^2 dx + \int_\Omega \left|\frac{d}{dx}(v(x)+c)\right|^2 dx}.$$

So we get

$$\|\dot{v}\|_1^2 := \min_{c\in\mathbb{R}} \int_\Omega |v(x)+c|^2 dx + \int_\Omega |v'(x)|^2 dx,$$

which is nothing but a minimization with respect to c. Also, it is a simple quadratic function of the number c, so the minimizer is the average of $v(x)$, $c = \frac{1}{|\Omega|}\int_\Omega v(x)dx$. For the domain $\Omega = (0,1)$, let $\bar{v} = \int_0^1 v(x)dx$ be the average of the function $v(x)$ over Ω. Then the quotient space $H^1(\Omega)/P^0(\Omega)$ can be equivalently written as

$$\|\dot{v}\|_1^2 = \int_\Omega |v(x)-\bar{v}|^2 dx + \int_\Omega |v'(x)|^2 dx.$$

This quotient space norm is also equivalent to the seminorm $|v|_1$:

$$C\|\dot{v}\|_1 \le |v|_1 \le \|\dot{v}\|_1, \quad C>0.$$

The first inequalilty is true because of the following *Poincaré inequalilty* (see Appendix for a generic statement):

$$\int_\Omega |v(x)-\bar{v}|^2 dx \le C\int_\Omega |v'(x)|^2 dx.$$

3.9.2 Variational formulation and coercivity

Multiplying a test function and intergration by parts, we can get a variational form:

$$\int_0^1 a(x)u'(x)v'(x)dx = \int_0^1 f(x)v(x)dx + a_1\sigma_1 v(1) - a_0\sigma_0 v(0).$$

Obviously, both side stay the same if we replace $u(x)$ by $u(x)+c$ for any constant c. Now if we replace $v(x)$ by $v(x)+c$, the left hand side stays the same, and the right hand side also stays the same because of the compatibility condition (3.18).

So the equivalent variational formulation is to seek $\dot{u} \in H^1(\Omega)/P^0(\Omega)$ such that

$$\int_\Omega a(x)u'(x)v'(x)dx = \int_\Omega f(x)v(x)dx + a_1\sigma_1 v(1) - a_0\sigma_0 v(0), \quad \forall \dot{v} \in H^1(\Omega)/P^0(\Omega).$$

It can be denoted by the same short hand notation as:

$$\mathcal{A}(u,v) = (f,v) + a_1\sigma_1 v(1) - a_0\sigma_0 v(0), \quad \forall \dot{v} \in H^1(\Omega)/P^0(\Omega).$$

The Cauchy-Schwartz inequalilty implies the continuity of the bilinear form. Since quotient space norm is also equivalent to the H^1 seminorm, we also have the coercivity:

$$\mathcal{A}(v,v) \geq C\|\dot{v}\|_1, \quad \forall v \in H^1(\Omega)/P^0(\Omega).$$

3.9.3 The finite element method

On a mesh with intervals I_j, we define the space V^h as an approximation to $H^1(0,1)$:

$$V^h = \{v_h(x) \in C(0,1) : v_h(x) \in P^k(I_j), \forall j\}.$$

We can also define a quotient space V^h/P^0 similarly:

$$V^h/P^0 = \{\dot{v}_h(x) : v_h(x) \in V^h\}.$$

The finite element method is to seek $\dot{u}_h(x) \in V^h/P^0$ such that

$$\mathcal{A}(u_h,v_h) = (f,v_h) + a_1\sigma_1 v_h(1) - a_0\sigma_0 v_h(0), \quad \forall \dot{v}_h \in V^h/P^0.$$

Notice that we use representations u_h and v_h in the bilinear form $\mathcal{A}(u_h,v_h)$, instead of their equivalent classes $\dot{u}_h$ and $\dot{v}_h$. All the previous arguments for error estimates can be established similarly, and the only difference is that the underlying function space is the quotient space $H^1(\Omega)/P^0(\Omega)$, even though we just plug in functions into the variational form as before.

3.9.4 Coercivity implies the stiffness matrix null space

For simplicity, we assume homogeneous Neumann boundary condition $\sigma_0 = \sigma_1 = 0$, and constant coefficient $a(x) = 1$. Then for the P^1 basis finite element method, the bilinear form with trapezoidal quadrature $\mathcal{A}_h(u_h,v_h)$ is the same as $\mathcal{A}(u_h,v_h)$.

Recall our uniform grid points are

$$0 = x_0 < x_1 < \cdots < x_N < x_{N+1} = 1.$$

Let $\phi_i(x), i = 0,1,\cdots,N+1$ be the Lagrangian basis or *nodal basis* of V^h. Then the stiffness matrix $S \in \mathbb{R}^{(N+2)\times(N+2)}$ has entries $S_{ij} = \mathcal{A}_h(\phi_j,\phi_i) = \mathcal{A}(\phi_j,\phi_i)$. Here we have abused notation by allowing indices i,j to take value 0.

With similar notation as before, e.g., $\mathbf{v} \in \mathbb{R}^{N+2}$ denoting a vector of point values $v_h(x_j)$, we have

$$\mathbf{v}^T S\mathbf{v} = \mathcal{A}(v_h,v_h) \geq C\|\dot{v}_h\|_1 \geq 0,$$

thus S is still real symmetric and positive semi-definite.

Since the boundary value problem does not have a unique solution, the stiffness matrix S must have a nontrival null space. As a matter of fact, the constant one vector $\mathbf{1}$ is in its null space. We first have

$$\forall \mathbf{v}, \quad \mathbf{v}^T S\mathbf{1} = \mathcal{A}(1, v_h) = 0 \Rightarrow \mathbf{v} \perp S\mathbf{1}, \quad \forall \mathbf{v} \Rightarrow S\mathbf{1} = \mathbf{0}.$$

Next, we want to show that the coercivity implies that the null space of S is one-dimensional:

$$S\mathbf{v} = \mathbf{0} \Leftrightarrow \mathbf{v}^T S\mathbf{v} = 0 \Leftrightarrow \mathcal{A}(v_h, v_h) = 0 \Rightarrow \|\dot{v}_h\|_1 = 0,$$

where the last step is due to the coercivity. Thus

$$S\mathbf{v} = \mathbf{0} \Rightarrow \|\dot{v}_h\|_1 = 0 \Leftrightarrow \dot{v}_h(x) = \dot{0} \Leftrightarrow v_h(x) \equiv c \Leftrightarrow \mathbf{v} = c\mathbf{1},$$

because a function in the quotient space has zero norm if and only if it is $\dot{0}$, which is the property of a norm.

3.9.5 The finite difference form

For simplicity, just consider the constant coefficient case $a(x) = 1$, for piecewise linear basis with trapezoidal quadrature on the uniform grid, the finite element method can be equivalently written as

$$\begin{aligned}
&\frac{1}{h}(u_1 - u_0) = \frac{h}{2} f_0 + a_0\sigma_0 \\
&\frac{1}{h}(-u_{j-1} + 2u_j - u_{j+1}) = h f_j, \quad j = 1, \cdots, N \\
&\frac{1}{h}(u_{N+1} - u_N) = \frac{h}{2} f_{N+1} + a_1\sigma_1
\end{aligned}$$

which is exactly the same as the traditional finite difference scheme in Section 2.6.3.

Now the finite element theory can give error estimates like (3.12) and (3.14). On the other hand, it is straightforward to check the truncation error at x_0 or x_{N+1} is only first order, even though the Neumann boundary condition was approximated by a second order centered difference in Section 2.6.3. It is quite difficult to show that this scheme is second order accurate following arguments in Chapter 2, epsecially for a variable coefficient problem in multiple dimensions. But we know this scheme is indeed second order accurate in the sense of (3.14), which demonstrates the superiority of the finite element method compared to traditional finite difference method.

For $a(x) = 1$, the stiffness matrix S with entries $\mathcal{A}_h(\phi_j, \phi_i)$ can be written out as

$$S = \frac{1}{h}\begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{pmatrix}_{(N+2)\times(N+2)}.$$

In general, we have

$$S = \frac{1}{h}\frac{1}{2}\begin{pmatrix} a_0+a_1 & -a_0-a_1 & & & & \\ -a_0-a_1 & a_0+2a_1+a_2 & -a_1-a_2 & & & \\ & -a_1-a_2 & a_1+2a_2+a_3 & -a_2-a_3 & & \\ & & & & & \\ & & & \ddots & \ddots & \ddots \end{pmatrix}_{(N+2)\times(N+2)}.$$

3.9.6 How to solve the singular linear system

Now consider the finite element method with quadrature for a variable coefficient problem:

$$\mathcal{A}_h(u_h, v_h) = \langle f, v_h\rangle_h + a_1\sigma_1 v_h(1) - a_0\sigma_0 v_h(0), \quad \forall \dot{v}_h \in V^h/P^0.$$

For piecewise linear finite element method, if using a Lagrangian basis or *nodal basis* $\phi_j(x)$, by plugging in $v_h = \phi_i(x)$, we get a singular linear system

$$S\begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \\ u_{N+1} \end{pmatrix} = h\begin{pmatrix} \frac{1}{2}f_0 \\ f_1 \\ \vdots \\ f_N \\ \frac{1}{2}f_{N+1} \end{pmatrix} + \begin{pmatrix} a_0\sigma_0 \\ 0 \\ \vdots \\ 0 \\ a_1\sigma_1 \end{pmatrix}.$$

Since the null space of S is $Span\{\mathbf{1}\}$, the least square solution of this singular linear system corresponds to the facts that the finite element method is defined on a quotient space V^h/P^0, and we look for a solution in a quotient space $H^(\Omega)/P^0(\Omega)$ in the original variational formulation. We do not elaborate more about it here, but at least all discussions about quotient spaces and the null spaces are consistent.

Let $S\mathbf{u} = \tilde{f}$ denote this singular linear system. The first question we should ask is whether it has a solution since S is not invertible. A linear system $S\mathbf{u} = \tilde{f}$ if and only if $\tilde{f}$ is in the column space of S, denoted by $Col(S)$. To this end, we need the orthogonal complement of $Col(S)$, denoted by $Col(S)^\perp$:

$$Col(S)^\perp = \{\mathbf{y} \in \mathbb{R}^{N+2} : \mathbf{y}^T S = \mathbf{0}.\}$$

Since S is symmetric, $\mathbf{y}^T S = \mathbf{0} \Leftrightarrow S\mathbf{y} = \mathbf{0}$, thus $Col(S)^\perp$ is the null space of S. In particular, we know that $\mathbf{1}$ is the basis of $Col(S)^\perp$. We have

$$\tilde{f} \in Col(S) \Leftrightarrow \tilde{f} \perp Col(S)^\perp \Leftrightarrow \tilde{f} \perp \mathbf{1}$$

$$\Leftrightarrow \frac{1}{2}hf_0 + h\sum_{j=1}^{N} f_j + \frac{1}{2}hf_{N+1} + a_0\sigma_0 + a_1\sigma_1 = 0$$

which is nothing but a discrete compatibility condition.

For a function $f(x)$ satisfying the compatibility condition, its point values may not necessarily satisfy the discrete compatibility condition. We can simply project $\tilde{f}$ to the column space of S. Let $\bar{f}$ be the projection vector, then $S\mathbf{u} = \bar{f}$ is ensured to have a solution, and we can use iterative solvers in Chapter 8 such as conjugate gradient method or its preconditioned version directly on $S\mathbf{u} = \bar{f}$ to find the least square solution to $S\mathbf{u} = \tilde{f}$. Since we know what $Col(S)^\perp$ is, the projection $\bar{f}$ is quite easy to find. We summerize it as follows:

1. The projection $\bar{f}$ is computed as

$$\bar{f} = \tilde{f} - \frac{\langle \mathbf{1}, \tilde{f} \rangle}{\|\mathbf{1}\|^2}\mathbf{1}.$$

   It is easy to verify $\langle \mathbf{1}, \bar{f} \rangle = 0$.

2. Solve $S\mathbf{u} = \bar{f}$ by direct or iterative solvers. See Chapter 8.

Remark 3.12. *Iterative solvers like conjugate gradient may not work well directly on $S\mathbf{u} = \tilde{f}$ epsecially if the discrete compatibility error is large.*

Remark 3.13. *To find the least square solution to $S\mathbf{u} = \tilde{f}$, it is mathematically equivalent to solve the normal equation $S^T S\mathbf{u} = S^T\tilde{f}$ which is ensured to have a solution for any $\tilde{f}$. However, $S^T S\mathbf{u} = S^T\tilde{f}$ is much harder to solve. For example, if S is invertible, then the condition number of $S^T S$ is about the square of the condition number of S.*

Remark 3.14. *If S is not symmetric, in order to find the projection $\bar{f}$, we need to compute the left null vector $\mathbf{y}$ first: solving $S^T\mathbf{y} = \mathbf{0} = 0 * \mathbf{y}$ is an eigenvector problem, which is much more expensive than solving a linear system of the same size. For instance, for a nonsingular system $A\mathbf{x} = b$, iterative solvers are based on minimizing a function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - b^T\mathbf{x}$. For $A\mathbf{x} = 0$, if minimizing $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x}$, we simply get $\mathbf{x} = \mathbf{0}$, which is a solution that we do not want at all. For getting the nonzero solution to $A\mathbf{x} = 0$, roughtly speaking, we would have to minimize $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x}$ over the sphere $\{\mathbf{x} : \|\mathbf{x}\| = 1\}$.*

Remark 3.15. *For all remarks above, it is highly desired to have a symmetric S. The symmetry of the matrix S with entries $S_{ij} = \mathcal{A}_h(\phi_j, \phi_i) = \mathcal{A}_h(\phi_i, \phi_j)$ holds trivially even for a two-dimensional or three-dimensional problem $-\nabla \cdot (A\nabla u) = f$ with a real symmetric matrix coefficient A. This is one of the key advantages of using finite element method for purely Neumann boundary conditions. It is in general quite difficult to construct a real symmetrix matrix for variable coefficient problems with Neumann boundary in multiple dimensions by traditional finite difference method.*

3.10 Generalization: nonhomogeneou Dirichlet b.c.

Consider solving

$$-u''(x) = f(x), \quad x \in (0,1),$$
$$u(0) = \sigma_0, u(1) = \sigma_1.$$

The standard approach is to assume that there exists a smooth enough function $g(x)$ satisfying the same boundary condition. Then the function $\tilde{u} = u - g$ satisfies

$$-\tilde{u}''(x) = f(x) + g''(x), \quad x \in (0,1),$$
$$\tilde{u}(0) = \tilde{u}(1) = 0.$$

Obviously everything in Section 3.3 can be easily applied to construct and anaylze a finite element method for $\tilde{u} \in H_0^1(\Omega)$, provided that we know what $g(x)$ is, which is easy to construct in one-dimension but not necessarily in multiple dimensions.

However, we only need to know existence of the smooth function $g(x)$ and an actual implementation can be made irrevelant to what exactly $g(x)$ should be. The same order from the L^2-norm error estimate (3.14) can still hold.

By multiplying a test function $v \in H_0^1(\Omega)$ and intergration by parts, we get the equivalent variational form for seeking $\tilde{u} \in H_0^1(\Omega)$ satisfying

$$\int_0^1 \tilde{u}'(x)v'(x)dx = \int_0^1 f(x)v(x)dx - \int_0^1 g'(x)v'(x)dx, \quad \forall v \in H_0^1(\Omega),$$

which can be denoted as

$$\mathcal{A}(\tilde{u}, v) = (f, v) - \mathcal{A}(g, v), \quad \forall v \in H_0^1(\Omega).$$

3.10.1 A scheme in theory

An abstract finite element method that we should never implement is to find $\tilde{u}_h \in V_0^h$ satisfying

$$\mathcal{A}(\tilde{u}_h, v_h) = (f, v_h) - \mathcal{A}(g, v_h), \quad \forall v_h \in V_0^h.$$

Assume $g(x)$ is a nice function so that we can still derive the error estimates (3.12) and (3.14). For example, if $g''(x)$ exists, then after integration by parts for test function $v_h(x) \in V_0^h$, the abstract finite element is equivalent to seeking $\tilde{u}_h \in V_0^h$ satisfying

$$\mathcal{A}(\tilde{u}_h, v_h) = (f + g'', v_h), \quad \forall v_h \in V_0^h.$$

If we treat $f - g''$ as the right hand side function, then the error estimates (3.12) and (3.14) can still hold for $\tilde{u}_h - \tilde{u}$.

The numerical solution that we want is

$$u_h := \tilde{u}_h + g(x).$$

Be careful that we no longer have $u_h \in V^h$. By moving $\mathcal{A}(g, v_h)$ to the left hand side, we get

$$\mathcal{A}(u_h, v_h) = (f, v_h), \quad \forall v_h \in V_0^h.$$

Also $u_h - u$ satisfies the error estimates (3.12) and (3.14).

Next, assume we use quadrature, so we have

$$\mathcal{A}_h(\tilde{u}_h, v_h) = \langle f, v_h \rangle_h - \mathcal{A}_h(g, v_h), \quad \forall v_h \in V_0^h, \tag{3.19}$$

or equivalently

$$\mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \quad \forall v_h \in V_0^h. \tag{3.20}$$

Assume the estimates (3.12) and (3.14) still hold after using quadrature for the scheme (3.19).

3.10.2 A scheme for implementation

We consider the piecewise linear Lagrangian interpolation polynomial for $g(x)$ at grid points x_i, denoted by $g_h(x) = \Pi_1 g(x) \in V^h$. For *nodal basis* $\{\phi_j(x)\}_{j=0}^{N+1}$ of V^h, we simply have

$$g_h(x) = \sum_{j=0}^{N+1} g_j \phi_j(x) \in V^h,$$

where $g_0 = \sigma_0, g_{N+1} = \sigma_1, x_0 = 0, x_{N+1} = 1$. Then we consider a new scheme seeking $\tilde{u}_h \in V_0^h$ satisfying

$$\mathcal{A}_h(\tilde{u}_h, v_h) = \langle f, v_h \rangle_h - \mathcal{A}_h(g_h, v_h), \quad \forall v_h \in V_0^h. \tag{3.21}$$

The difference between the scheme (3.21) and the scheme (3.19) is where using $g(x)$ or its polynomial interpolation $g_h(x)$.

Let $u_h(x) = \tilde{u}_h(x) + g_h(x) \in V^h$, then we can rewrite (3.21) equivalently as

$$\mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \quad \forall v_h \in V_0^h. \tag{3.22}$$

This time, since $u_h(x) = \tilde{u}_h(x) + g_h(x) \in V^h$, we have

$$u_h(x) = \sum_{j=0}^{N+1} u_j \phi_j(x),$$

where $u_i = u_h(x_i)$.

Here I need to emphasize that V^h is $(N+2)$-dimensional with basis $\{\phi_j(x)\}_{j=0}^{N+1}$, whereras the test function space V_0^h is only N-dimensional with basis $\{\phi_j(x)\}_{j=1}^{N}$.

Obviously, plugging this representation into (3.22) and test function space basis $\phi_i(x)$ for $i = 1, \cdots, N$, we get a linear system

$$\sum_{j=0}^{N+1} \mathcal{A}_h(\phi_j, \phi_i) u_j = h f_j, \quad \forall i = 1, \cdots, N.$$

Of course the linear system should have only N unknowns because of Dirichlet boundary $u_0 = \sigma_0$ and $u_{N+1} = \sigma_1$. The scheme is precisely

$$\frac{1}{h}(-u_{j-1} + 2u_j - u_{j+1}) = h f_j, \quad j = 1, \cdots, N, \tag{3.23}$$

where $u_0 = \sigma_0, u_{N+1} = \sigma_1$.

Remark 3.16. *Notice that the scheme* (3.22) *is equivalent to the following scheme seeking* $u_h(x) \in V_0^h$ *satisfying*

$$A_h(u_h, v_h) = \langle f, v_h \rangle_h - A_h(\sigma_h, v_h), \quad \forall v_h \in V_0^h, \tag{3.24}$$

where $\sigma_h \in V^h$ *is the Lagrangian interpolation of the trivial nonsmooth extension function:*

$$\sigma_h(x_0) = \sigma_0, \sigma_h(x_{N+1}) = \sigma_1, \quad \sigma_h(x_i) = 0, i = 1, \cdots, N.$$

Remark 3.17. *The scheme* (3.22) *or* (3.24) *has nothing to do with what* $g(x)$ *is. On the other hand, with the existence of smooth* $g(x)$*, the error estimates can be easily established via the analysis of the scheme* (3.19)*. To establish error estimates for* (3.22) *or* (3.24)*, notice that their only difference from* (3.19) *is the following*

$$\mathcal{A}_h(g, v_h) - \mathcal{A}_h(g_h, v_h),$$

which can be analyzed through the interpolation error estimates on $\|g - g_h\|_1$ *and* $\|g - g_h\|_0$*. For instance, for convergence in* H^1 *norm, similar to the First Strang Lemma Theorem 3.5, we will have to deal with*

$$\sup_{w_h \in V_0^h} \frac{|\mathcal{A}_h(g, w_h) - \mathcal{A}_h(g_h, w_h)|}{\|w_h\|_1},$$

which can be easily done by discrete continuity of the bilinear form:

$$\frac{|\mathcal{A}_h(g, w_h) - \mathcal{A}_h(g_h, w_h)|}{\|w_h\|_1} \leq C \|g - g_h\|_1.$$

The scheme (3.23) is exactly the same as taking the scheme for purely Neumann boundary at interior grid points $j = 1, \cdots, N$ in Section 3.9. This is not a coincidence at all. This fact remains true even for high order polynomial basis with variable coefficients, which means that we have a neat treatment of boundary condition in finite element method. In particular, for a variable coefficient problem, by taking the scheme at interior grid points $j = 1, \cdots, N$ in Section 3.9, we obtain the P^1 finite element method with trapezoidal quadrature for the nonhomogeneous Dirichlet boundary as

$$\frac{-(a_{j-1} + a_j)u_{j-1} + (a_{j-1} + 2a_j + a_{j+1})u_j - (a_j + a_{j+1})u_{j+1}}{2h} = hf_j, \quad j = 1, \cdots, N,$$

where $u_0 = \sigma_0, u_{N+1} = \sigma_1$.

3.10.3 A scheme in theory for 2D general domain Ω

Consider solving a two-dimensional Poisson equation with nonhomogeneous Dirichlet boundary condition for a bounded region Ω:

$$-\nabla \cdot (A(x, y)\nabla u(x, y)) = f(x, y), \quad (x, y) \in \Omega,$$

$$u(x, y) = \sigma(x, y), \quad (x, y) \in \partial\Omega.$$

where $A(x, y)$ is a 2×2 matrix coefficient.

Assume there exists a smooth extension function $g(x, y)$ satisfying that $g|_{\partial\Omega}(x, y) = \sigma(x, y)$, then $\tilde{u} = u - g \in H_0^1(\Omega)$ satisfying

$$\mathcal{A}(\tilde{u}, v) = (f, v) - \mathcal{A}(g, v), \quad \forall v \in H_0^1(\Omega)$$

where the bilinear form is $\mathcal{A}(u, v) = \iint_\Omega \nabla v^T A \nabla u dx dy$.

Given a triangulation of the domain Ω_h as shown in (3.3), assume either Ω is polygonal or we use curved triangles, so that $\partial\Omega_h = \partial\Omega$. Define a continuous piecewise polynomial space $V_0^h \subset H_0^1(\Omega)$, then an abstract finite element method that can be easily analyzed is to find $\tilde{u}_h \in V_0^h$ satisfying

$$\mathcal{A}(\tilde{u}_h, v_h) = (f, v_h) - \mathcal{A}(g, v_h), \quad \forall v_h \in V_0^h.$$

The scheme with quadrature is written as

$$\mathcal{A}_h(\tilde{u}_h, v_h) = \langle f, v_h \rangle_h - \mathcal{A}_h(g, v_h), \quad \forall v_h \in V_0^h.$$

or equivalently

$$u_h = \tilde{u}_h + g, \quad \mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \quad \forall v_h \in V_0^h. \tag{3.25}$$

3.10.4 A scheme for implementation for 2D general domain Ω

The error estimates of (3.25) can be easily established. For the ease of implementation, we define $g_h(x)$ as the Lagrangian interpolation of $g(x)$ over nodal points in the mesh, which will be explained below.

Then we implement a different scheme

$$\tilde{u}_h \in V_0^h, \quad \mathcal{A}_h(\tilde{u}_h, v_h) = \langle f, v_h \rangle_h - \mathcal{A}_h(g_h, v_h), \quad \forall v_h \in V_0^h.$$

or equivalently

$$u_h = \tilde{u}_h + g_h \in V^h, \quad \mathcal{A}_h(u_h, v_h) = \langle f, v_h \rangle_h, \quad \forall v_h \in V_0^h. \tag{3.26}$$

For convenience, let $\mathbf{x}$ denote (x, y). Now we need to make some assumptions which are quite practical at least for P^1 and P^2:

I. V_0^h is N-dimensional and V^h is $(N+n)$-dimensional.

II. V^h has a Lagrangian basis (*nodal basis*) $\{\phi_j(\mathbf{x})\}_{j=1}^{N+n}$ satisfying

$$\phi_j(\mathbf{x}_i) = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases},$$

for the points $\mathbf{x}_i : i = 1, \cdots, N+n$.

III. V_0^h has a Lagrangian basis $\{\phi_j(\mathbf{x})\}_{j=1}^{N}$ satisfying

$$\phi_j(\mathbf{x}_i) = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases},$$

for the points $\mathbf{x}_i, i = 1, \cdots, N$. For example, $\mathbf{x}_i$ are three vertices of all triangles for a continuous piecewise linear polynomial on a triangular mesh. For a continuous piecewise quadratic polynomial on a triangular mesh, $\mathbf{x}_i$ are three vertices and three edge centers of all triangles.

IV. The quadrature points used in $\mathcal{A}_h(\cdot, \cdot)$ is a subset of $\{\mathbf{x}_i, i = 1, \cdots, N\}$. For instance, the quadrature using three vertices with equal weight is exact for integrating a linear polynomial on a triangle thus second order accurate by Bramble-Hilbert Lemma, and the quadrature using only three edge centers with equal weight is exact for integrating a quadratic polynomial on a triangle thus third order accurate by Bramble-Hilbert Lemma.

So the points $\{\mathbf{x}_i\}_{i=1}^{N}$ are interior points inside the domain Ω and the points $\{\mathbf{x}_i\}_{i=N+1}^{N+n}$ are boundary points, along the boundary $\partial\Omega_h = \Omega$ (not true in general but we assumed it).

Let $u_j = u_h(\mathbf{x}_j)$ and $\sigma_j = \sigma(\mathbf{x}_j)$, then

$$u_h(\mathbf{x}) = \sum_{j=1}^{N+n} u_j\phi_j(\mathbf{x}) = \sum_{j=1}^{N} u_j\phi_j(\mathbf{x}) + \sum_{j=N+1}^{N+n} \sigma_j\phi_j(\mathbf{x}).$$

Under these assumptions, the scheme (3.26) is exactly the same as

$$\mathcal{A}_h(\sum_{j=1}^{N+n} u_j\phi_j, v_h) = \langle f, v_h\rangle_h, \quad \forall v_h \in V_0^h. \tag{3.27}$$

or equivalently

$$\sum_{j=1}^{N} \quad \mathcal{A}_h(\phi_j, \phi_i)u_j = \langle f, \phi_i\rangle_h - \sum_{j=N+1}^{N+n} \mathcal{A}_h(\phi_j, \phi_i)\sigma_j, \quad i = 1, \cdots, N.$$

If you ever wonder what the simplest boundary treatment for a high order accurate scheme should be, (3.26) gives a perfect answer.

To establish the convergence in H^1-norm and L^2-norm for the scheme (3.26) or (3.27), we first can have the error estimates for (3.25), then analyze the only difference between (3.26) and (3.25):

$$\mathcal{A}_h(g, v_h) - \mathcal{A}_h(g_h, v_h),$$

which is related to the interpolation error estimates on $\|g - g_h\|_1$ and $\|g - g_h\|_0$.

3.10.5 The error in the 2-norm over grid point values

Obviously, the implementation in previous subsection has absolutely nothing to do with what $g(x)$ is. As a matter of fact, the implementation (3.23) is our classical finite difference scheme. But there is still one catch that I have not mentioned, for implementing the finite element method as a finite difference scheme.

To be specific, in (3.23), we can only get point values of $u_h(x)$ at x_j, even though in practice we are quite happy with that already. On the other hand, if we have $g(x)$ and we solve (3.19), then we get $u_h(x) = \tilde{u}_h(x) + g(x)$ for any $x \in (0, 1)$.

In terms of the error estimates, the L^2-norm (3.14) measures the error for all x in the interval $(0, 1)$. In the scheme (3.23), since we only have $u_h(x_j)$, the errors can be measured only at these grid points. For P^1 finite element method, it is straightforward to show that (3.14) for $u_h(x)$ implies the scheme (3.23) is second order accurate in the 2-norm:

$$\|\mathbf{e}\|_2 = \sqrt{h\sum_{j=1}^{N} e_j^2} = \sqrt{h\sum_{j=1}^{N} |u_j - u(x_j)|^2}.$$

The 2-norm above is an approximation to L^2-norm error $\|e_h\|_0$ by the trapezoidal quadrature for the error $e_h = u_h - u$:

$$\|e_h\|_0 = \sqrt{\int_0^1 |e_h(x)|^2 dx},$$

where $e_h(0) = e_h(1) = 0$ because $u_h(x)$ satisfies the boundary condition.

Remark 3.18. *For P^k basis finite element with $k \geq 2$, the error order for function values at (k+1)-point Gauss-Lobatto quadrature points are $(k+2)$-th order in the 2-norm. This one order higher phenomenon is called superconvergence of function values. We can use finite element method with quadratic polynomial to get a fourth order accurate finite difference scheme! Of course it can no longer be derived from L^2-norm estimate* (3.14)*, which is only third order accurate for P^2.*

3.11 Generalization: a general elliptic operator

Next, we consider an elliptic equation in the following form

$$-(a(x)u'(x))' + b(x)u'(x) + c(x)u(x) = f(x), \quad x \in (0,1), \qquad u(0) = u(1) = 0,$$

where $a(x) \geq \min_x a(x) > 0$ and $c(x) \geq 0$.

The variational form is still $\mathcal{A}(u,v) = (f,v)$ where

$$\mathcal{A}(u,v) = \int_0^1 a(x)u'(x)v'(x) + b(x)u'(x)v(x) + c(x)u(x)v(x)dx.$$

First of all, unless $b(x) = 0$, we lose the symmetry of the bilinear form, and $\mathcal{A}(u,v) = \mathcal{A}(v,u)$ is not true in general. Thus the stiffness matrix will no longer be symmetric. But other than this, almost everything above can be extended, under suitable assumptions.

For simplicity, we will just focus on how to establish the coercivity. Since $c(x) \geq 0$, we have

$$\mathcal{A}(v,v) \geq \int_0^1 a(x)|v'(x)|^2 dx + \int_0^1 b(x)v'(x)v(x)dx.$$

For the second order derivative term, recall that by *Poincaré inequality* we have

$$\int_0^1 a(x)|v'(x)|^2 dx \geq \min_x a(x)|v|_1^2 \geq C \min_x a(x)\|v\|_1^2, \quad \forall v \in H_0^1(\Omega),$$

where the constant C depends only on Ω.

For the first order derivative, after integration by parts, we get

$$\int_0^1 b(x)v'(x)v(x)dx = \int_0^1 b(x)\frac{d}{dx}\frac{v^2(x)}{2}dx = -\int_0^1 b'(x)\frac{v^2(x)}{2}dx, \quad \forall v \in H_0^1(\Omega). \tag{3.28}$$

In two dimensions, for a first order derivative term like $\mathbf{b} \cdot \nabla u$, after integration by parts, we have

$$\iint_\Omega (\mathbf{b} \cdot \nabla v) v d\mathbf{x} = -\iint_\Omega \frac{v^2}{2} (\nabla \cdot \mathbf{b}) d\mathbf{x}, \quad \forall v \in H_0^1(\Omega). \tag{3.29}$$

So we can get the *coercivity* $\mathcal{A}(v, v) \geq C\|v\|_1^2$ under the following assumptions:

1. If $b'(x) \equiv 0$, then the term in (3.28) is gone. In two dimensions, if $\nabla \cdot \mathbf{b} \equiv 0$, i.e., $\mathbf{b}$ is *incompressible*, then the term in (3.29) is gone in two dimensions.

2. If $b'(x) \leq 0$ in one dimension or $\nabla \cdot \mathbf{b} \leq 0$ in two dimensions, then we have
$$\mathcal{A}(v, v) \geq \int_0^1 a(x)|v'(x)|^2 dx \geq C\|v\|_1^2.$$

3. If $b'(x) \geq 0$, then we have to assume $\max_x b'(x) < 2C \min_x a(x)$ where C is the constant in the *Poincaré inequality*, thus
$$\mathcal{A}(v, v) \geq \min_x a(x) C\|v\|_1^2 - \max_x b'(x) \frac{1}{2}\|v\|_0^2 \geq (C \min_x a(x) - \frac{1}{2} \max_x b'(x))\|v\|_1^0.$$

Remark 3.19. *For the case $b'(x) \geq 0$, obviously we need the diffusion term $-(au')'$ to be strong enough to dominate the convection term bu'. However, if the diffusion coefficient is very small compared to $b'(x)$, then the coercivity will be lost, thus all arguments in finite element theory based on coercivity will also break down. In pratice, this reflects on the difficulties of using finite element theory to construct a scheme for convection dominated problems, e.g., $\max_x b'(x) >> \max_x a(x)$ or $a(x)$ is nearly zero.*

3.12 Generalization: higher order accuracy via P^2

We only discuss the constant coefficient case. If you are interested, you can find the variable coefficient case in [9].

3.12.1 Dirichlet b.c.

Let V^h and V_0^h denote the corresponding spaces of continuous piecewise quadratic polynomial was shown in Figure 3.2. The difference between V^h and V_0^h is that elements in V_0^h are always zero on the boundary.

The scheme (3.8) with piecewise quadratic basis and Simpson's quadrature (3-point Gauss-Lobatto quadrature) has a matrix form $S\mathbf{u} = M\mathbf{f}$ where

S is stiffness matrix and M is the *lumped mass matrix*. On a uniform mesh, it can be written as $H\mathbf{u} = \mathbf{f}$ with

$$H = M^{-1}S = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & & & & \\ -2 & \frac{7}{2} & -2 & \frac{1}{4} & & & & \\ & -1 & 2 & -1 & & & & \\ & \frac{1}{4} & -2 & \frac{7}{2} & -2 & \frac{1}{4} & & \\ & & & -1 & 2 & -1 & & \\ & & & & \ddots & \ddots & & \\ & & & \frac{1}{4} & -2 & \frac{7}{2} & -2 & \\ & & & & & -1 & 2 & \end{pmatrix}$$

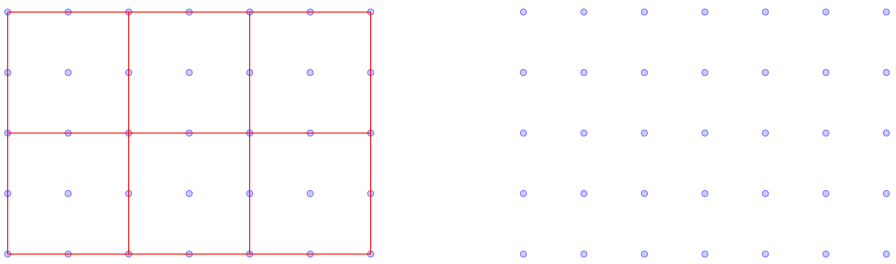
For the nonhomogeneou Dirichlet boundary value problem, the scheme (3.25) can be written as

$$\frac{-u_{i-1} + 2u_i - u_{i+1}}{h^2} = f_i, \quad \text{if } x_i \text{ is a mid point,} \tag{3.30a}$$

$$\frac{u_{i-2} - 8u_{i-1} + 14u_i - 8u_{i+1} + u_{i+2}}{4h^2} = f_i, \quad \text{if } x_i \text{ is a cell end,} \tag{3.30b}$$

where $u_0 = \sigma_0$ and $u_{N+1} = \sigma_1$

Remark 3.20. *The finite difference scheme* (3.30) *can only be defined on a grid with odd number of grid points, because it is obtained by taking end points and midpoints in a finite element mesh as shown in Figure 3.2. This is the only drawback when using* (3.30) *as a finite difference scheme.*



(a) The quadrature points (b) Finite difference grid

Figure 3.4: An illustration of Q^2 element and the 3×3 Gauss-Lobatto quadrature.

Now consider solving $-\Delta u = f$ on a rectangular domain with homogeneous Dirichlet boundary condition. We can use (3.8) with piecewise Q^2 (tensor product of quadratic polynomial) basis and Simpon's rule or 3×3 Gauss-Lobatto quadrature rule on an uniform rectangular mesh as shown in Figure 3.4. With the same array notation in Chapter 2 The equivalent matrix vector form of the scheme (3.8) is

$$(H \otimes I + I \otimes H)vec(U) = vec(F), \tag{3.31}$$

or equivalently

$$(S \otimes M + M \otimes S)vec(U) = (M \otimes M)vec(F).$$

Remark 3.21. *The linear system in* (3.31) *can be easily solved by first computing eigenvalue decomposition of H then the eigenvector method as in the Chapter 2. The eigenvalue decomposition of H can be computed in MATLAB, which is affordable since H is a small matrix compared to $H \otimes I + I \otimes H$.*

Remark 3.22. *The stiffness matrix S is always symmetric and the lumped mass matrix M is diagonal. The matrix H or $H \otimes I + I \otimes H$ is not symmetric, but $S \otimes M + M \otimes S$ is real symmetric. If a symmetric linear system is preferred, then the original symmetric form can be used.*

3.12.2 Neumann b.c.

For one-dimensional homogeneous Neumann boundary, the scheme can be written as

$$\begin{aligned}
\frac{7u_0 - 8u_1 + u_2}{2h^2} &= f_0, \\
\frac{-u_{i-1} + 2u_i - u_{i+1}}{h^2} &= f_i, \quad \text{if } x_i \text{ is a mid point} \\
\frac{u_{i-2} - 8u_{i-1} + 14u_i - 8u_{i+1} + u_{i+2}}{4h^2} &= f_i, \quad \text{if } x_i \text{ is a cell end but not a boundary point,} \\
\frac{u_{N-1} - 8u_N + 7u_{N+1}}{2h^2} &= f_{N+1}.
\end{aligned}$$

3.12.3 The fourth order accuracy as a finite difference scheme

The fourth order accuracy of (3.31) is proved in [9].

The standard finite element error estimate for schemes in this section is third order in L^2-norm. But it can be proven that (3.30) is actually fourth order accurate in the 2-norm over grid points.

First of all, we can check that the finite difference approximation to the second order derivative in (3.30) is only second order accurate, even for the one in (3.30b). Second, if we use this second order approximation to solve a second order PDE such as $-u''(x) = f$, we get a fourth order accurate scheme! As a matter of fact, it can be rigorously proven that this scheme is fourth order accurate for commonly used linear second order PDEs [9, 8] for

- Elliptic equation $-\Delta u = f$.
- Parabolic equation $u_t = \Delta u$.
- Wave equation $u_{tt} = \Delta u$.

- Schrödinger equation $\mathbb{i}u_t = \Delta u$.
- Variable coefficient version of the equations above.

All error estimates in this notes are *a priori* error estimates, which means that the order holds if the exact solution $u(x)$ is smooth enough. For instance, the fourth order accuracy of (3.31) can be proven only if assuming $u \in H^4(\Omega)$. In practice, we often use high order accurate schemes for nonsmooth solutions, for which high order *a priori* error estimates can no longer hold. So a natural question is whether it still makes sense to use a high order accurate scheme like (3.30) on uniform meshes, which is nonethesless often used in applications. In Figure 3.5, there is a comparison of between the second order finite difference (3.23) and the fourth order finite difference(3.30) for solving the following generalized Allen-Cahn equation

$$\phi_t + u\phi_x + v\phi_y = \mu\Delta\phi - \frac{F'(\phi)}{\varepsilon}, \quad (x, y) \in \Omega, \tag{3.32}$$

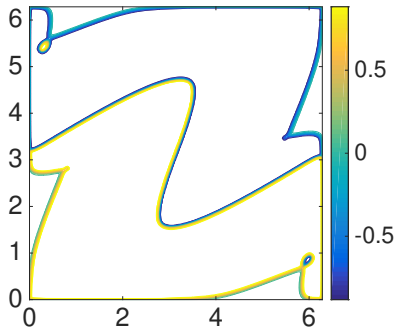
where u, v are given incompressible velocity field, and $F'(\cdot)$ is some fixed energy potential term. With the first order accurate implicit explicit (IMEX) time discretization, it becomes

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} + u^{n+1}\phi_x^{n+1} + v^{n+1}\phi_y^{n+1} = \mu\Delta\phi^{n+1} - \frac{F'(\phi^n)}{\varepsilon}. \tag{3.33}$$

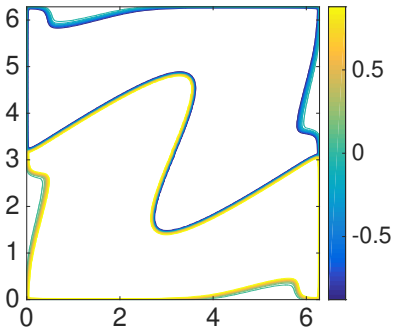
For the differential operators in (3.33), we can used two finite difference schemes derived from P^1 and P^2 finite element method with quadrature. For the second order derivative, they are (3.23) and (3.30). In Figure 3.5, we can see that the solution has a sharp interface, which gives large gradient thus smoothness or regularity of $\phi(x, y)$ is lost, yet the fourth order spatial discretization is still superior because the second order spatial discretization gives a wrong solution on the same coarse 239×239 grid. Higher order time accuracy here does not help the second order spatial discretization on the same coarse 239×239 grid. This is somehow intuitive since usually time evolution is a lot smoother thus spatial error is dominant in these problems.

3.13 Superconvergence

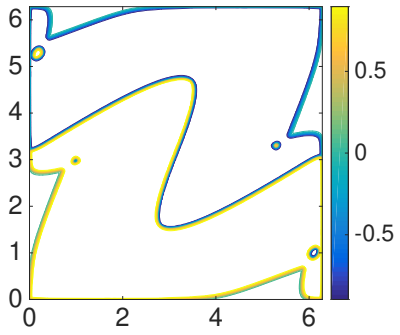
For the scheme (3.30), the error order at quadrature points (two cell ends and the middle point) is one order higher than L^2 error, which is computed for all x in the domain. Such a phenomenon that error at certain points is smaller is called *superconvergence*. On the other hand, it is straightforward to verify that the local truncation error of (3.30a) and (3.30b) is only second order. Recall that the local truncation error is not the true error. The phenomenon that local truncation error at particular locations has lower order than the true error order is called *supraconvergence*. The full proof



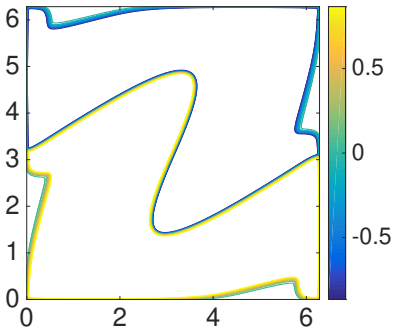
(a) Second order scheme with first order IMEX on a 239×239 grid



(b) Fourth order scheme with first order IMEX on a 239×239 grid



(c) Second order scheme with third order IMEX BDF on a 239×239 grid



(d) Reference Solution

Figure 3.5: Allen-Cahn with log energy at $T = 1.8$.

of why the scheme (3.30) is fourth order accurate in 2-norm over all grid points is quite complicated, see [9, 8]. In this section, we will only see some quick reasons why superconvergence is even possible.

3.13.1 The delta function

Our heuristic understanding of the delta function is $\delta(x) = \begin{cases} +\infty, & x = 0 \\ 0, & x \neq 0 \end{cases}$, which is however not a conventional function at all. One rigorous understanding of it is to define it as a *functional*, mapping any continuous function with compact support $f(x)$ linearly to a number $f(0)$. It is often denoted by an integral, i.e., the definition of the symbol $\delta(x)$ is defined to satisfy

$$\int_{-\infty}^{+\infty} f(x)\delta(x)dx = f(0), \quad \forall f(x) \in C_0(\mathbb{R}).$$

Recall that the function $f(x) = |x|$ is not differentiable but we can define its *weak or generalized derivative* as the step function $g(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$. Now let us compute the weak derivative of the step function by integration by parts:

$$\forall v(x) \in C_0^\infty(\mathbb{R}), \quad \int_{-\infty}^{+\infty} g(x)v'(x)dx = \int_0^{+\infty} v'(x)dx - \int_{-\infty}^{0} v'(x)dx = -2v(0).$$

With the definition of $\delta(x)$ above, we have

$$\forall v(x) \in C_0^\infty(\mathbb{R}) \subset C_0(\mathbb{R}), \quad \int_{-\infty}^{+\infty} v(x)\delta(x)dx = v(0),$$

thus

$$\forall v(x) \in C_0^\infty(\mathbb{R}), \quad \int_{-\infty}^{+\infty} g(x)v'(x)dx = -\int_{-\infty}^{+\infty} v(x)\frac{1}{2}\delta(x)dx.$$

Therefore, we have obtained $\frac{d^2}{dx^2}|x| = 2\delta(x)$, in the weak derivative sense. The symbol $\delta_a(x) := \delta(x - a)$ satisfies

$$\int_{-\infty}^{+\infty} f(x)\delta_a(x)dx = \int_{-\infty}^{+\infty} f(x)\delta(x - a)dx = f(a), \quad \forall f(x) \in C_0(\mathbb{R}).$$

Thus we also have $\frac{d^2}{dx^2}\frac{1}{2}|x - a| = \delta(x - a)$.

3.13.2 The one-dimensional Green's function

For the boundary value problem $-u''(x) = f(x), \quad x \in (0, 1), \quad u(0) = u(1) = 0$, its Green's function $G_a(x)$ is defined to satisfy

$$-\frac{d^2}{dx^2}G_a(x) = \delta_a(x), \quad G_a(0) = G_a(1) = 0,$$

where $a \in (0,1)$ is a fixed number.

Following the discussion in the previous subsection, it is straightforward to verify that

$$G_a(x) = \begin{cases} \frac{1}{2}(1-a)x, & x \le a \\ -\frac{1}{2}ax + \frac{1}{2}a, & x > a \end{cases},$$

thus

$$\frac{d}{dx}G_a(x) = \begin{cases} \frac{1}{2}(1-a), & x \le a \\ -\frac{1}{2}a, & x > a \end{cases}, \quad \frac{d^2}{dx^2}G_a(x) = \delta_a(x).$$

Notice that $G_a(x)$ is a continous piecewise linear function, but this is true only for one-dimensional problem.

3.13.3 Superconvergence at knots in one dimension

For the one-dimensional problem $-u''(x) = f(x), \quad x \in (0,1), \quad u(0) = u(1) = 0$, assume we have a mesh of intervals I_j, on which we define continuous piecewise polynomial spaces V^h and V_0^h.

The abstract finite element method is to seek $u_h \in V_0^h$ satisfying

$$(u_h', v_h') = (f, v_h), \quad \forall v_h \in V_0^h. \tag{3.34}$$

Recall that the solution u_h has Galerkin Orthogonality:

$$(u' - u_h', v_h') = 0, \quad \forall v_h \in V_0^h,$$

Let $e(x) = u(x) - u_h(x) \in H_0^1([0,1]) \subset C_0([0,1])$, then Galerkin Orthogonality can be written as

$$(e', v_h') = 0, \quad \forall v_h \in V_0^h.$$

Let x_i be the cell end of some interval I_j and we call x_i a knot. Then we consider the Green's function at $a = x_i$, e.g, $G_{x_i}(x)$, which is a piecewise linear polynomial defined on the same mesh, thus $G_{x_i}(x) \in V_0^h$. Now let us take a special test function $v_h = G_{x_i}(x)$ in the Galerkin Orthogonality:

$$(e', G_{x_i}(x)') = 0 \Rightarrow \int_0^1 e'(x) G_{x_i}(x)' dx = 0 \Rightarrow -\int_0^1 e(x) \frac{d^2}{dx^2} G_{x_i}(x) dx = 0$$

$$\Rightarrow -\int_0^1 e(x)\delta_{x_i}(x)dx = 0 \Rightarrow -\int_{-\infty}^{+\infty} e(x)\delta_{x_i}(x)dx = 0 \Rightarrow e(x_i) = 0,$$

where we have extended $e(x)$ to the whole real line by zero extension.

This means that the error at knots x_i are zero! Notice that this is the property to the abstract scheme (3.34) for any P^k basis, which we however

do not implement. For instance, for P^1 basis on a uniform mesh, the scheme (3.34) is the same as

$$\frac{1}{h}\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} (f,\phi_1) \\ (f,\phi_2) \\ (f,\phi_3) \\ \vdots \\ (f,\phi_{N-1}) \\ (f,\phi_N) \end{bmatrix}. \tag{3.35}$$

But usually we implement it by approximating the integral (f,ϕ_i) by the trapezoidal rule, which is second order accurate. If we do compute the integrals (f,ϕ_i) exactly, then the scheme (3.35) has zero error.

For the P^2 basis on uniform mesh, the Simpson's rule is exact for the left hand integral (u_h', v_h'), thus (3.34) can be written as

$$\frac{2h}{3}\frac{-u_{i-1}+2u_i-u_{i+1}}{h^2} = (f,\phi_i), \quad \text{if } x_i \text{ is a mid point,} \tag{3.36a}$$

$$\frac{h}{3}\frac{u_{i-2}-8u_{i-1}+14u_i-8u_{i+1}+u_{i+2}}{4h} = (f,\phi_i), \quad \text{if } x_i \text{ is a cell end.} \tag{3.36b}$$

The error of the scheme (3.36) is zero at the cell end x_i (knots). Of course, in the scheme (3.30), we use Simpson's rule for approximating the integrals (f,ϕ_i), which is fourth order accurate. So at least now intuitively it is not a surprise that the scheme (3.30) should be fourth order accurate at the knots. For the fourth order accurate at the midpoint, we need some more arguments, which will not be explained in this notes.

Remark 3.23. *In general, by the standard superconvergence theory of P^k ($k \geq 2$) finite element method* (3.34) *(even for a variable coefficient problem in multiple dimensions), function values of $u_h(x)$ are $(k+2)$-th order accurate at Gauss-Labotto points for each small interval in 2-norm, as opposed to $(k+1)$-th order in the L^2-norm error estimate, and derivatives of $u_h(x)$ are $(k+2)$-th order accurate at Gauss points, as opposed to k-th order in the H^1-norm error estimate.*

3.14 Comparison with traditional finite difference method

3.14.1 Advantages of the finite element method

Troughout this chapter, we have seen many things that cannot happen or cannot be explained in the traditional finite difference method. Even on

uniform meshes for a rectangular domain, the finite element method is still superior from any perspective, because it gives us a finite difference with all desired properties. We summerize some comparsions in Table 3.1.

3.14.2 Limitations of the finite element method

In general, the finite element method is quite successful, for solving an elliptic equation $-\Delta u = f$ or some other types of equations including parabolic equations $u_t = \Delta u$, wave equations $u_{tt} = \Delta u$, Schrödinger equation $\mathbb{i}u_t = \Delta u$, etc. These equations all contain the Laplacian operator $-\Delta u$, for which a *coercive* bilinear form $\mathcal{A}(u, v) = (u', v')$ can be defined. Another different example is the biharmonic equation $u''''(x) = f$, for which we can also define a similar variational formulation with coercivity, thus the finite element method for this kind of fourth order PDE is also quite successful.

The foundation of the success for the finite element method, when missing, is also source of the limitations of the finite element method in applications. It could be quite or extremely difficult to use finite element method for equations lack of coercive operators. One simple example of such equations is the simple convection $u_t + u_x = 0$ which will be discussed in Chapter 7, or its nonlinear version *nonlinear conservation laws* $u_t + f(u)_x = 0$ which will be discussed in Chapter 9. Another example is the *Hamilton-Jacobi* equation $u_t + f(u_x) = 0$, e.g., $u_t + |\nabla u| = 0$, which is also closely related to *nonlinear conservation laws*. A formal application of the finite element method to these equations, with certain modifications to achieve stability or even convergence, is always possible, but many provable properties in this chapter will be no longer true.

Table 3.1: Comparison of traditional FD and finite element method for solving $-\nabla \cdot (A\nabla u) = f$ on Ω.

	traditional FD	FEM
Equation	approximates PDE	approximates variational form
Boundary condition	direct approximation	absorbed in V_0^h and variational form
Curved domain	a mapping to rectangular grid	Ω is easily approximated by Ω_h
Rectangular Ω	a rectangular grid	becomes finite difference
S matrix	nonsymmetric in general	always symmetric
Consistency	Taylor expansion	Galerkin Orthogonality
Stability	singular values	coercivity
Convergence	in 2-norm	H^1 and L^2 estimates
General tools	Calculus & Linear Algebra	functional analysis, PDE theory, etc
Error order	truncation error order	interpolation error order
Higher order schemes	large stencil, inducing difficulty near boundary	no difficulty at the boundary
Variable coefficient	difficult to construct higher order schemes	easy to to construct higher order schemes
Superconvergence	N/A	P^2 gives a 4th order FD
General implementation	form a matrix directly	computing some $S_{ij} = \mathcal{A}(\phi_j, \phi_i)$ to get S
Rectangular Ω	just solve a linear system	implement it as a FD scheme
Purely Neumann b.c.	left null vector is expensive to compute	left null vector is always $\mathbf{1}$

4

Fourier Analysis

This chapter will be a very brief introduction to Fourier transform, Semidiscrete Fourier transform, the discrete Fourier transform, and Fourier series.

4.1 The Fourier transform

We will take the Fourier transform of integrable functions of one variable $x \in \mathbb{R}$.

Definition 4.1. *(Integrability) A function f is called integrable, or absolutely integrable, when*

$$\int_{-\infty}^{\infty} |f(x)|\, dx < \infty$$

in the sense of Lebesgue integration. One also writes $f \in L^1(\mathbb{R})$ for the space of integrable functions.

We denote the physical variable as x, but it is sometimes denoted by x in contexts in which its role is time, and one wants to emphasize that. The frequency, or wavenumber variable is denoted k. Popular alternatives choices for the frequency variable are ω (engineers) or ξ (mathematicians), or p (physicists).

Definition 4.2. *The Fourier transform (FT) of an integrable function $f(x)$ is defined as*

$$\hat{f}(k) = \int_{-\infty}^{\infty} e^{-\mathrm{i}kx} f(x)\, dx. \tag{4.1}$$

When $\hat{f}(k)$ is also integrable, $f(x)$ can be recovered from $\hat{f}(k)$ by means of the inverse Fourier transform (IFT)

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{\mathrm{i}kx} \hat{f}(k)\, dk. \tag{4.2}$$

Intuitively, $\hat{f}(k)$ is the amplitude density of f at frequency k. The formula for recovering f is a decomposition of f into constituent waves. The justification of the inverse FT formula belongs in a real analysis class. We will justify the form of (4.2) heuristically when we see Fourier series in the next section. The precaution of assuming integrability is so that the integrals can be understood in the usual Lebesgue sense. In that context, taking integrals over infinite intervals is perfectly fine. If (4.1) and (4.2) are understood as limits of integrals over finite intervals, it does not matter how the bounds are chosen to tend to $\pm\infty$. One may in fact understand the formulas for the FT and IFT for much larger function classes than the integrable functions, namely distributions, but this is also beyond the scope of the class. We will generally not overly worry about these issues. It is good to know where to draw the line: the basic case is that of integrable functions, and anything beyond that requires care and adequate generalizations. Do not be surprised to see alternative formulas for the Fourier transform in other classes or other contexts. Here are some important properties of Fourier transforms:

- (Differentiation)
$$\widehat{f'(x)} = \mathbb{i}k\hat{f}(k).$$
Justification: integration by parts in the integral for the FT.
- (Translation) If $g(x) = f(x + a)$, then
$$\hat{g}(k) = e^{\mathbb{i}ka}\hat{f}(k).$$
Justification: change of variables in the integral for the FT.

Another basic property of Fourier transforms is the convolution theorem.

Theorem 4.1. *Denote the convolution as* $f * g(x) = \int_\infty^\infty f(y)g(x - y)\,dy$. *Then*
$$\widehat{f * g}(k) = \hat{f}(k)\hat{g}(k).$$

The Fourier transform is an important tool in the study of linear differential equations because it turns differential problems into algebraic problems. For instance, consider a polynomial $\mathcal{P}(x) = \sum a_n x^n$ and the ODE
$$\mathcal{P}\left(\frac{d}{dx}\right) u(x) = f(x), \quad x \in \mathbb{R}.$$
which means $\sum a_n \frac{d^n}{dx^n} u(x) = f(x)$. Upon Fourier transformation, the equation becomes
$$\mathcal{P}(ik)\hat{u}(k) = \hat{f}(k),$$
which is simply solved as
$$\hat{u}(k) = \frac{\hat{f}(k)}{\mathcal{P}(ik)},$$

and then

$$u(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{\mathbb{i}kx} \frac{\hat{f}(k)}{\mathcal{P}(ik)}\, dk.$$

Beware the zeros of $\mathcal{P}$ when applying this formula. They always carry important physical interpretation. For instance, they could be resonances of a mechanical system. The formula $\hat{u}(k) = \frac{\hat{f}(k)}{\mathcal{P}(ik)}$ also lends itself to an application of the convolution theorem. Let $K(x)$ be the inverse Fourier transform of $1/\mathcal{P}(ik)$. Then we have

$$u(x) = K(x-y)f(y)\, dy.$$

The function K is called Green's function for the original ODE.

4.2 Sampling and restriction

We aim to use Fourier transforms as a concept to help understand the accuracy of representing and manipulating functions on a grid, using a finite number of degrees of freedom. We also aim at using a properly discretized Fourier transform as a numerical tool itself. For this purpose, $x \in \mathbb{R}$ and $k \in \mathbb{R}$ must be replaced by x and k on finite grids. Full discretization consists of sampling and restriction. Let us start by sampling $x \in h\mathbb{Z}$, i.e., considering $x_j = jh$ for $j \in \mathbb{Z}$. The important consequence of sampling is that some complex exponential waves $e^{\mathbb{i}kx}$ for different k will appear to be the same on the grid x_j. We call *aliases* such functions that identify on the grid.

Definition 4.3. *(Aliases) The functions $e^{\mathbb{i}k_1 x}$ and $e^{\mathbb{i}k_2 x}$ are aliases on the grid $x_j = jh$ if*

$$e^{\mathbb{i}k_1 x_j} = e^{\mathbb{i}k_2 x_j}, \quad \forall j \in \mathbb{Z}.$$

Aliases happen if

$$k_1 jh = k_2 jh + 2\pi n, \quad n \in \mathbb{Z}.$$

Letting $j = 1$, we have

$$k_1 - k_2 = \frac{2\pi}{h} n.$$

Two wave numbers k_1 and k_2 are indistinguishable on the grid if they differ by an integer multiple of $2\pi/h$. For this reason, we restrict without loss of generality the wavenumber to the interval $k \in [-\pi/h, \pi/h]$. We also call this interval the fundamental cell in frequency. Real-life examples of aliases are rotating wheels looking like they go backwards in a movie, Moiré patterns on jackets on TV, and stroboscopy. The proper notion of Fourier transform on a grid is the following.

Definition 4.4. *Let $x_j = hj$, $f_j = f(x_j)$. Semidiscrete Fourier transform (SFT):*

$$\hat{f}(k) = h \sum_{j=-\infty}^{\infty} e^{-\mathbb{i}kx_j} f_j, \quad k \in [-\frac{\pi}{h}, \frac{\pi}{h}]. \tag{4.3}$$

Inverse semidiscrete Fourier transform (ISFT):

$$f_j = \frac{1}{2\pi} \int_{-\frac{\pi}{h}}^{\frac{\pi}{h}} e^{\mathbb{i}kx_j} \hat{f}(k) dk. \tag{4.4}$$

As we saw, sampling in x corresponds to a restriction in k. If one still wanted to peek outside $[-\frac{\pi}{h}, \frac{\pi}{h}]$ for the SFT, then the SFT would simply repeat by periodicity:

$$\hat{f}(k + \frac{2n\pi}{h}) = \hat{f}(k),$$

(why?). That's why we restrict it to the fundamental cell.

Remark 4.1. *Assume $\hat{f}(k) = 0$ if $|k| > \frac{\pi}{h}$. Then (4.4) is the same as (4.2), which implies that no error is made in sampling and interpolating $f(x)$ at rate h. This is known as the Shannon sampling theorem: a function bandlimited in $[-\frac{\pi}{h}, \frac{\pi}{h}]$ in k space is perfectly interpolated by bandlimited interpolation, on a grid of spacing h or smaller.*

Theorem 4.2 (Nyquist-Shannon Sampling Theorem)**.** *Let $\hat{f}(k) = \int_{-\infty}^{\infty} e^{-\mathbb{i}kx} f(x)\, dx$. If $\hat{f}(k) = 0$ for $|k| \geq \pi$, then*

$$f(x) = \sum_{n=-\infty}^{+\infty} f(n) \frac{\sin \pi(x-n)}{\pi(x-n)}, \quad \forall x \in \mathbb{R}.$$

Proof. Consider the Fourier Series of $\hat{f}(k)$ on the interval $[-\pi, \pi]$:

$$\hat{f}(k) = \frac{1}{2\pi} \sum_{n=-\infty}^{+\infty} c_n e^{\mathbb{i}\, nk},$$

where

$$\begin{aligned} c_n &= \int_{-\pi}^{\pi} e^{-\mathbb{i}\, nk} \hat{f}(k) dk \\ &= \int_{-\infty}^{+\infty} e^{-\mathbb{i}\, nk} \hat{f}(k) dk \quad (\hat{f} \text{ has compact support}) \\ &= 2\pi f(-n) \quad (\text{the inverse Fourier transform}). \end{aligned}$$

So

$$\hat{f}(k) = \sum_{n=-\infty}^{+\infty} f(-n) e^{\mathbb{i}\, nk} = \sum_{n=-\infty}^{+\infty} f(n) e^{-\mathbb{i}\, nk}. \tag{4.5}$$

Finally we have

$$\begin{aligned} f(x) &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{\mathbb{i}\,xk}\hat{f}(k)dk \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\mathbb{i}\,xk}\hat{f}(k)dk \quad (\hat{f} \text{ has compact support}) \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\mathbb{i}\,xk}\left(\sum_{n=-\infty}^{+\infty} f(n)e^{-\mathbb{i}\,nk}\right)dk \\ &= \sum_{n=-\infty}^{+\infty} f(n)\left(\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\mathbb{i}(x-n)k}dk\right) \quad (\text{if } \int\sum|f(n)| < +\infty \text{ by Fubini's Theorem }) \end{aligned}$$

□

Remark 4.2. *If $\hat{f}(k) = 0$ for $|k| \geq \frac{\pi}{h}$, then the sampling points should be jh, $j \in \mathbb{Z}$. In particular,* (4.5) *implies that SFT* (4.3) *is equivalent to FT* (4.1) *for the bandlimited functions.*

We can now define the proper notion of Fourier analysis for functions that are restricted to x in some interval, namely $[-\pi, \pi]$ for convention. Then the frequency is sampled as a result. The following formulas are dual to those for the SFT.

Definition 4.5. *Fourier series (FS):*

$$\hat{f}_k = \int_{-\pi}^{\pi} e^{-\mathbb{i}\,kx} f(x)\,dx. \tag{4.6}$$

Inverse Fourier series (IFS)

$$f(x) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} e^{\mathbb{i}\,kx}\hat{f}_k, \quad x \in [-\pi, \pi]. \tag{4.7}$$

If one uses the Fourier series inversion formula for x outside of its intended interval $[-\pi, \pi]$, then the function simply repeats by periodicity: $f(x + 2n\pi) = f(x)$. (again, why?)

The two formulas of (4.6) and (4.7) can be justified quite intuitively. The expression $\int f(x)\overline{g(x)}\,dx$ is an inner product on functions. It is easy to see that the complex exponentials form an orthonormal set of functions on $[-\pi/h, \pi/h]$, for this inner product. Hence, up to normalization constants, (4.6) is simply calculation of the coefficients in an orthobasis (analysis), and (4.7) is the synthesis of the function back from those coefficients. We'd have to understand more about the peculiarities of infinite-dimensional linear algebra to make this fully rigorous, but this is typically done in a real analysis class.

Sampling in x corresponds to restriction/periodization in k, and restriction/periodization in k corresponds to sampling in x.

4.3 The DFT and its algorithm, the FFT

The discrete Fourier transform is what is left of the Fourier transform when both space and frequency are sampled and restricted to some interval. Consider $x_j = jh, j = 1, ..., N$. The point $j = 0$ is identified with $j = N$ by periodicity, so it is not part of the grid. If the endpoints are $x_0 = 0$ and $x_N = 2\pi$, then N and h relate as $h = \frac{2\pi}{N} \Rightarrow \frac{\pi}{h} = \frac{N}{2}$.

For the dual grid in frequency, consider that N points should be equispaced between the bounds $[-\pi/h, \pi/h]$. The resulting grid is

$$k = -\frac{N}{2} + 1, ..., \frac{N}{2}.$$

We have the following definition.

Definition 4.6. *Discrete Fourier transform (DFT):*

$$\hat{f}_k = h \sum_{j=1}^{N} e^{-\mathbb{i}\, kjh} f_j, \quad k = -\frac{N}{2} + 1, ..., \frac{N}{2}. \tag{4.8}$$

Inverse discrete Fourier transform (IDFT)

$$f_j = \frac{1}{2\pi} \sum_{k=-\frac{N}{2}+1}^{\frac{N}{2}} e^{\mathbb{i}\, kjh} \hat{f}_k, \quad j = 1, \cdots, N. \tag{4.9}$$

The DFT can be computed as is, by implementing the formula (4.8) directly on a computer. The complexity of this calculation is a $\mathcal{O}(N^2)$, since there are N values of j, and there are N values of k over which the computation must be repeated. There is, however, a smart algorithm that allows to group the computation of all the f_k in complexity $\mathcal{O}(N \log N)$. It is called the fast Fourier transform (FFT). It is traditionally due to Tukey and Cooley (1965), but the algorithm had been discovered a few times before that by people who are not usually credited as much: Danielson and Lanczos in 1942 [1], as well as Gauss in 1805.

[1] Danielson, Gordon C.; Lanczos, Cornelius (1942). "Some improvements in practical Fourier analysis and their application to X-ray scattering from liquids". Journal of the Franklin Institute. 233 (4): 365–380. The Danielson-Lanczos lemma is the basis of FFT. Danielson and Lanczos performed their work in the late 1930's at Purdue University, where Cornelius Lanczos (1893-1974) was a professor of mathematical physics from 1931-1946. Gordon Danielson (1912-83) was a graduate student in physics at Purdue working on applications of Fourier analysis to X-ray scattering. Danielson became a professor of physics at Iowa State University in 1948 and a distinguished professor in 1964.

4.4 Smoothness and truncation

In this section, we study the accuracy of truncation of Fourier transforms to finite intervals. This is an important question not only because real-life numerical Fourier transforms are restricted in k, but also because, as we know, restriction in k serves as a proxy for sampling in x. Every claim that we make concerning truncation of Fourier transforms will have an implication in terms of accuracy of sampling a function on a grid, i.e., how much information is lost in the process of sampling a function $f(x)$ at points $x_j = jh$. We will manipulate functions in the spaces L^1, L^2, and L^∞. We have already encountered L^1.

Definition 4.7. *Let $1 \leq p \leq \infty$. A function f of $x \in \mathbb{R}$ is said to belong to the space $L^p(\mathbb{R})$ when $\int_{-\infty}^{\infty} |f(x)|^p \, dx < \infty$. Then the norm of f in $L^p(\mathbb{R})$ is $\left(\int_{-\infty}^{\infty} |f(x)|^p \, dx\right)^{\frac{1}{p}}$. A function f of $x \in \mathbb{R}$ is said to belong to $L^\infty(\mathbb{R})$ when ess sup$|f(x)| < \infty$. Then the norm of f in $L^\infty(\mathbb{R})$ is ess sup$|f(x)|$.*

In the definition above, "ess sup" refers to the essential supremum, i.e., the infimum over all dense sets $X \subset \mathbb{R}$ of the supremum of f over X. A set X is dense when $\mathbb{R} \setminus X$ has measure zero. The notions of supremum and infimum correspond to maximum and minimum respectively, when they are not necessarily attained. All these concepts are covered in a real analysis class. For us, it suffices to heuristically understand the L^∞ norm as the maximum value of the modulus of the function, except possibly for isolated points of discontinuity which don't count in calculating the maximum. It is an interesting exercise to relate the L^∞ norm to the sequence of L^p norms as $p \to \infty$. We will need the very important Parseval and Plancherel identities. They express "conservation of energy" from the physical domain to the frequency domain.

Theorem 4.3. *(Parseval's identity). Let $f, g \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$. Then*

$$\int_{-\infty}^{\infty} f(x)\overline{g(x)} \, dx = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(k)\overline{\hat{g}(k)} \, dk.$$

Theorem 4.4. *(Plancherel's identity). Let $f \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$. Then*

$$\int_{-\infty}^{\infty} |f(x)|^2 \, dx = \frac{1}{2\pi} \int_{-\infty}^{\infty} |\hat{f}(k)|^2 \, dk. \tag{4.10}$$

(With the help of these formulas, it is in fact possible to extend their validity and the validity of the FT to $f, g \in L^2(\mathbb{R})$, and not simply $f, g \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$. This is a classical density argument covered in many good analysis texts.) We need one more concept before we get to the study of truncation of Fourier transforms. It is the notion of total variation. We assume that the reader is familiar with the spaces $C^k(\mathbb{R})$ of bounded functions which are k times continuously differentiable.

Definition 4.8. *(Total variation) Let $f \in C^1(\mathbb{R})$. The total variation of f is the quantity*

$$\|f\|_{TV} = \int_{-\infty}^{\infty} |f'(x)|\,dx. \tag{4.11}$$

For functions that are not C^1, the notion of total variation is given by either expression

$$\|f\|_{TV} = \lim_{h\to 0} \int_{-\infty}^{\infty} \frac{|f(x) - f(x-h)|}{|h|}\,dx = \sup_{\{x_p\} \text{ finite subset of } \mathbb{R}} \sum_p |f(x_{p+1}) - f(x_p)|.$$

These more general expressions reduce to $\int_{-\infty}^{\infty} |f'(x)|\,dx$ when $f \in C^1(\mathbb{R})$. When a function has finite total variation, we say it is in the space of functions of bounded variation, or $BV(\mathbb{R})$.

The total variation of a piecewise constant function is simply the sum of the absolute value of the jumps it undergoes. This property translates to a useful intuition about the total variation of more general functions if we view them as limits of piecewise constant functions. The important meta-property of the Fourier transform is that decay for large $|k|$ corresponds to smoothness in x. There are various degrees to which a function can be smooth or rates at which it can decay, so therefore there are several ways that this assertion can be made precise. Let us go over a few of them. Each assertion either expresses a decay (in k) to smoothness (in x) implication, or the converse implication.

- Let $\hat{f} \in L^1(\mathbb{R})$ (decay), then $f \in L^\infty(\mathbb{R})$ and f is continuous (smoothness). That's because $|e^{\mathbb{i}\,kx}| = 1$, so

  $$|f(x)| \le \frac{1}{2\pi} \int_{-\infty}^{\infty} |e^{\mathbb{i}\,kx} \hat{f}(k)|\,dk = \frac{1}{2\pi} \int_{-\infty}^{\infty} |\hat{f}(k)|\,dk,$$

  which proves boundedness. As for continuity, consider a sequence $y_n \to 0$ and the formula $f(x - y_n) = \int_{-\infty}^{\infty} \frac{1}{2\pi} e^{\mathbb{i}\,k(x-y_n)} \hat{f}(k)\,dk$. The integrand converges modulus by the integrable pointwise function to $e^{\mathbb{i}\,kx}\hat{f}(k)$, and is uniformly bounded. Hence Lebesgue's dominated convergence theorem applies and yields $f(x - y_n) \to f(x)$, i.e., continuity in x.

- Let $\hat{f}(k)(1 + |k|^p) \in L^1(\mathbb{R})$ (decay). Then $f \in C^p$ (smoothness). We saw the case $p = 0$ above; the justification is analogous in the general case. We write

  $$|f^{(n)}(x)| \le \frac{1}{2\pi} \int_{-\infty}^{\infty} |e^{\mathbb{i}\,kx} (\mathbb{i}\,k)^n \hat{f}(k)|\,dk = \frac{1}{2\pi} \int_{-\infty}^{\infty} |k|^n |\hat{f}(k)|\,dk,$$

  which needs to be bounded for all $0 \le n \le p$. This is obviously the case if $(1 + |k|^p)\hat{f}(k) \in L^1(\mathbb{R})$. Continuity of $f^{(p)}$ is proved like before.

- Let $f \in BV(\mathbb{R})$ (smoothness). Then $\hat{f}(k) \leq \|f\|_{TV}|k|^{-1}$ (decay). If $f \in C^1 \cap BV(\mathbb{R})$, then this is justified very simply from (4.11), and

$$\mathbb{i}\, k\hat{f}(k) = \int_{-\infty}^{\infty} e^{-\mathbb{i}\, kx} f'(x)\, dx$$

  Take a modulus on both sides, and get the desired relation. When $f \in BV(\mathbb{R})$, but $f \notin C^1$, either of the more general formulas for the total variation definition must be used instead. It is a great practice exercise to articulate a modified proof using the $\lim_{h\to 0}$ formula, and properly pass to the limit.

- Let f satisfy $f^{(n)} \in L^2(\mathbb{R})$ for $0 \leq n < p$, and assume $f^{(p)} \in BV(\mathbb{R})$ (smoothness). Then there exists $C > 0$ such that $|\hat{f}(k)| \leq |k|^{-p-1}$ (decay). The justification is very simple when $f \in C^{p+1}$: we then get

$$(\mathbb{i}\, k)^{p+1}\hat{f}(k) = \int_{-\infty}^{\infty} e^{-\mathbb{i}\, kx} f^{(p+1)}(x)\, dx,$$

  so

$$|\,\mathbb{i}\, k|^{p+1}|\hat{f}(k)| \leq \int_{-\infty}^{\infty} |e^{-\mathbb{i}\, kx}||f^{(p+1)}(x)|\, dx = \|f^{(p)}\|_{TV} \leq \infty.$$

  Again, it is a good exercise to try and extend this result to functions not in C^{p+1}.

5

Well Posedness

In this chapter we consider initial value, linear partial differential equations, and address the concept of well *posedness* of the problem.

5.1 Definition and examples

Before attempting to approximate the solution of a partial differential equation by numerical methods, one has to analyze some of the basic properties of the problem itself and its solution. Roughly speaking, the solution of a given problem has to be a function of its initial values, since the future states of the system must be completely determined by the dynamics of the system together with the initial state. For those who prefer a more "mathematical" formulation of the above statement, it means that we are looking for a representation of the solution $u(x,t)$ of a partial differential equation as a function of the form:

$$u(x,t) = S(t,t_0)u(x,t_0); t \geq t_0$$

here S is an operator, called the solution operator, such that the above expression satisfies thee partial differential equation. The first step in analyzing the properties of the system dynamics is to answer the basic questions:

Does there exist the solution at all? That is, if such an operator S exists. If it does, how does the solution depend on the initial functions? In other words: what is the domain of S? Is S a bounded operator? Finally, how smooth is the solution? (the range of S might include even non-differentiable functions). Well posedness is a property of the partial differential equation, related to particular answers to these questions. We shall make this concept clear by examing some examples before stating the formal definitions.

The general problem that we study is concerned with linear, homogeneous, partial differential equations with initial values and can be stated as follows:

Find a vector valued function of p components $u(x,t) = (u_1(x,t), ..., u_p(x,t))$, where $x = (x_1, ...x_s)$ and $t \geq 0$, that satisfies the equation:

$$\begin{aligned} u_t(x,t) &= \mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) u(x,t), \\ u(x,0) &= f(x), \end{aligned} \tag{5.1}$$

where $\mathcal{P}$ is a polynomial in the operator argument $\frac{\partial}{\partial x}$. For example, if x is a scalar (s = 1), then $\mathcal{P}$ has the form:

$$\mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) = \sum_{k=1}^{r} a_k(x,t) \frac{\partial^k}{\partial x^k}$$

and it is a polynomial of degree r if $a_r(x,t)$ does not vanish identically.

If $x = (x_1, ..., x_s)$, let $\alpha = (\alpha_1, ..., \alpha_s)$ denote a multi-index, i.e., each component α_j is an integer, and use the notation:

$$\frac{\partial^\alpha}{\partial x^\alpha} = \frac{\partial^{\alpha_1 + \cdots + \alpha_x}}{\partial x_1^{\alpha_1} \cdots \partial x_s^{\alpha_s}}$$

It is customary to write $|\alpha|$ for $\alpha_1 + \cdots + \alpha_s$, so that in the multidimensional case the general form of the operator $\mathcal{P}$ is:

$$\mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) = \sum_{\alpha : |\alpha| \leq r} a_\alpha(x,t) \frac{\partial^\alpha}{\partial x^\alpha}$$

and if for some α with $\alpha = r$ the function $a_\alpha(x,t)$ is not zero, then $\mathcal{P}$ is a polynomial of degree r.

Definition 5.1. *If $\mathcal{P}$ is a polynomial of degree r in $\frac{\partial}{\partial x}$, then we call $u(x,t)$ a classical solution of the problem* (5.1) *if u has continuous derivatives up to order r in space and first continuous derivative in time, provided that $u(x,t)$ satisfies* (5.1).

An important quantity in analyzing whether the solution of (5.1) is well defined (that is, bounded in some suitable norm) is that of the *energy* of the system. The definition of an energy for a particular system depends on its physical properties. Nonetheless, we shall generally define the energy in terms of some norm (induced by an inner product) of the solution. Indeed, for each $t \geq 0$, we can regard $u(t)$ as a function on the a Hilbert space where different definitions of inner products give rise to different norms. We will often define the energy as the L^2-norm of $u(t)$, in which case it takes the form:

$$E(t) = \int |u(x,t)|^2 \, dx_1 \cdots dx_s,$$

where x is the vector of size s.

More generally, if $\langle f, g\rangle$ denotes the inner product on $\mathbb{R}^p$, we can define:

$$E(t) = \int \langle u(x,t), u(x,t)\rangle \, dx_1 \cdots dx_s.$$

We will illustrate the concept of energy and its properties through the examples that follow, so that the reader might become familiar with it in a more natural way.

Example 5.1. *Consider the one-way wave equation, that is, the differential operator $\mathcal{P}$ is given by:*

$$\mathcal{P}(x,t,\frac{\partial}{\partial x}) = a\frac{\partial}{\partial x}$$

for x a real variable and a any constant. This yields the problem:

$$u_t = au_x, \quad u(x,0) = f(x).$$

In order to analyze the behavior of the solution $u(x,t)$ of the problem, we will use Fourier transforms. Recall that if $\hat{f}(\omega)$ denotes the Fourier transform of the function $f(x)$, then:

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

and for the function $u(x,t)$ we have:

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

As already mentioned in the previous chapter, smoothness of the initial function $f(x)$ is reflected in the behavior of $\hat{f}(\omega)$ for large values of ω. The extreme case is to consider band limited initial functions f for which $\hat{f}(\omega) = 0$ for $|\omega| \geq \omega_0$, but we shall not investigate this case in the general formulation because it is too restrictive.

By taking the Fourier transform in x in both the PDE and the initial condition, we get:

$$\frac{\partial \hat{u}}{\partial t} = \mathbb{i}\, a\omega\hat{u}, \quad \hat{u}(\omega,0) = \hat{f}(\omega),$$

which is an ordinary differential equation. By means of Fourier transforms we can therefore reduce a partial differential equation in the physical space x into an ordinary differential equation in the Fourier space ω. Solving this ODE problem, we get:

$$\hat{u}(\omega,t) = e^{\mathbb{i}\,\omega a t}\hat{f}(\omega),$$

so that the solution is given by:

$$u(x,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{u}(\omega,t) e^{\mathbb{i}\,\omega x}\, d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega) e^{\mathbb{i}\,\omega(at+x)}\, d\omega = f(x+at).$$

This method has other advantages. For instance, by Plancherel's identity we know that:

$$\int_{-\infty}^{\infty} |u(x,t)|^2\,dx = \frac{1}{2\pi}\int_{-\infty}^{\infty} |\hat{u}(\omega,t)|^2\,d\omega.$$

For this example, where $|\hat{u}(\omega,t)|^2 = |\hat{f}(\omega)e^{\mathbb{i}\,\omega at}|^2 = |\hat{f}(\omega)|^2$*, we have, using again Plancherel's identity on* f *that:*

$$\int_{-\infty}^{\infty} |u(x,t)|^2\,dx = \int_{-\infty}^{\infty} |u(x,0)|^2\,dx$$

for all time $t \geq 0$*, which is nothing but the conservation of energy. This fact also tells us something about the existence of solutions: if* $\int_{-\infty}^{\infty} |f(x)|^2\,dx\langle\infty$*, then the solution remains bounded at any given tine (that is, there is no "blow-up").*

Remark 5.1. *In the equation* $u_t - au_x = 0$ *there appear one derivative in space and one in time. Space and time here play an interchangeable role, a fact that is reflected in the solution itself.*

Example 5.2. *We shall consider now a very different example, where the equation is not reversible in time:*

$$u_t(x,t) = au_{xx}(x,t), \quad a > 0, \quad u(x,0) = f(x).$$

Using again Fourier transforms we obtain in the Fourier space the following ordinary differential equation:

$$\frac{\partial\,\hat{u}}{\partial\,t}(\omega,t) = a(\mathbb{i}\,\omega)^2\hat{u}(\omega,t) = -a\omega^2\hat{u}(\omega,t), \quad \hat{u}(\omega,0) = \hat{f}(\omega),$$

which yields:

$$\hat{u}(\omega,t) = \hat{f}(\omega)e^{-a\omega^2 t},$$

and thus

$$u(x,t) = \frac{1}{2\pi}\int_{\infty}^{\infty} \hat{f}(\omega)e^{-a\omega^2 t}e^{\mathbb{i}\,\omega x}\,d\omega.$$

Since $e^{-a\omega^2 t} \in (0,1)$ *for all* ω *and all* $t > 0$*, we have that* $|\hat{u}(\omega,t)| \leq |\hat{f}(\omega)|$*, and using Plancherel's identity we obtain for the energy of the system the inequality:*

$$E(t) = \int_{\infty}^{\infty} |u(x,t)|^2\,dx = \frac{1}{2\pi}\int_{\infty}^{\infty} |\hat{u}(\omega,t)|^2\,d\omega \leq \frac{1}{2\pi}\int_{\infty}^{\infty} |\hat{f}(\omega)|^2\,d\omega = E(0),$$

which implies that the energy decreases in time.

This equation is known as the heat conduction equation. It is parabolic and its Properties are quite different from those of hyperbolic equations like the previous example. Some of these properties are:

1. The energy is being dissipated,
2. the system tends to a steady state, known as the equilibrium: $\lim_{t\to\infty} u(x,t) = 0$
3. From the expression for $\hat{u}$ we see that roughness of the initial data is smoothed out, since even if $\hat{f}(\omega)$ is large for high modes, $\hat{u}(\omega,t)$ is decreasing in time. Therefore, due to dissipation, the solution becomes smoother as time goes by.

Example 5.3. *We go back now to study hyperbolic equations, considering the two-way wave equation:*

$$\begin{aligned} u_{tt} &= u_{xx} \\ u(x,0) &= f_1(x) \\ u_t(x,0) &= f_2(x) \end{aligned}$$

First we rewrite this equation in the general form stated at the beginning of this chapter, where only one time derivative appears. For this purpose, assume the solution $u(x,t)$ exists and let v be a function defined such that:

$$\begin{aligned} \frac{\partial v}{\partial x} &= \frac{\partial u}{\partial t}, \\ \frac{\partial v}{\partial t} &= \frac{\partial u}{\partial x}. \end{aligned}$$

So we get,

$$\frac{\partial}{\partial t}\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \frac{\partial}{\partial x}\begin{pmatrix} u \\ v \end{pmatrix}.$$

Since $v_x(x,0) = u_t(x,0) = f_2(x)$, we have,

$$v(x,0) = F_2(x) \equiv \int_0^x f_2(\xi)\, d\xi, \quad u(x,0) = f_1(x).$$

In the notation of expressions (5.1) *we have:*

$$\mathcal{P}(x,t,\frac{\partial}{\partial x}) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \frac{\partial}{\partial x},$$

$$\begin{pmatrix} u \\ v \end{pmatrix}(x,0) = f(x) = \begin{pmatrix} f_1(x) \\ F_2(x) \end{pmatrix}.$$

Let $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, then the ODEs that are satisfied by this system in the Fourier space are:

$$\frac{\partial}{\partial t}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \mathbb{i}\,\omega A \begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \begin{pmatrix} \mathbb{i}\,\omega\hat{u} \\ \mathbb{i}\,\omega\hat{v} \end{pmatrix}$$

which we can decouple into two simpler equations, namely:

$$\frac{\partial}{\partial t}(\hat{u}+\hat{v}) = \mathbb{i}\,\omega(\hat{u}+\hat{v})$$
$$\frac{\partial}{\partial t}(\hat{u}-\hat{v}) = -\mathbb{i}\,\omega(\hat{u}-\hat{v})$$

Both equations can now be treated separately in the same way as the one-way wave equation.

Remark 5.2. *To rewrite the two-way wave equation as a first order system, we can also introduce new functions*

$$v(x,t) = u_x(x,t), \quad w(x,t) = u_t(x,t),$$

then we get

$$v_t = u_{xt} = w_x, \quad w_t = u_{tt} = u_{xx} = v_x,$$

thus

$$\frac{\partial}{\partial t}\begin{pmatrix} v \\ w \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \frac{\partial}{\partial x}\begin{pmatrix} v \\ w \end{pmatrix},$$

$$\begin{pmatrix} v \\ w \end{pmatrix}(x,0) = \begin{pmatrix} f_1'(x) \\ f_2(x) \end{pmatrix}.$$

The new initial conditions involve $f_1'(x)$ so we can get only weakly well posedness.

The fact that we have obtained two equations of the form of one-way wave equations follows because the dynamics given by the two-way wave equation are equivalent to two scalar equations of the form:

$$z_t^{(1)} = z_x^{(1)}$$
$$z_t^{(2)} = -z_x^{(2)}.$$

In general, decoupling is a consequence of the symmetry of the matrix A. Indeed, consider the problem:

$$\mathbf{u}_t = A\mathbf{u}_x, \quad \mathbf{u}(x,0) = \mathbf{f}(x),$$

where $\mathbf{u}(x,t)$ and $\mathbf{f}(x)$ are vector-valued functions. Assume A is diagonalizable (e.g., when A is real symmetric), which implies the existence of a matrix T such that $T^{-1}AT$ is a diagonal matrix. Define $\Lambda = T^{-1}AT$ and define the transformation of variables: $\mathbf{w} = T^{-1}\mathbf{u}$ so that $\mathbf{w}$ satisfies:

$$\mathbf{w}_t = T^{-1}\mathbf{u}_t = T^{-1}A\mathbf{u}_x = (T^{-1}AT)\mathbf{w}_x = \Lambda\mathbf{w}_x$$

Since Λ is diagonal, this system is equivalent to a collection of scalar equations of the form of one-way wave equations. Let $\mathbf{w} = (w_1, w_2, \cdots, w_n)^T$. Then we get the energy for each w_i as

$$\int_{-\infty}^{\infty} |w_i(x,t)|^2 \, dx = \int_{-\infty}^{\infty} |w_i(x,0)|^2 \, dx.$$

On the other hand,

$$\langle \mathbf{u}, \mathbf{u} \rangle = \langle T\mathbf{w}, T\mathbf{w} \rangle = \mathbf{w}^* T^* T \mathbf{w},$$

where the superscript $*$ denotes the conjugate transpose. Let $\lambda_1, \cdots, \lambda_n$ be the eigenvalues of the matrix T^*T. Then λ_i are real positive numbers because T^*T is a Hermitian positive definite matrix ($\forall \mathbf{x} \neq \mathbf{0}, x^*T^*T\mathbf{x} = \langle T\mathbf{x}, T\mathbf{x} \rangle = \|\mathbf{x}\|^2 \rangle 0$). Let λ_n be the largest and λ_1 be the smallest eigenvalue. By the Courant-Fischer-Weyl min-max principle, we have

$$\lambda_1 \leq \frac{\mathbf{w}^* T^* T \mathbf{w}}{\mathbf{w}^* \mathbf{w}} \leq \lambda_n,$$

thus

$$\lambda_1 \langle \mathbf{w}, \mathbf{w} \rangle \leq \langle \mathbf{u}, \mathbf{u} \rangle \leq \lambda_n \langle \mathbf{w}, \mathbf{w} \rangle.$$

Finally, we have the strongly well posedness,

$$E(t) = \int_{-\infty}^{\infty} \langle \mathbf{u}, \mathbf{u} \rangle(x,t) \, dx \leq \lambda_n \int_{-\infty}^{\infty} \langle \mathbf{w}, \mathbf{w} \rangle(x,t) \, dx = \lambda_n \int_{-\infty}^{\infty} \langle \mathbf{w}, \mathbf{w} \rangle(x,0) \, dx$$

$$\leq \frac{\lambda_n}{\lambda_1} \int_{-\infty}^{\infty} \langle \mathbf{u}, \mathbf{u} \rangle(x,0) \, dx = \frac{\lambda_n}{\lambda_1} E(0).$$

In the examples given so far, the norm of the solution at any time can be bounded in terms of the norm of the initial condition. In order to illustrate how things can go wrong, we present now two examples in which this is no longer the case.

Example 5.4. *Consider the backward time heat equation:*

$$u_t = -u_{xx}, \quad u(x,0) = f(x).$$

In the Fourier space we have:

$$\hat{u}_t(\omega, t) = -(\mathbb{i}\,\omega)^2 \hat{u}(\omega, t) = \omega^2 \hat{u}(\omega, t)$$

and therefore $\hat{u}(\omega, t) = \hat{f}(\omega) e^{\omega^2 t}$, *which yields the solution in the physical space as:*

$$u(x,t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{\omega^2 t} \hat{f}(\omega) e^{\mathbb{i}\,\omega x} \, d\omega.$$

The above expression is well defined **only** *in the extreme case that* f *is analytic, which means that there is a positive finite number* ω_0 *such that*

$\hat{f}(\omega) = 0$ for all $\omega > \omega_0$. Nonetheless, in most physical problems the initial function is not analytic and $\hat{f}(\omega)$ does not have a compact support. In these situations, the integral $u(x,t)$ might not even exist for some values of t. When $\hat{f}(\omega)$ does not tend to zero fast enough to counteract the growth of the integrand as $\omega \to \infty$, then even for finite intervals of time the integral will not converge and thus there is no solution. We shall say that this problem is not well posed.

Physically, we can relate the forward and backward heat equations. In the former one the energy of the system is being lost or dissipated as time increases, and so the function $u(x,t)$ smooths out loosing information on the initial condition. The backward problem could be seen as the time-reversed problem, where energy is now being "pumped' into the system. The initial function, therefore, does not give us enough information to bound the energy at future times.

Example 5.5. *Consider now the equation:*

$$\begin{aligned} \frac{\partial}{\partial t}\begin{pmatrix} u \\ v \end{pmatrix} &= \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \frac{\partial}{\partial x}\begin{pmatrix} u \\ v \end{pmatrix}, \\ u(x,0) &= f_1(x) \\ v(x,0) &= f_2(x). \end{aligned}$$

The matrix is already in Jordan form and therefore it cannot be diagonalized, so we cannot represent the system in terms of scalar equations directly. Transforming the functions, we get in the Fourier space:

$$\frac{\partial}{\partial t}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \mathbb{i}\,\omega \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix}$$

We solve for $\hat{v}$ first, getting

$$\hat{v}(\omega,t) = e^{\mathbb{i}\,\omega t}\hat{f}_2(\omega)$$

and substituting this function in the differential equation for $\hat{u}$ we obtain:

$$\frac{\partial}{\partial t}\hat{u}(\omega,t) - \mathbb{i}\,\omega\hat{u}(\omega,t) = \mathbb{i}\,\omega e^{\mathbb{i}\,\omega t}\hat{f}_2(\omega)$$

or, equivalently:

$$\frac{\partial}{\partial t}\left[e^{-\mathbb{i}\,\omega t}\hat{u}(\omega,t)\right] = \mathbb{i}\,\omega\hat{f}_2(\omega)$$

Integrating this equation we finally get:

$$\begin{aligned} \hat{u}(\omega,t) &= \hat{f}_1(\omega)e^{\mathbb{i}\,\omega t} + \mathbb{i}\,\omega t\hat{f}_2(\omega)e^{\mathbb{i}\,\omega t} \\ \hat{v}(\omega,t) &= \hat{f}_2(\omega)e^{\mathbb{i}\,\omega t}, \end{aligned}$$

and so $\hat{u}$ contains a term that grows linearly on ω. We evaluate now the energy of this system, using Plancherel's identity:

$$\begin{aligned} E(t) &= \int_{-\infty}^{\infty} (|u(x,t)|^2 + |v(x,t)|^2)\, dx \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} (|\hat{u}(\omega,t)|^2 + |\hat{v}(\omega,t)|^2)\, d\omega \\ &\leq 2\frac{1}{2\pi} \int_{-\infty}^{\infty} (|\hat{f}_1(\omega)|^2 + |\hat{f}_2(\omega)|^2 + t^2|\omega \hat{f}_2(\omega)|^2)\, d\omega \\ &= 2\int_{-\infty}^{\infty} (|u(x,0)|^2 + |v(x,0)|^2)\, dx + 2t^2 \int_{-\infty}^{\infty} |\frac{\partial}{\partial x} v(x,0)|^2\, dx \end{aligned}$$

This implies that if we start with an initial function $f_2(x)$ with p continuous derivatives, we end up with $v(x,t)$ having $p-1$ continuous derivatives. This example illustrates a case, which is sort of "in between" the extreme cases given in the previous examples.

We are now ready to give the formal definitions of well posedness, keeping in mind the examples seen so far.

Definition 5.2. *The Sobolev p-norm of a function $f(x)$ of the vector $x = (x_1, \cdots, x_s)$, denoted by $\|f\|_p$ is defined by*

$$\|f\|_p = \left(\sum_{\alpha:|\alpha|\leq p} \int \left| \frac{\partial^\alpha f}{\partial x^\alpha}(x) \right|^2 \right)^{\frac{1}{2}}.$$

In particular, if x is a scalar, we have:

$$\|f\|_p^2 = \int_{-\infty}^{\infty} |f(x)|^2\, dx + \sum_{k=1}^{p} \left| \frac{\partial^k f}{\partial x^k}(x) \right|^2 dx.$$

and it should be noted that, in order for the above definition to make sense, we must assume some suitable conditions on f and its derivatives up to order p. We shall, however, work with functions that have p continuous derivatives, so their Sobolev p-norm will be well defined.

Definition 5.3. *If $f(x)$ is an n-vector valued function of $x = (x_1, \cdots, x_s)$ which has continuous derivatives up to order r and which has compact support, we write $f \in C_0^r$.*

Definition 5.4. *The initial value problem*

$$\begin{aligned} u_t(x,t) &= \mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) u(x,t), \\ u(x,0) &= f(x), \end{aligned}$$

is said to be weakly well posed if for every $f \in C_0^r$ and for each time $T_0 > 0$ there exists a unique solution $u(x,t)$ which is a classical solution and such that:

$$\|u(t)\| \leq K(t)\|f\|_p, \quad 0 \leq t \leq T_0, \tag{5.2}$$

for $p \leq r$ and $K(t) < Ce^{\alpha t}$ for some positive constants C and α. It is called strongly well posed (or simply well posed) if (5.2) holds with $p = 0$, in which case $\|f\|_p = \|f\|$ is just the usual L^2-norm.

The initial value problems of Examples 5.1, 5.2 and 5.3 are all strongly well posed. The problem given in Example 5.5 is weakly, but not strongly well posed, with $p = 1$. Finally the problem of Example 5.4 is not well posed, since there is no integer p for which (5.2) is satisfied.

So far we have considered classical solutions of the initial value problems. But the concept of "solution" may be broadened to include functions $u(x,t)$ that might fail differentiability at some points. We shall show now that the definition of well posedness allows natural extension for initial conditions that are not continuously differentiable and yet (5.2) holds.

Example 5.6. *Consider the heat equation with $a = 1$ in Example 5.2 and a classical solution will be a function $u(x,t)$ which is differentiable in time and twice differentiable in space, such that $u_t = u_{xx}$ with $u(x,0) = f(x)$. On the other hand, we have already evaluated the expression that any such solutions satisfy in terms of the initial condition $f(x)$, namely:*

$$u(x,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)e^{-\omega^2 t}e^{\mathbb{i}\,\omega x}\,d\omega. \tag{5.3}$$

where, as usual,

$$\hat{f}(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} f(x)e^{-\mathbb{i}\,\omega x}\,dx. \tag{5.4}$$

We may therefore define the solution of the heat equation to be a function $u(x,t)$ that satisfies (5.3). Notice that (5.4) is well defined even when $f(x)$ has a finite number of discontinuities, and so is (5.3), in which case $u(x,t)$ will no longer be a classical solution. As long as f has a bounded L^2-norm, we will have: $\|u(t)\| \leq \|f\|$ for all $t > 0$. We have thus constructed expression (5.3) which is satisfied by all classical solutions, but does not require differentiability. From here we were able to extend the concept of a solution of the heat equation.

More generally, assume that the linear, homogeneous partial differential equation:

$$u_t = \mathcal{P}(x,t,\partial\,/\,\partial\,x)u$$

is strongly well posed. (The case for weak well posedness follows in an analogous way, replacing the L^2-norm by the corresponding Sobolev p-norms

and the details are left to the reader.) Then for any initial condition on C_0^r there is a classical solution satisfying (5.2).

Let $\mathcal{H}$ be any space of functions such that C_0^r is dense in $\mathcal{H}$ in the L^2-norm (e.g., $C_0^\infty(\mathbb{R})$ is dense in $\mathcal{H} = L^2(\mathbb{R})$), and let $f(x)$ belong to $\mathcal{H}$. Then there is a scquence $\{f_n\}$ of functions with $f_n \in C_0^r$ for all n and such that $\|f - f_n\| \to 0$ as $n \to \infty$.

Since the problem is well posed, to each f_n, there corresponds a function $u(x,t)$ which solves $u_t = \mathcal{P}(x,t,\partial/\partial x)u$ with initia] condition $u(x,0) = f_n(x)$. It follows from (5.2) that the sequency is Cauchy in the L^2-norm, for each $t\rangle 0$. To prove this, let

$$v^{n,m} \equiv u_n(x,t) - u_m(x,t).$$

Since the differential equation is linear and homogeneous,

$$\mathcal{P}(x,t,\partial/\partial x)v^{n,m} = \mathcal{P}(x,t,\partial/\partial x)u_n - \mathcal{P}(x,t,\partial/\partial x)u_m$$

$$= \frac{\partial}{\partial t}u_n - \frac{\partial}{\partial t}u_m = (u_n - u_m)_t = v_t^{n,m},$$

so that $v^{n,m}$ is a solution of the equation with initial condition $f_n - f_m$.

$$f_n - f_m \in C_0^r \Rightarrow K(t)\|f_n - f_m\|,$$

which holds for each $t \geq 0$ and for any integers n, m. Therefore as $n, m \to \infty$ we get:

$$\lim_{n,m\to\infty} \|u_n - u_m\| \leq \lim_{n,m\to\infty} K(t)\|f_n - f_m\| = 0,$$

which proves the assertion.

It follows now that for each t, the limit of $\{u_n(t)\}$ in the L^2-norm exists, although it may not be continuously differentiable. Define $u(x,t)$ as the limit $\lim_{n\to\infty} u_n(x,t)$ for each t and now define $u(x,t)$ to be the solution of the equation with initial condition $u(x,0) = f(x)$. Since all limits are in the L^2-norm, it follows that u(x,t) satisfies (5.2):

$$\|u(t)\| \leq \lim_{n\to\infty} \|u_n(x,t)\| \leq K(t) \lim_{n\to\infty} \|f_n(x)\| = K(t)\|f\|.$$

Remark 5.3. *The concept of Sobolev spaces underlies this type of generalization. Actually, Sobolev spaces are constructed as the complction of the spaces C_0^r under a Sobolev p-norm. This way the concept of a solution of a well posed or weakly well posed problem can be generalized and the corresponding Sobolev space contains all possible initial functions for which that solution is well defined and such that* (5.2) *holds. The notions of weak and strong well posedness are related to the bound of the L^2-norm of the solution $u(x,t)$ in terms of the Sobolev p-norm or L^2-norm of the initial condition f.*

All examples seen so far involve constant coefficients. The following two examples consider cases of variable coefficients. Generally in this situation we cannot use Fourier transforms to get simpler problems in the Fourier space, so we have to resort to the so-called energy estimate in order to check well posedness.

Example 5.7. *Consider the scalar hyperbolic equation:*

$$u_t = a(x)u_x, \quad u(x,0) = f(x),$$

where $a \in C_0^r, f \in C \in C_0^r$ for $r \geq 2$. We define the energy of the system at time t by the L^2-norm of the solution:

$$E(t) = \int_{-\infty}^{\infty} |u(x,t)|^2 \, dx.$$

Differentiating $E(t)$ with respect to t, we get:

$$\frac{d}{dt}E(t) = 2\int_{-\infty}^{\infty} u(x,t)u_t(x,t)\, dx = 2\int_{-\infty}^{\infty} u(x,t)a(x)u_x(x,t)\, dx$$

$$= \int_{-\infty}^{\infty} a(x)\frac{\partial}{\partial x}[u(x,t)]^2 \, dx.$$

Integrating by parts and using the fact that $a(x)$ has a compact support we obtain:

$$\frac{d}{dt}E(t) = -\int_{-\infty}^{\infty} a_x(x)u^2(x,t)\, dx.$$

By assumption, a has $r \geq 2$continuous derivatives, and all of them have compact support. Since any continuous function on a compact set is bounded, we have

$$\sup_{x\in\mathbb{R}}\{a_x(x)\} \leq K < \infty,$$

for some constant K, yielding:

$$\frac{d}{dt}E(t) \leq K\int_{-\infty}^{\infty} u^2(x,t)\, dx = KE(t)$$

which yields the differential inequality $\frac{d}{dt}E - KE \leq 0$ or $\frac{d}{dt}[e^{-Kt}E(t)] \leq 0$. This implies that $e^{-Kt}E(t) \leq E(0)$ thus $E(t) \leq e^{Kt}E(0)$ for all $t \geq 0$. In terms of the L^2-norm, this inequality becomes:

$$\|u(x,t)\| \leq e^{\frac{K}{2}t}\|u(x,0)\|.$$

This proves that the problem is strongly well posed. Notice that we did not prove the existence of the classical solution, we simply assumed it, but methods to prove existence and uniqueness of solutions are beyond the scope of this text.

Example 5.8. *Consider the scalar partial differential equation:*

$$u_t = \frac{\partial}{\partial x}[a(x)u_x], \quad u(x,0) = f(x),$$

where $a \in C_0^r$ and $a(x) \geq 0$. We define the energy as the L^2-norm of the solution. In order to obtain in this case the appropriate estimate, we multiply the equation by $u(x,t)$ and integrate by parts to get:

$$\frac{d}{dt}E(t) = 2\int_{-\infty}^{\infty} u(x,t)u_t(x,t)\,dx = 2\int_{-\infty}^{\infty} u(x,t)\frac{\partial}{\partial x}[a(x)u_x(x,t)]\,dx$$

$$= -2\int_{-\infty}^{\infty} a(x)[u_x(x,t)]^2\,dx \leq 0,$$

where we have used that a has compact support and it is non-negative. Therefore the energy itself is not increasing, $E(t) \leq E(0)$ for all $t \geq 0$, which implies that the problem is well posed.

5.2 Lower Order Terms

In the previous section we discussed well posedness of linear, homogeneous partial differential equations. For more general problems it is often difficult to characterize well posedness. In this section we address the question of well posedness of a particular type of problems, relating then to a "simpler" problem. In some cases the properties of a differential equation are the same as those of a "perturbation" of the problem, and these are precisely the cases we shall focus on later in this section, but before we address this subject, we present an example where the situation is quite different.

Example 5.9. *Consider the problem studied in Example 5.5 of the previous section:*

$$\frac{\partial}{\partial t}\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\frac{\partial}{\partial x}\begin{pmatrix} u \\ v \end{pmatrix},$$

with initial functions $u_0(x)$ and $v_0(x)$. In the Fourier space we obtain:

$$\frac{\partial}{\partial t}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \mathbb{P}(\mathtt{i}\,\omega)\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix},$$

where $\mathbb{P}(\mathtt{i}\,\omega)$ denotes the matrix:

$$\mathbb{P}(\mathtt{i}\,\omega) = \mathtt{i}\,\omega\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathtt{i}\,\omega & \mathtt{i}\,\omega \\ 0 & \mathtt{i}\,\omega \end{pmatrix}.$$

This is not a diagonalizable matrix and, as we have scen, the proble in is not strongly well posed. Consider now the perturbed problem:

$$\frac{\partial}{\partial t}\begin{pmatrix} u^\varepsilon \\ v^\varepsilon \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\frac{\partial}{\partial x}\begin{pmatrix} u^\varepsilon \\ v^\varepsilon \end{pmatrix} + \varepsilon\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} u^\varepsilon \\ v^\varepsilon \end{pmatrix},$$

with same initial functions. The new equation involves a lower order term perturbation and we want to know how the solution of the new problem behaves. The ordinary differential equation in the Fourier space now becomes:

$$\frac{\partial}{\partial t}\begin{pmatrix}\hat{u}^\varepsilon\\ \hat{v}^\varepsilon\end{pmatrix} = \mathbb{P}^\varepsilon(\mathtt{i}\,\omega)\begin{pmatrix}\hat{u}^\varepsilon\\ \hat{v}^\varepsilon\end{pmatrix},$$

where $\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)$ *denotes the matrix:*

$$\mathbb{P}^\varepsilon(\mathtt{i}\,\omega) = \begin{pmatrix}\mathtt{i}\,\omega & \mathtt{i}\,\omega\\ \varepsilon & \mathtt{i}\,\omega\end{pmatrix}.$$

The solution in the Fourier space is given in terms of the exponential matrix $e^{\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)t}$ *as:*

$$\begin{pmatrix}\hat{u}^\varepsilon(\omega,t)\\ \hat{v}^\varepsilon(\omega,t)\end{pmatrix} = e^{\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)t}\begin{pmatrix}\hat{u}_0(\omega)\\ \hat{v}_0(\omega)\end{pmatrix}.$$

It is well known that the growth of the vector $\begin{pmatrix}\hat{u}^\varepsilon(\omega,t)\\ \hat{v}^\varepsilon(\omega,t)\end{pmatrix}$ *with respect to* ω *is related to the growth of eigenvalues of* $e^{\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)t}$*. Furthermore, if* $\lambda_1^\varepsilon(\omega)$ *and* $\lambda_2^\varepsilon(\omega)$ *denote the eigenvalues of* $\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)$*, then the eigenvalues of the exponential matrix* $e^{\mathbb{P}^\varepsilon(\mathtt{i}\,\omega)t}$ *are* $e^{\lambda_1^\varepsilon(\omega)t}$ *and* $e^{\lambda_2^\varepsilon(\omega)}$*. We have:*

$$\lambda_1^\varepsilon(\omega) = \mathtt{i}\,\omega + \sqrt{\varepsilon\omega}(\mathtt{i}+1)/\sqrt{2}, \quad \lambda_2^\varepsilon(\omega) = \mathtt{i}\,\omega - \sqrt{\varepsilon\omega}(\mathtt{i}+1)/\sqrt{2}.$$

If $\varepsilon = 0$*, then* $\left|e^{\lambda_i^\varepsilon(\omega)t}\right| = 1$ *for any* ω*. If* $\varepsilon \neq 0$*, then* $\left|e^{\lambda_i^\varepsilon(\omega)t}\right|$ *cannot be uniformly bounded on* ω*. Unboundedness of the eigenvalues implies that the perturbed problein is not well posed at all. This illustrates the fact that adding lower order terms may change the behavior of the solution.*

However, the situation is far from hopeless, and there is a lot we can say about lower order term perturbations, provided that the original problein is strongly well posed. In what follows we shall not be concerned with weakly well posed problems. Let $v(x,t)$ be the solution of the initial value problem:

$$v_t = \mathcal{P}\left(x,t,\frac{\partial}{\partial x}\right)v, \quad v(x,t_0) = g(x), \tag{5.5}$$

where t_0 is an arbitrary initial time and $v(x,t)$ is defined for $t \geq t_0$. We assume that for any such t_0 this problem is strongly well posed. This is equivalent to assume that there exists a solution operator $S(t,t_0)$ with the following properties:

1. For any $t \geq t_0$, $v(x,t) = S(t,t_0)v(x,t_0)$.
2. For any t_0, $S(t_0,t_0)$ is the identity operator.

3. For all t, t_1, t_0 s.t. $t_0 \leq t_1 \leq t$, we have $S(t, t_0) = S(t, t_1)S(t_1, t_0)$.

4. For all $t \geq t_0$, $|S(t, t_0)| \leq Ke^{a(t-t_0)}$ for some constants K and a.

Theorem 5.1. The Duhamel Principle. *Consider the non-homogeneous, initial value problem:*

$$u_t = \mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) u + F(x, t), \quad u(x, 0) = f(x), \tag{5.6}$$

where $\mathcal{P}$ is the polynomial on $\frac{\partial}{\partial x}$ of degreer r, $f \in C_0^r$, $F(x, t) \in C_0^r$ for all t. Then this problem has a unique solution given by:

$$u(x, t) = S(t, 0)f(x) + \int_0^t S(t, \tau)F(x, \tau)d\tau, \tag{5.7}$$

where S is the solution operator for the strongly well posed homogeneous problem (5.5) *defined above.*

Proof. First we prove uniqueness. Assume that u_1 and u_2 are both solutions to (5.6). Then their difference $v = u_1 - u_2$ satisfies (5.5) with initial condition $v(x, 0) = 0$. Since (5.5) is a well posed problem, it follows that $v(x, t) = 0$ and thus $u_1(x, t) = u_2(x, t)$, yielding uniqueness.

The rest of the proof follows by differentiating directly expression (5.7) and verifying that it satisfies (5.6). We define first the functions:

$$v(x, t) = S(t, 0)f(x), \quad w(x, t, \tau) = S(t, \tau)F(x, \tau).$$

It is clear that v solves the problem (5.5) with initial condition $v(x, 0) = f(x)$. For any fixed τ, $w(r, t, \tau)$ can be viewed as the solution of (5.5) with initial value $g(x) = F(x, \tau)$ starting at time $t_0 = \tau$. Therefore:

$$\frac{\partial}{\partial t} w(x, t, \tau) = \mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) w(x, t, \tau), \quad w(x, \tau, \tau) = F(x, \tau).$$

Differentiating now (5.7) with respect to time t and substituting $S(t, 0)f(x) = v$ and $S(t, \tau)F(x, \tau) = w$, we get:

$$\begin{aligned}
u_t &= v_t + w(x, t, t) + \int_0^t \frac{\partial}{\partial t} w(x, t, \tau)d\tau \\
&= v_t + S(t, t)F(x, t) + \int_0^t \frac{\partial}{\partial t} w(x, t, \tau)d\tau \\
&= P\left(x, t, \frac{\partial}{\partial x}\right) v + F(x, t) + \int_0^t P\left(x, t, \frac{\partial}{\partial x}\right) w(x, t, \tau)d\tau \\
&= P\left(x, t, \frac{\partial}{\partial x}\right) \left\{ v + \int_0^t w(x, t, \tau)d\tau \right\} + F(x, t) \\
&= P\left(x, t, \frac{\partial}{\partial x}\right) u + F(x, t)
\end{aligned}$$

For the initial condition, let $t = 0$ in (5.7) we get $u(x, 0) = f(x)$. □

Before stating the main result on well posedness of a lower order term perturbation of the problem (5.5), we state and prove a technical lemma.

Lemma 5.1. *Assume that the problem* (5.5) *is strongly (or weakly) well posed. Then the solution* $u(x,t)$ *of* (5.6) *satisfies: for any* $T>0$ *there exist constants* K *and* a *such that:*

$$\|u(t)\| \leq Ke^{at}\left\{\|f\|_p + \sup_{0\leq\tau\leq t}\|F(x,\tau)\|_p\left(\frac{1-e^{-at}}{a}\right)\right\}, \quad 0\leq t\leq T. \tag{5.8}$$

Remark 5.4. *If* (5.5) *is weakly well posed, then the solution operator* $S(t,t_0)$ *satisfies*

$$\|S(t,t_0)f\| \leq Ke^{a(t-t_0)}\|f\|_p,$$

for some integer p *and any initial function* f*. Notice that in the proof of Theorem 5.1 the condition* $\|S(t,t_0)f\| \leq Ke^{a(t-t_0)}\|f\|$ *was not used, thus its conclusion holds also in the case that the original unperturbed problem is only weakly well posed. If the problem is strongly well posed, we simply replace the Sobolev p-norms by the* L^2*-norm, both in* (5.8) *as well as in the proof that follows.*

Proof. The proof of the lemma follows as a straightforward application of Duhamel's principle. Taking norms in (2.19) we have:

$$\|u(t)\| \leq \|S(t,0)f(x)\| + \int_0^t \|S(t,\tau)F(\tau)\|d\tau,$$

and using the well posedness, we obtain:

$$\|u(t)\| \leq Ke^{at}\|f\|_p + \int_0^t Ke^{a(t-\tau)}\|F(\tau)\|_p d\tau$$

$$\leq Ke^{at}\|f\|_p + Ke^{at}\sup_{0\leq\tau\leq t}\|F(\tau)\|_p\int_0^t e^{-a\tau}d\tau.$$

□

Theorem 5.2. *Let the initial value problem* (5.5) *be strongly well posed and assume the perturbed problem:*

$$u_t = \mathcal{P}\left(x,t,\frac{\partial}{\partial x}\right)u + B(x,t)u, \quad u(x,0)=f(x), \tag{5.9}$$

has a solution, where $f\in C_0^\infty$*. Assume also that:*

$$\sup_{0\leq\tau\leq t}\|B(x,\tau)u(\tau)\| \leq b_0\|u(t)\|,$$

where b_0 *is a positive constant and* $t\geq 0$*. Then the problem* (5.9) *is also strongly well posed.*

Proof. We define the function:

$$y(x,t) = e^{-\beta t}u(x,t),$$

where $\beta \geq 0$ is a real number to be determined later on. Then y satisfies:

$$\frac{\partial y}{\partial t} = \beta y + \mathcal{P}(x,t,\partial\,/\,\partial\, x)y + B(x,t)y = (\mathcal{P}(x,t,\partial\,/\,\partial\, x) - \beta)y + B(x,t)y$$

Using the Duhamel principle with $F(x,t) = B(x,t)y(x,t)$ we get:

$$y(x,t) = \bar{S}(t,0)f(x) + \int_0^t \bar{S}(t,\tau)B(x,\tau)y(x,\tau)d\tau,$$

where $\bar{S}(t,t_0)$ is now the solution operator of the modified problem:

$$w_t = (\mathcal{P}(x,t,\partial\,/\,\partial\, x) - \beta)w.$$

By the assumption on strong well posedness of (5.5) it follows that $\bar{S}$ also satisfes: $|\bar{S}(t,\tau)| \leq Ke^{(\alpha-\beta)(t-\tau)}$ in the operator norm. Therefore:

$$\|y(t)\| \leq Ke^{(\alpha-\beta)t}\|f\| + K\int_0^t e^{(\alpha-\beta)(t-\tau)}b_0\|y(\tau)\|d\tau$$

$$\leq Ke^{(\alpha-\beta)t}\|f\| + Kb_0 \sup_{0\leq\tau\leq t}\|y(\tau)\|\frac{|1-e^{(\alpha-\beta)t}|}{|\alpha-\beta|}.$$

Now we choose β large enough so that $\alpha\langle\beta$ and also, given $T \geq 0$,

$$\gamma \equiv \sup_{0\leq t\leq T} Kb_0\frac{1-e^{(\alpha-\beta)t}}{\beta-\alpha} \leq \frac{1}{2},$$

then taking the supremum over $0 \leq t \leq T$, we have:

$$(1-\gamma)\sup_{0\leq t\leq T}\|y(t)\| \leq K\|f\|,$$

and therefore for all t with $0 \leq t \leq T$:

$$\|y(t)\| \leq \sup_{0\leq t\leq T}\|y(t)\| \leq \frac{K}{1-\gamma}\|f\|,$$

which, in terms of $u(x,t) = e^{\beta t}y(x,t)$ yields the inequality:

$$\|u(x,t)\| \leq \frac{K}{1-\gamma}e^{\beta t}\|f\|.$$

□

5.3 General results on constant coefficient problems

Well posedness is often difficult to establish for general problems. Now we restrict our attention to constant coefficient problems, for which a number of useful results are available. As usual, we shall present a series of examples in order to illustrate the formal results and their applications. We have chosen to summarize the main concepts and theorems for the specific case of hyperbolic equations.

By constant coefficient problems we mean an initial value problem of the form (5.1) where the operator $\mathcal{P}$ depends neither on x nor on t. That is, we consider the problem:

$$\begin{aligned} u_t &= \mathcal{P}(\partial / \partial x)u, \\ u(x,0) &= f(x). \end{aligned} \tag{5.10}$$

Here, $\mathcal{P}(\partial / \partial x)$ represents a polynomial whose coefficients are, in general, $n \times n$ matrices With constant entries, and $u(x,t) = (u_1(x,t), \cdots, u_n(x,t))^T$ is a function of $x = (x_1, \cdots, x_s)$ and time t. In the Fourier space, (5.10) becomes:

$$\begin{aligned} \hat{u}_t(\omega,t) &= \mathbb{P}(\mathbb{i}\,\omega)\hat{u}(\omega,t), \\ \hat{u}(\omega,0) &= \hat{f}(\omega). \end{aligned} \tag{5.11}$$

The matrix $\mathcal{P}(\mathbb{i}\,\omega)$ depends on $\omega = (\omega_1, \cdots, \omega_s)$. Therefore the above equations represent a system of ordinary differential equations with constant coefficients, and ω appears as a parameter. Denote by $\langle \omega, x \rangle$ the inner product $\sum_i \omega_i x_i$, then the solution is given by

$$\hat{u}(\omega,t) = e^{\mathbb{P}(\mathbb{i}\,\omega)t}\hat{f}(\omega),$$

thus

$$u(x,t) = \frac{1}{(2\pi)^s}\int_{\mathbb{R}^s} e^{\mathbb{P}(\mathbb{i}\,\omega)t}\hat{f}(\omega)e^{i\langle\omega,x\rangle}d\omega.$$

Therefore in order to establish conditions for well posedness of the problem (5.10) we only need to study the properties of $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$.

Definition 5.5. *The matrix $\mathbb{P}(\mathbb{i}\,\omega)$ is called the symbol of the partial differential equation it is associated with.*

For a vector-valued function $u(x,t) = (u_1(x,t), \cdots, u_n(x,t))^T$ we denote by $\|u\|$ the L^2-norm:

$$\|u\|^2 = \sum_{k=1}^{n}\int_{\mathbb{R}^s} |u_k(x,t)|^2 dx.$$

The problem of well posedness reduces to the study of the growth of the matrix $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ as a function of ω. By the Plancherel's identity (4.10), we get

$$\|u(x,t)\|^2 = \frac{1}{(2\pi)^s}\|\hat{u}(\omega,t)\|^2 = \frac{1}{(2\pi)^s}\int_{\mathbb{R}^s}\left|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\hat{f}(\omega)\right|^2 d\omega$$

so that

$$\|u(x,t)\|^2 = \frac{1}{(2\pi)^s}\|\hat{u}(\omega,t)\|^2 \leq \frac{1}{(2\pi)^s}\int_{\mathbb{R}^s}\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\|^2\left|\hat{f}(\omega)\right|^2 d\omega,$$

where $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\|$ is the matrix norm of the symbol. If this norm is uniformly bounded on ω, say $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| \leq e^{\alpha t}$ for some constant α and all values of ω, then $\|u\| \leq e^{\alpha t}\|f\|$ which guarantees strong well posedness. We first work on some examples and later focus on more general results.

Example 5.10. *Let* $u = \begin{pmatrix} u_1(x,t) \\ u_2(x,t) \end{pmatrix}$ *and consider*

$$\frac{\partial u}{\partial t} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\frac{\partial u}{\partial x}.$$

The symbol is given by

$$\mathbb{P}(\mathbb{i}\,\omega) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}(\mathbb{i}\,\omega).$$

The eigenvalues of the symbol are $\pm\,\mathbb{i}\,\omega$ *with two eigenvectors* $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ *and* $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$*, thus it can be diagonalized by an unitary matrix* $\mathbb{P}(\mathbb{i}\,\omega) = T\begin{pmatrix} \mathbb{i}\,\omega & 0 \\ 0 & -\mathbb{i}\,\omega \end{pmatrix}T^{-1}$ *with* $T = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$*. Therefore,* $e^{\mathbb{P}(\mathbb{i}\,\omega)t} = T\begin{pmatrix} e^{\mathbb{i}\,\omega t} & 0 \\ 0 & e^{-\mathbb{i}\,\omega t} \end{pmatrix}T^{-1}$*. Since* T *is an unitary matrix* ($TT^* = I$)*, the singular values of* $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ *are equal to* 1*, thus* $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| = 1$*. Therefore the IVP for this equation is strongly wellposed.*

Example 5.11. *Let* $u = \begin{pmatrix} u_1(x,y,t) \\ u_2(x,y,t) \end{pmatrix}$ *and consider*

$$\frac{\partial u}{\partial t} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\frac{\partial u}{\partial x} + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\frac{\partial u}{\partial y}.$$

Then the symbol is given by

$$\mathbb{P}(\mathbb{i}\,\omega) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\mathbb{i}\,\omega_1 + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbb{i}\,\omega_2 = \mathbb{i}\begin{pmatrix} \omega_1 & \omega_2 \\ \omega_2 & \omega_1 \end{pmatrix}.$$

The eigenvalues are $\mathbb{i}(\omega_1 \mp \omega_2)$ *with eigenvectors* $\begin{pmatrix}1\\1\end{pmatrix}$ *and* $\begin{pmatrix}1\\-1\end{pmatrix}$. *Thus it can be diagonalized by an unitary matrix* $\mathbb{P}(\mathbb{i}\,\omega) = T\,\mathbb{i}\begin{pmatrix}\omega_1-\omega_2 & 0\\ 0 & \omega_1+\omega_2\end{pmatrix}T^{-1}$ *with* $T = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$. *Therefore,* $e^{\mathbb{P}(\mathbb{i}\,\omega)t} = T\begin{pmatrix}e^{\mathbb{i}(\omega_1-\omega_2)t} & 0\\ 0 & e^{\mathbb{i}(\omega_1+\omega_2)t}\end{pmatrix}T^{-1}$. *Since* T *is an unitary matrix* ($TT^* = I$), *the singular values of* $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ *are equal to* 1, *thus* $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| = 1$. *Therefore the IVP for this equation is strongly wellposed.*

Example 5.12. *Consider* $u = \begin{pmatrix}u_1(x,y,t)\\ u_2(x,y,t)\end{pmatrix}$ *and*

$$u_t = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}u_{xx} + \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}u_{xy} + \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}u_{yy}.$$

The symbol is

$$\mathbb{P}(\mathbb{i}\,\omega) = \begin{pmatrix}-\omega_1^2-\omega_2^2 & -\omega_1\omega_2\\ -\omega_1\omega_2 & -\omega_1^2-\omega_2^2\end{pmatrix}$$

with eigenvalues

$$\lambda = -\omega_1^2 - \omega_2^2 \pm \omega_1\omega_2 \le 0.$$

The symmetry of the symbol implies that it is diagonalizable and we get

$$e^{\mathbb{P}(\mathbb{i}\,\omega)t} = T\begin{pmatrix}e^{(-\omega_1^2-\omega_2^2-\omega_1\omega_2)t} & 0\\ 0 & e^{(-\omega_1^2-\omega_2^2+\omega_1\omega_2)t}\end{pmatrix}T^{-1}.$$

Since $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ *is a negative semi-definite matrix, the singular values of* $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ *are equal to its eigenvalues, thus* $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| = e^{(-\omega_1^2-\omega_2^2+|\omega_1\omega_2|)t} \le e^{0*t} = 1$. *Therefore the IVP for this equation is strongly wellposed.*

Theorem 5.3. *The initial value problem* (5.10) *is weakly (strongly) well posed if and only if there exist constants* K, α *and an integer* p *independent of* ω, *such that*

$$\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| \le K(\|\omega\|^p + 1)e^{\alpha t}.$$

If $p = 0$, *then the problem is strongly well posed.*

Proof. Suppose that $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\right\| \le K(\|\omega\|^p+1)e^{\alpha t}$ holds. Using the fact that the Sobolev p-norm $\|f\|_p^2$ is equivalent to $\int_{\mathbb{R}^s}(\|\omega\|^p+1)^2|\hat{f}(\omega)|^2 d\omega$ and the Plancherel's identity, we get:

$$\|u(x,t)\| = \frac{1}{(2\pi)^{\frac{s}{2}}}\|\hat{u}(\omega,t)\| = \frac{1}{(2\pi)^{\frac{s}{2}}}\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)t}\hat{f}(\omega)\right\|$$

$$\leq \frac{1}{(2\pi)^{\frac{s}{2}}} K e^{\alpha t} \|(\|\omega\|^p + 1)\hat{f}(\omega)\| \leq K' e^{\alpha t} \|f\|_p.$$

□

Denote by A^* the conjugate transpose of the matrix A. We recall now that if A and B are two Hermitian matrices (that is, $A = A^*$ and $B = B^*$), we say that $A \leq B$ if $A - B$ is a negative definite matrix or, equivalently, if the eigenvalues of $A - B$ are all non-positive.

Theorem 5.4. *Suppose that there exists a constant α such that for all values of ω we have:*

$$\mathbb{P}(\mathrm{i}\,\omega) + \mathbb{P}^*(\mathrm{i}\,\omega) \leq \alpha I, \tag{5.12}$$

then $\left\| e^{\mathbb{P}(\mathrm{i}\,\omega)t} \right\| \leq e^{\alpha t}$ for all values of ω, thus the initial value problem (5.10) *is strongly well posed.*

Remark 5.5. *We are not interested in bounding α, only in the fact that α does not depend on ω. Also note that condition* (5.12) *is a sufficient although not necessary condition for well posedness, as will be made clearer later on.*

Proof. Let $\phi(\omega, t)$ denote the inner product

$$\phi(\omega, t) = \langle \hat{u}(\omega, t), \hat{u}(\omega, t) \rangle = \sum_{i=1}^{n} |\hat{u}_k(\omega, t)|^2,$$

so that the energy is given by

$$E(t) = \frac{1}{(2\pi)^s} \int_{\mathbb{R}^s} \phi(\omega, t) d\omega.$$

We now have

$$\frac{\partial\, \phi}{\partial\, t}(\omega, t) = \langle \hat{u}_t(\omega, t), \hat{u}(\omega, t) \rangle + \langle \hat{u}(\omega, t), \hat{u}_t(\omega, t) \rangle$$

where $\hat{u}_t(\omega, t) = \mathbb{P}(\mathrm{i}\,\omega)\hat{u}(\omega, t)$, thus

$$\begin{aligned} \frac{\partial\, \phi}{\partial\, t}(\omega, t) &= \langle \mathbb{P}(\mathrm{i}\,\omega)\hat{u}, \hat{u} \rangle + \langle \hat{u}, \mathbb{P}(\mathrm{i}\,\omega)\hat{u} \rangle \\ &= \langle [\mathbb{P}(\mathrm{i}\,\omega) + \mathbb{P}^*(\mathrm{i}\,\omega)]\hat{u}, \hat{u} \rangle \\ &\leq \alpha \langle \hat{u}, \hat{u} \rangle = \alpha \phi(\omega, t) \end{aligned}$$

Since α is independent of ω, we get $\phi(\omega, t) \leq e^{\alpha t}\phi(\omega, 0)$ for all ω, thus

$$E(t) = \frac{1}{(2\pi)^s} \int_{\mathbb{R}^s} \phi(\omega, t) d\omega \leq \frac{1}{(2\pi)^s} \int_{\mathbb{R}^s} e^{\alpha t} \phi(\omega, 0) d\omega = e^{\alpha t} E(0).$$

□

The parameter α is associated with the growth of the solution in time. Therefore negative value of or implies that the energy is decreasing, that is, the system in this case is *dissipative*.

Example 5.13. *Let x be a scalar and consider $u = \begin{pmatrix} u_1(x,t) \\ u_2(x,t) \end{pmatrix}$ in the equation*

$$u_t = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} u_x + \begin{pmatrix} 2 & 1 \\ 7 & \pi \end{pmatrix} u.$$

The symbol is given by

$$\mathbb{P}(\mathbb{i}\,\omega) = \mathbb{i}\,\omega \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 1 \\ 7 & \pi \end{pmatrix} = \begin{pmatrix} 2 & 1+\mathbb{i}\,\omega \\ 7+\mathbb{i}\,\omega & \pi \end{pmatrix},$$

and we have

$$\mathbb{P}(\mathbb{i}\,\omega) + \mathbb{P}^*(\mathbb{i}\,\omega) = \begin{pmatrix} 4 & 8 \\ 8 & 2\pi \end{pmatrix},$$

which is a constant matrix thus (5.12) *is satisfied for large α.*

Example 5.14. *Let A and B be real symmetric matrices (therefore Hermitian), with $A \leq 0$ and $B \geq 0$ and consider the problem*

$$u_t = Au_{xxxx} + Bu_{yy},$$

then $\mathbb{P} = A\omega_1^4 - B\omega_2^2$, which is a real, symmetric matrix. Therefore $\mathbb{P}(\mathbb{i}\,\omega) + \mathbb{P}^*(\mathbb{i}\,\omega) = 2\mathbb{P}(\mathbb{i}\,\omega) \leq 0$, because $A \leq 0$ and $B \geq 0$, which gives the well posedness without finding the eigenvalues of $\mathbb{P}(\mathbb{i}\,\omega)$ or $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$.

Theorem 5.5. *The initial value problem* (5.10) *is strongly well posed if and only if there exist a Hermitian matrix $H(\omega) > 0$ and a constant α such that*

$$\|H(\omega)\| \leq K, \quad \|H^{-1}(\omega)\| \leq K$$

for some $K > 0$ and

$$H(\omega)\mathbb{P}(\mathbb{i}\,\omega) + \mathbb{P}^*(\mathbb{i}\,\omega)H(\omega) \leq \alpha H(\omega). \tag{5.13}$$

Remark 5.6. *Theorem 5.4 is a special case of Theorem 5.5, taking $H(\omega)$ to be the identity. Theorem 5.5 is much stronger, since it states necessary and sufficient conditions for strong well posedness.*

Proof. To show the sufficiency, assume first that (5.13) holds and define an energy $E(t)$ according to the inner product: $\phi(\omega,t) = \langle \hat{u}(\omega,t), H(\omega)\hat{u}(\omega,t)\rangle$ so that:

$$E(t) = \frac{1}{(2\pi)^s} \int_{\mathbb{R}^s} \phi(\omega,t) d\omega.$$

Then we have

$$\begin{aligned}\frac{\partial \phi}{\partial t}(\omega,t) &= \langle \mathbb{P}(\mathbb{i}\,\omega)\hat{u}, H(\omega)\hat{u}\rangle + \langle \hat{u}, H(\omega)\mathbb{P}(\mathbb{i}\,\omega)\hat{u}\rangle \\ &= \langle [H(\omega)\mathbb{P}(\mathbb{i}\,\omega) + \mathbb{P}^*(\mathbb{i}\,\omega)H(\omega)]\hat{u}, \hat{u}\rangle \\ &\leq \alpha\langle H(\omega)\hat{u}, \hat{u}\rangle = \alpha\langle \hat{u}, H(\omega)\hat{u}\rangle = \alpha\phi(\omega,t).\end{aligned}$$

Since $\|H(\omega)\|$ and $\|H^{-1}(\omega)\|$ are the largest and smallest singular values of $H(\omega)$, and eigenvalues of $H(\omega)$ are also its singular values (because it is positive definite), we have

$$\frac{1}{K} \leq \lambda(H(\omega)) \leq K, \quad \forall\omega,$$

where $\lambda(H(\omega))$ denotes any eigenvalue of $H(\omega)$. By the Courant-Fischer-Weyl min-max principle, we have

$$\frac{1}{K}\langle \hat{u}, \hat{u}\rangle \leq \langle \hat{u}, H(\omega)\hat{u}\rangle = \phi(\omega,t) \leq e^{\alpha t}\phi(\omega,0).$$

Therefore, using the Courant-Fischer-Weyl min-max principle one more time, we have

$$\frac{1}{K}|\hat{u}(\omega,t)|^2 \leq e^{\alpha t}\langle \hat{u}(\omega,0), H(\omega)\hat{u}(\omega,0)\rangle \leq Ke^{\alpha t}|\hat{u}(\omega,0)|^2.$$

Integrating now with respect to ω yields well posedness of the problem.

The proof of the necessity is much more technical. See Theorem 2.3.2 in [6]. □

Theorem 5.5 is not very useful in practical applications, since it does not provide the construction of the matrix $H(\omega)$, which nakes it hard to prove conditions (5.13). The following two results are more useful in practice.

Theorem 5.6. *If $\mathbb{P}(\mathbb{i}\,\omega)$ is a normal matrix (a normal matrix means that $\mathbb{P}(\mathbb{i}\,\omega)\mathbb{P}^*(\mathbb{i}\,\omega) = \mathbb{P}^*(\mathbb{i}\,\omega)\mathbb{P}(\mathbb{i}\,\omega)$), then the initial value problem* (5.10) *is well posed if and only if there exists a constant α such that*

$$Re[\lambda_j(\omega)] \leq \alpha,$$

for all eigenvalues $\lambda_j(\omega)$ of $\mathbb{P}(\mathbb{i}\,\omega)$.

Proof. Normal matrices are unitarily diagonalizable, thus there is an unitary matrix T such that $\mathbb{P}(\mathbb{i}\,\omega) = T\Lambda T^*$ where Λ is diagonal with eigenvalues of $\mathbb{P}(\mathbb{i}\,\omega)$ as diagonal entries. Therefore $e^{\mathbb{P}(\mathbb{i}\,\omega)} = Te^{\Lambda t}T^*$ where $e^{\Lambda t}$ is a diagonal matrix with diagonal entries $e^{\lambda_j t}$. Let $\lambda_j = a_j + \mathbb{i}\,b_j$ with $a_j, b_j \in \mathbb{R}$. Recall that $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)}\right\|$ is the largest singular value of the matrix $e^{\mathbb{P}(\mathbb{i}\,\omega)}$, which is the square root of the eigenvalue (with largest magnitude) of $e^{\mathbb{P}(\mathbb{i}\,\omega)}(e^{\mathbb{P}(\mathbb{i}\,\omega)})^*$. We have $e^{\mathbb{P}(\mathbb{i}\,\omega)}(e^{\mathbb{P}(\mathbb{i}\,\omega)})^* = Te^{(\Lambda+\Lambda^*)t}T^*$ where $\Lambda + \Lambda^*$ is a diagonal matrix with diagonal entries $2a_j$. Thus $\left\|e^{\mathbb{P}(\mathbb{i}\,\omega)}\right\| = \max_j |a_j t| \leq \alpha t$. □

Theorem 5.7. *The initial value problem* (5.10) *is weakly well posed if and only if there exists a constant* α *such that* $Re[\lambda_j(\omega)] \leq \alpha$ *for all eigenvalues* $\lambda_j(\omega)$ *of* $\mathbb{P}(\mathtt{i}\,\omega)$.

See [6] for the proof.

Example 5.15. *Consider the normalized Schrödinger equation:*

$$u_t = \mathtt{i}\, u_{xx}.$$

Here we have $\mathbb{P}(\mathtt{i}\,\omega) = \mathtt{i}(\mathtt{i}\,\omega)^2 = -\mathtt{i}\,\omega^2$, *therefore:* $\mathbb{P}(\mathtt{i}\,\omega) + \mathbb{P}^*(\mathtt{i}\,\omega) = 0$ *which yields strong well posedness. As can be seen from the proof of Theorem 5.4, it turns out that* $\mathbb{P}(\mathtt{i}\,\omega) + \mathbb{P}^*(\mathtt{i}\,\omega)$ *is related to the time derivative of the energy:*

$$E(t) = \int_{\mathbb{R}} u^2(x,t)\, dx,$$

and in this system $E(t) = E(0)$ *remains constant, that is, it represents a conservative system.*

Example 5.16. *Let* A, B *be two matrices such that* $A = -A^*$ *and* $B = B^*$, *and let* C *be any matrix. Consider the equation:*

$$u_t = Au_{xx} + Bu_x + Cu.$$

Then we have

$$\mathbb{P}(\mathtt{i}\,\omega) = -A\omega^2 + \mathtt{i}\, B\omega + C,$$

$$\mathbb{P}(\mathtt{i}\,\omega) + \mathbb{P}^*(\mathtt{i}\,\omega) = -\omega^2(A + A^*) + \mathtt{i}\,\omega(B - B^*) + C + C^* = C + C^*$$

which is independent of ω *and so there is a constant* α *such that* $\mathbb{P}(\mathtt{i}\,\omega) + \mathbb{P}^*(\mathtt{i}\,\omega) \leq \alpha I$. *Notice that the growth in time depends on the matrix* C. *If, for instance,* $C = 0$, *then the energy remains constant.*

Example 5.17. *Consider now the scalar equation:*

$$u_t = -u_{xxxx} - u_{xx} + u_x + u,$$

where $u(x,t)$ *is a real valued function.* $\mathbb{P}(\mathtt{i}\,\omega)$ *is therefore just a polynomial:*

$$\mathbb{P}(\mathtt{i}\,\omega) = -\omega^4 + \omega^2 + \mathtt{i}\,\omega + 1,$$

and it satisfies the inequality:

$$\mathbb{P}(\mathtt{i}\,\omega) + \mathbb{P}^*(\mathtt{i}\,\omega) = -2\omega^4 + 2\omega^2 + 2 \leq 4,$$

so the problem is well posed.

5.4 Hyperbolic equations

In this section we shall focus on *hyperbolic equations* with constant coefficients given in its general form by the expression:

$$\begin{aligned} u_t(x,t) &= \sum_{j=1}^{s} A_j \frac{\partial u}{\partial x_j}(x,t) \\ u(x,0) &= u_0(x), \end{aligned} \tag{5.14}$$

where $x = (x_1, ..., x_s), u(x,t) = (u_1(x,t), \cdots, u_n(t,t))^T$, and each A_j is an $n \times n$ real matrix.

Definition 5.6. *Equation* (5.14) *is said to be weakly hyperbolic if the symbol*

$$\mathbb{P}(\mathbb{i}\,\omega) = \mathbb{i} \sum_{j=1}^{s} A_j \omega_j$$

has purely imaginary eigenvalues. It is called strongly hyperbolic if its has purely imaginary eigenvalues and if there exists a matrix $T(\omega)$ such that

- *There exists a constant K such that for all values of ω, $\|T(\omega)\| \leq K$, and $\|T^{-1}(\omega)\| \leq K$,*
- *$T(\omega)$ diagonalizes $\mathbb{P}(\mathbb{i}\,\omega)$, that is $T(\omega)^{-1}\mathbb{P}(\mathbb{i}\,\omega)T(\omega) = \Lambda(\omega)$ is a diagonal matrix with purely imaginary eigenvalues.*

Theorem 5.8. *A weakly (strongly) hyperbolic equation is weakly (respectively, strongly) well posed.*

Proof. If (5.14) is weakly hyperbolic, then by definition $\mathbb{P}(\mathbb{i}\,\omega)$ has purely imaginary eigenvalues, thus is weakly well posed, by Theorem 5.7.

Assume now that it is strongly hyperbolic. Since $\Lambda(\omega) + \Lambda^*(\omega) = 0$, we have

$$\begin{aligned} T(\omega)^{-1}\mathbb{P}(\mathbb{i}\,\omega)T(\omega) + [T(\omega)^{-1}\mathbb{P}(\mathbb{i}\,\omega)T(\omega)]^* &= 0, \\ T(\omega)^{-1}\mathbb{P}(\mathbb{i}\,\omega)T(\omega) + T(\omega)^*\mathbb{P}^*(\mathbb{i}\,\omega)[T^{-1}]^*(\omega) &= 0, \\ [T^{-1}]^*(\omega)T(\omega)^{-1}\mathbb{P}(\mathbb{i}\,\omega) + \mathbb{P}^*(\mathbb{i}\,\omega)[T^{-1}]^*(\omega)T^{-1}(\omega) &= 0. \end{aligned}$$

Let $H(\omega) = [T^{-1}]^*(\omega)T(\omega)^{-1}$, then by Theorem 5.5 we get strong well posedness. □

Example 5.18. *We consider here the Euler equation for gas dynamics. Calling ρ the density, u the velocity and p the pressure, the laws of conservation of mass, flow and energy yield:*

$$\begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_t = - \begin{pmatrix} u & \rho & 0 \\ 0 & u & \rho^{-1} \\ 0 & \rho c^2 & u \end{pmatrix} \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_x$$

where $c^2 = \gamma p/\rho$ *and* $\gamma = 1.4$. *The above is not a constant coefficient equation, since the matrix depends on the state variables. Linearizing the problems around some fixed state* (ρ_0, u_0, p_0) *we get the problem:*

$$w_t = Aw_x$$

Where $w = (\rho_0, u_0, p_0)^T$ *and* A *is the constant matrix:*

$$A = -\begin{pmatrix} u_0 & \rho_0 & 0 \\ 0 & u_0 & \rho_0^{-1} \\ 0 & \rho_0 c_0^2 & u_0 \end{pmatrix}$$

For this problem $\mathbb{P}(\mathbb{i}\,\omega) = \mathbb{i}\,\omega A$. *The eigenvalues of* $-A$ *are* u_0, $u_0 + c_0$ *and* $u_0 - c_0$. *So it has three distinct eigenvalues when* c_0 *is a nonzero real number and thus it can be diagonalized, yielding strong hyperbolicity. If either* p_0 *or* ρ_0 *is negative then* c_0 *is purely imaginary, which results in illposedness.*

Example 5.19. *Consider the system:*

$$\begin{pmatrix} u \\ v \end{pmatrix}_t = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x,$$

then the symbol

$$\mathbb{P}(\mathbb{i}\,\omega) = \mathbb{i}\,\omega \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

is already in Jordan form. Although its eigenvalues are purely imaginary, it cannot be diagonalized. This is therefore a weakly hyperbolic equation. This system is indeed only weakly but not strongly well posed as we discussed in Example 5.5.

Example 5.20. *Consider now the equations*

$$\begin{pmatrix} u \\ v \end{pmatrix}_t = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x.$$

The symbol

$$\mathbb{P}(\mathbb{i}\,\omega) = \begin{pmatrix} 0 & \mathbb{i}\,\omega \\ -\mathbb{i}\,\omega & 0 \end{pmatrix}$$

is a Hermitian matrix with real eigenvalues. Thus it is neither weakly nor strongly hyperbolic. In fact, this problem is not well posed at all, since the eigenvalues $\pm\omega$ *are not bounded. This gives rise to an exponential growth of the matrix* $e^{\mathbb{P}(\mathbb{i}\,\omega)t}$ *with respect to* ω. *The equations above can be rewritten as the Laplace equation:*

$$u_t = v_x, v_t = -u_x$$

and therefore $u_{tt} = v_{xt} = v_{tx} = -u_{xx}$, *which yields* $u_{tt} + u_{xx} = 0$. *So it is not well posed for the initial value problem but well posed for boundary value problem in a bounded space-time domain.*

Lemma 5.2. *If*

$$u_t(x,t) = \sum_{j=1}^{s} A_j \frac{\partial u}{\partial x_j}(x,t)$$
$$u(x,0) = u_0(x),$$

is weakly (strongly) well posed, then so is the system

$$u_t(x,t) = -\sum_{j=1}^{s} A_j \frac{\partial u}{\partial x_j}(x,t)$$
$$u(x,0) = u_0(x).$$

The proof of the lemma is straightforward, since the conditions on the eigenvalues of $\mathbb{P}(\mathtt{i}\,\omega)$ (which is a purely imaginary matrix in this case) for well posedness do not depend on the sign of the matrices A_j. This is equivalent to time reversal, which means that in hyperbolic systems knowledge of the "present" ($t = 0$) gives as much information about the "future" ($t < 0$) as it gives about the "past" ($t < 0$). Recall that this is not the situation in all cases, e.g., heat equations, for which the time reversal leads to a blow-up of the solution.

Definition 5.7. *We call system* (5.14) *is strictly hyperbolic if all eigenvalues of the symbol* $\mathbb{P}(\mathtt{i}\,\omega)$ *are purely imaginary and are distinct from each other.*

Definition 5.8. *We call system* (5.14) *symmetric hyperbolic if there exists a constant real matrix* S *such that* $B_j = S^{-1}A_jS$ *are real symmetric matrices, for all* j*.*

Theorem 5.9. *If the system* (5.14) *symmetric hyperbolic, then it is also strongly well posed*

Proof. If the system is symmetric hyperbolic, then $\sum_j B_j\omega_j$ is real symmetric thus has real eigenvalues, thus $S^{-1}P(\mathtt{i}\,\omega)S$ has purely imaginary eigenvalues, and so does $P(\mathtt{i}\,\omega)$. So symmetry at least implies the weak hyperbolicity. To see why it is also strongly well posed, consider the matrix $S^{-1}P(\mathtt{i}\,\omega)S$ which satisfes

$$S^{-1}P(\mathtt{i}\,\omega)S + [S^{-1}P(\mathtt{i}\,\omega)S]^* = 0,$$

thus

$$S^{-1}P(\mathtt{i}\,\omega)S + S^*P^*(\mathtt{i}\,\omega)(S^{-1})^* = 0,$$
$$[S^*]^{-1}S^{-1}P(\mathtt{i}\,\omega) + P^*(\mathtt{i}\,\omega)[S^{-1}]^*S^{-1} = 0.$$

Notice that $[S^*]^{-1} = [S^{-1}]^*$, thus we find a constant matrix $H = [S^*]^{-1}S^{-1}$ such that

$$HP(\mathtt{i}\,\omega) + P^*(\mathtt{i}\,\omega)H = 0.$$

□

If the equation (5.14) is symmetric hyperbolic, then it is also strongly hyperbolic, see [4, 6]. Notice, though that a strictly hyperbolic equation might not be symmetric hyperbolic and vice versa

Suppose now that equation (5.14) is symmetric hyperbolic. It is not straightforward to calculate the matrix S that symmetrizes all matrices A_j. We give now a procedure shown how this matrix can be constructed.

Method for Symmetrizing

Let the matrices $A_1, ..., A_s$, be given and suppose that there exists the matrix S such that all $B_j = S^{-1}A_jS$ are symmetric. Then for each integer j, B_j can in turn be diagonalized, although in general there will be a different transformation for each B_j. Let U_1 be the orthogonal, unitary transformation that diagonalizes B_1, that is,

$$U_1^* B_1 U_1 = \Lambda_1,$$

with Λ_1 diagonal, and define C_j, for $j \geq 2$ by:

$$C_j = U_1^* B_j U_1,$$

Since B_j is a real symmetric matrix, then $B_j = B_j^*$ and therefore $C_j^* = U_1^* B_j^* U_1 = U_1^* B_j U_1 = C_j$ is also symmetric for each j. From this observation we can now conclude that if the matrices $A_1, \cdots, A_s$ can all be symmetrized by one matrix S, then one matrix $\bar{S} = SU_1$ can diagonalize A_1 while symmetrizing A_j for $j = 2, \cdots, s$. Let us assume now that we do not know explicitly S and U_1 and let $\bar{S}$ be a matrix such that

$$\bar{S}^{-1} A_1 \bar{S} = \Lambda_1,$$

and note that $\bar{S}$ is determined from the above requirement up to multiplication by a diagonal matrix. Once $\bar{S}$ is chosen, one evaluates the matrices:

$$\bar{A}_j = \bar{S}^{-1} A_j \bar{S}$$

and now it remains to check whether all these matrices $\bar{A}$ can be symmetrized by a single diagonal matrix D with diagonal entries $d_1, \cdots, n$. Try to find the scalars $d_1, \cdots, n$, such that $D^{-1}\bar{A}_j D$, $j = 2, ..., s$ are all symmetric. If there exists such a matrix D, then it is possible to symmetrize all A_j with a single matrix S and the problem is symmetric hyperbolic. If such a matrix D does not exist, then the problem is not symmetric hyperbolic.

Example 5.21. *Consider one form of the two dimensional Euler equation for gas dynamics on a plane, and denote now the velocity components by u*

and v:

$$\begin{pmatrix}\rho\\u\\v\\p\end{pmatrix}_t = -\begin{pmatrix}u & \rho & 0 & 0\\0 & u & 0 & \rho^{-1}\\0 & 0 & u & 0\\0 & \rho c^2 & 0 & u\end{pmatrix}\begin{pmatrix}\rho\\u\\v\\p\end{pmatrix}_x - \begin{pmatrix}v & 0 & \rho & 0\\0 & v & 0 & 0\\0 & 0 & v & \rho^{-1}\\0 & 0 & \rho c^2 & v\end{pmatrix}\begin{pmatrix}\rho\\u\\v\\p\end{pmatrix}_y$$

For the linearized problem, where we evaluate the matrices at some fixed value of the state variables, say (ρ_0, u_0, v_0, p_0), we have:

$$\mathbb{P}(\mathbb{i}\,\omega) = \mathbb{i}(\omega_1 A_1 + \omega_2 A_2),$$

where

$$A_1 = \begin{pmatrix}u_0 & \rho_0 & 0 & 0\\0 & u_0 & 0 & \rho_0^{-1}\\0 & 0 & u_0 & 0\\0 & \rho_0 c_0^2 & 0 & u_0\end{pmatrix}, A_2 = \begin{pmatrix}v_0 & 0 & \rho_0 & 0\\0 & v_0 & 0 & 0\\0 & 0 & v_0 & \rho_0^{-1}\\0 & 0 & \rho_0 c_0^2 & v_0\end{pmatrix}$$

Now the eigenvalues of A_1 are: u_0, u_0, and $u_0 \pm c_0$ those of A_2 are: v_0, v_0, and $v_0 \pm c_0$. Therefore the equation is not strictly hyperbolic. Nonetheless, the matrix:

$$\bar{S} = \begin{pmatrix}0 & \rho_0 & 1 & -\rho_0\\0 & c_0 & 0 & c_0\\\sqrt{2}c_0 & 0 & 0 & 0\\0 & \rho_0 c_0^2 & 0 & -\rho_0 c_0^2\end{pmatrix}$$

diagonalizes A_1 and symmetrizes A_2, so the problem is symmetric hyperbolic and thus it is strongly well posed.

6

Ordinary differential equations

6.1 Exact solutions

- For the constant coefficient linear system

$$\mathbf{u}'(t) = A\mathbf{u}(t),$$

the solution to the initial value problem is

$$\mathbf{u}(t) = e^{A(t-t_0)}\mathbf{u}(t_0).$$

- For the nonhomogeneous linear system

$$\mathbf{u}'(t) = A\mathbf{u}(t) + \mathbf{g}(t),$$

the solution formula is known as *Duhamel's principle*

$$\mathbf{u}(t) = e^{A(t-t_0)}\mathbf{u}(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}\mathbf{g}(\tau)\, d\tau.$$

- For the nonlinear case,

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}, t), \quad \mathbf{u}(0) = \mathbf{a},$$

the Lipschitz continuity on $\mathbf{f}$ can guarantee the existence and uniqueness of the solution. A function $f(u)$ is Lipschitz continuous w.r.t u if $|f(u) - f(v)| \leq L|u - v|$ for some constant L. For instance, $f(u) = |u|$ is Lipschitz continuous because $||x| - |y|| \leq |x - y|$.

6.2 Some numerical methods

Consider solving $u'(t) = f(u,t), \quad u(0) = a$ with a uniform mesh in time $0 = t_0 < t_1 < \cdots t_n < \cdots$ and $t_n = n\Delta t$. Let U^n be the numerical solution at time t_n. Here are a few standard numerical methods:

- *Forward Euler*
$$U^{n+1} = U^n + \Delta t f(U^n, t_n).$$
- *Backward Euler*
$$U^{n+1} = U^n + \Delta t f(U^{n+1}, t_{n+1}).$$
- *Trapezoidal method*
$$U^{n+1} = U^n + \Delta t \frac{f(U^n, t_n) + f(U^{n+1}, t_{n+1})}{2}.$$
- *Midpoint method* (also called *leapfrog method*)
$$\frac{U^{n+1} - U^{n-1}}{2\Delta t} = f(U^n, t_n).$$
- A second order backward differentiation formula (BDF) method
$$\frac{3U^n - 4U^{n-1} + U^{n-2}}{2\Delta t} = f(U^n, t_n)$$
- A second order explicit Runge-Kutta method
$$U^* = U^n + \frac{1}{2}\Delta t f(U^n, t_n)$$
$$U^{n+1} = U^n + \Delta t f(U^*, t_n + \frac{1}{2}\Delta t).$$

Remark 6.1. *The BDF method and the leapfrog method are multi-step methods, for which the initial conditions must be generated by other methods. For instance, given the initial value $U^0 = u(0)$, to start the computation $\frac{3U^2 - 4U^1 + U^0}{2\Delta t} = f(U^2, t_2)$, we still need U^1, which can be generated by using forward Euler on a much finer mesh for the time interval $[0, \Delta t]$.*

6.3 Truncation errors

The local truncation error (LTE) is defined similarly as before: it is the residue of the scheme (in the form which recovers the differential equation

$u' - f(u,t) = 0$ as $\Delta t \to 0$) after replacing the numerical solution by the exact solution. For instance, the LTE of the leapfrog method is

$$\tau^n = \frac{u(t_{n+1}) - u(t_{n-1})}{2\Delta t} - f(u(t_n), t_n)$$

$$= \left[u'(t_n) + \frac{1}{6}\Delta t^2 u'''(t_n) + \mathcal{O}(\Delta t^4)\right] - u'(t_n) = \frac{1}{6}\Delta t^2 u'''(t_n) + \mathcal{O}(\Delta t^4).$$

If the local truncation error of a scheme is $\mathcal{O}(\Delta t^p)$, we say that the scheme is *consistent* of order p.

6.4 Convergence of the forward Euler's method

Let T be a given terminal time and assume $N = \frac{T}{\Delta t}$. We say the scheme is *convergent* if $\lim_{\Delta t \to 0} U^N = u(t_N)$. The scheme is *convergent* of order p if the global error $e^n = U^n - u(t_n) = \mathcal{O}(\Delta^p)$ for $n = 1, 2, \cdots, N$.

6.4.1 Linear problems

Now we prove the convergence of the forward Euler method solving $u' = \lambda u, u(0) = a$ with the exact solution as $u(t) = ae^{\lambda t}$. We have

$$U^{n+1} = U^n + \Delta t \lambda U^n = (1 + \Delta t \lambda) U^n,$$

thus

$$U^{n+1} = (1+\Delta t\lambda)U^n = (1+\Delta t\lambda)^2 U^{n-1} = (1+\Delta t\lambda)^{n+1} U^0 = a(1+\Delta t\lambda)^{n+1}.$$

So we want $\lim_{\Delta t \to 0}(1 + \Delta t\lambda)^N = e^{\lambda T}$ which is equivalent to $\lim_{N\to\infty}(1 + \frac{T}{N}\lambda)^N = e^{\lambda T}$. By the change of variable $N = N/(T\lambda)$, it becomes

$$\lim_{N\to\infty}\left[(1 + \frac{1}{N})^N\right]^{\lambda T} = e^{\lambda T}.$$

To obtain the global error $e^n = U^n - u(t_n)$, consider the LTE

$$\tau^n = \frac{u(t_{n+1}) - u(t_n)}{\Delta t} - \lambda u(t_n) = \frac{1}{2}\Delta t u''(t_n) + \mathcal{O}(\Delta t^2)$$

which implies

$$u(t_{n+1}) = (1 + \Delta t\lambda)u(t_n) + \Delta t\tau^n,$$

thus

$$e^{n+1} = (1 + \Delta t\lambda)e^n - \Delta t\tau^n.$$

Therefore,

$$\begin{aligned} e^{n+1} &= (1+\Delta t\lambda)e^n - \Delta t\tau^n \\ &= (1+\Delta t\lambda)((1+\Delta t\lambda)e^{n-1} - \Delta t\tau^{n-1}) - \Delta t\tau^n \\ &= \cdots \\ &= (1+\Delta t\lambda)^{n+1}e(0) - \Delta t\sum_{m=1}^{n}(1+\Delta t\lambda)^{n-m+1}\tau^m \\ &= -\Delta t\sum_{m=1}^{n}(1+\Delta t\lambda)^{n-m+1}\tau^m \end{aligned}$$

With the fact $|1+\Delta t\lambda| \le e^{\Delta t|\lambda|}$, we have

$$(1+\Delta t\lambda)^{n-m+1} \le e^{(n-m+1)\Delta t|\lambda|} \le e^{n\Delta t|\lambda|} \le e^{T|\lambda|}.$$

So we get

$$|e^n| \le e^{T|\lambda|}\left(\Delta t\sum_{m=1}^{n}|\tau^m|\right) \le e^{T|\lambda|}n\Delta t\max_{1\le m\le n}|\tau^m| \le e^{T|\lambda|}T\max_{1\le m\le n}|\tau^m| = \mathcal{O}(\Delta t)$$

6.4.2 Nonlinear problems

Now consider solving $u'(t) = f(u)$ and f satisfies the Lipschitz continuity $|f(u) - f(v)| \le L|u-v|$. For instance, if $f(u) = |u|$ then $L = 1$; if $f(u) = \sin(2u)$ then $L = 2$. We have

$$U^{n+1} = U^n + \Delta t f(U^n),$$

$$\tau^n = \frac{u(t_{n+1}) - u(t_n)}{\Delta t} - f(u(t_n)) = \frac{1}{2}\Delta t u''(t_n) + \mathcal{O}(\Delta t^2),$$

and

$$u(t_{n+1}) = u(t_n) + \Delta t f(u(t_n)) + \Delta t\tau^n.$$

Therefore

$$e^{n+1} = e^n + \Delta t\left(f(U^n) - f(u(t_n))\right) - \Delta t\tau^n.$$

The Lipschitz continuity implies

$$|f(U^n) - f(u(t_n))| \le L|U^n - u(t_n)| = L|e^n|,$$

thus

$$|e^{n+1}| \le |e^n| + \Delta tL|e^n| + \Delta t|\tau^n| = (1+\Delta tL)|e^n| + \Delta t|\tau^n|,$$

from which we can show by induction that

$$|e^n| \le (1+\Delta tL)^n|e^0| + \Delta t\sum_{m=1}^{n}(1+\Delta tL)^{n-m}|\tau^{m-1}|.$$

Assuming $|e^0| = 0$, we get $|e^n| \le e^{LT}T\max_{1\le m\le n}|\tau^{m-1}| = \mathcal{O}(\Delta t)$.

6.5 0-stability

We can define a more abstract concept 0-stability (0 refers to the stability when $\Delta t \to 0$) to conclude what we have seen for the convergence of forward Euler's method.

Consider solving a nonlinear equation $u' = f(u,t)$ with $u(0) = a$. Let $\mathcal{N}_{\Delta t}$ denote a difference scheme operator on any fuction y defined on the mesh points t_n (we say y is a mesh function). For example, the forward Euler operator is

$$\mathcal{N}_{\Delta t} y(t_n) = \frac{y(t_{n+1}) - y(t_n)}{\Delta t} - f(y(t_n), t_n).$$

Definition 6.1. *A scheme is 0-stable if there are positive constants K and h_0 such that for any two mesh functions x and z with $\Delta t \le h_0$,*

$$|x(t_n) - z(t_n)| \le K \left[|x(0) - z(0)| + \max_{1 \le m \le N} |\mathcal{N}_{\Delta t} x(t_m) - \mathcal{N}_{\Delta t} z(t_m)| \right], \quad 1 \le n \le N.$$

If we choose the mesh function x to be the numerical solution U^n and z to be the exact solution $u(t_n)$ in the definition above, then 0-stability simply says that the global error of the scheme has the same order as the local truncation error. Thus we have following fact

$$\text{consistency+0-stability} \;\Rightarrow\; \text{convergence}.$$

6.6 Absolute stability

The 0-stability is the "mathematical stability" to guarantee convergence. In practice, it is not sufficient to ensure the *numerical stability*. To understand the importance of numerical stability, consider the IVP for the scalar *test equation* $u' = \lambda u$ and $u(0) = u_0$ with $\lambda = a + \mathbb{i}\, b$ where a, b are real numbers. The exact solution is $u(t) = e^{(a+\mathbb{i}\, b)t} u_0$:

- If $a > 0$, then the solution is *unstable* in the sense that $\lim_{t \to +\infty} |u(t)| = +\infty$.
- If $a = 0$, then the solution oscillates/circles around the origin as time evolves.
- if $a < 0$, then the solution is *stable* in the sense that $\lim_{t \to +\infty} |u(t)| = 0$.

Now we focus on the stable case and consider the forward Euler scheme: $U^{n+1} = (1 + \Delta t \lambda) U^n$. The exact solution satisfies $|u(t_{n+1})| \le |u(t_n)|$ thus we would like to have the same property for our numerical solution $|U^{n+1}| \le |U^n|$, which we call *absolute stability*. For forward Euler, the absolute stability requires $|1 + \Delta t \lambda| \le 1$.

Let us see what happens if we do not have the absolute stability. Let $u_0 = 0$ then the exact solution $u(t) = 0$. Assume the numerical initial condition is $U^0 = 10^{-15}$ due to the round-off error in double precision floating point computation. If $|1 + \Delta t\lambda| = c > 1$, then $U^N = (1 + \Delta t\lambda)^N U^0$ thus $|U^N| = c^N|U^0| \to +\infty$ when $\Delta t \to 0$.

Definition 6.2. *For a given numerical method, the* **region of absolute stability** *(also called stability region) is that region of the complex z plane such that applying the method for the equation $u' = \lambda u$, with $z = \lambda\Delta t$ from within this region, yields an approximate solution satisfying the absolute stability requirement $|U^{n+1}| \le |U^n|$.*

So the region of absolute stability for forward Euler is $\{z : |1 + z| \le 1\}$ which is a disk of radius 1 with -1 as origin.

Consider backward Euler method $U^{n+1} = U^n + \Delta t\lambda U^{n+1}$. Then $U^{n+1} = \frac{1}{1-\Delta t\lambda}U^n$ and the absolute stability requires $|1 - z| \ge 1$.

For the trapezoidal method, the stability region is $\{z : |2 + z| \le |2 - z|$, which is the whole left plane including the imaginary axis.

Definition 6.3. *A method is said to be A-stable if its region of absolute stability contains the entire left plane $\Delta t Re(\lambda) < 0$.*

The backward Euler method is A-stable and the forward Euler is not. If λ is real and negative, a A-stable method is numerically stable with any positive time step Δt, but the forward Euler is stable only when $\Delta t \le -\frac{2}{\lambda}$.

For the trapezoidal method, we have

$$U^{n+1} = U^n + \Delta t\frac{\lambda U^n + \lambda U^{n+1}}{2},$$

thus $U^{n+1} = \frac{1+\frac{1}{2}\lambda\Delta t}{1-\frac{1}{2}\lambda\Delta t}U^n$. Therefore its stability region is $\{z : \frac{|2+z|}{|2-z|} \le 1\}$, which is A-stable.

6.7 Method of lines

Linear systems of ODE naturally arise in solving PDE. For example, consider solving the following initial boundary value problem

$$u_t = u_{xx}, \quad x \in (0, 1), \quad u(0) = u(1) = 0, \quad u(x, 0) = f(x). \tag{6.1}$$

We first discretize the spatial variable x on a uniform grid $x_i = i\Delta x$ $(i = 1, \cdots, M)$ with $\Delta x = \frac{1}{M+1}$, then we get an ODE system

$$U_i'(t) = \frac{1}{\Delta x^2}(U_{i-1}(t) - 2U_i(t) + U_{i+1}(t)),$$

or the matrix-vector form $\mathbf{U}'(t) = -\frac{1}{\Delta x^2}K\mathbf{U}$. Then we can use a numerical method for ODE to solve this system. Such an approach is called method of lines (MOL) discretization of the PDE. The ODE system obtained above is called a *semidiscrete* method,

6.8 A-stability in solving linear systems

Consider solving $\mathbf{u}'(t) = A\mathbf{u}(t)$ with a constant diagonalizable matrix $A = T\Lambda T^{-1}$. Take forward Euler method as an example, we have

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t A\mathbf{U}^n.$$

Now we analyze what the stability region implies by consider the change of variable $\mathbf{W} = T^{-1}\mathbf{U}$. Then the scheme above is equivalent to

$$\mathbf{W}^{n+1} = \mathbf{W}^n + \Delta t \Lambda \mathbf{W}^n.$$

Recall in Chapter 5, by Courant-Fischer-Weyl min-max principle, we use the largest and smallest eigenvalues of T^*T to estimate $\|\mathbf{U}\|/\|\mathbf{W}\|$. The eigenvalues of T^*T are preciously the square of singular values of T^*T. A different and simpler approach is to use $\|\mathbf{U}\| = \|T\mathbf{W}\| \le \|T\|\|\mathbf{W}\|$ and $\|\mathbf{W}\| = \|T^{-1}\mathbf{U}\| \le \|T^{-1}\|\|\mathbf{U}\|$. Notice that $\|T\|$ and $\|T^{-1}\|$ are the largest and smallest singular values.

If $|1 + \Delta t\lambda_i| \le 1$ then $\|W^n\| \le \|W^0\|$ thus $\|\mathbf{U}^n\| \le \|T\|\|T^{-1}\|\|\mathbf{U}^0\|$. By considering the SVD of T, we can see that $\|T\|\|T^{-1}\|$ is the ratio of the largest and smallest singular values of T. This ratio is also called the *condition number* of the matrix T. In general condition number may depend on the size of $\mathbf{U}$, which is huge in a semi discrete method solving PDEs. Fortunately, if A is real symmetric or complex Hermitian, then we can pick orthonormal eigenvectors so that $\|T\| = \|T^{-1}\|$=1.

Example 6.1. *Consider solving* (6.1) *in Section 6.7. The semidiscrete method is* $\mathbf{U}'(t) = -\frac{1}{\Delta x^2}K\mathbf{U}$. *The matrix* $A = -\frac{1}{\Delta x^2}K$ *with eigenvalues* $-\frac{1}{\Delta x^2}(2 - 2\cos(i\pi\frac{1}{M+1}))$ $(i = 1, \cdots, M)$, *where* M *is the number of grid points for spatial discretization. Thus forward Euler is stable if*

$$\Delta t \le \frac{1}{2}\Delta x^2.$$

Example 6.2. *Consider solving the wave equation with periodic boundary conditions*

$$u_t = u_x, \quad x \in (0, 1), \quad u(0) = u(1), \quad u(x, 0) = f(x). \tag{6.2}$$

Let us approximate the spatial derivative by the centered difference:

$$U_i'(t) = \frac{1}{2\Delta x}(U_{i+1}(t) - U_{i-1}(t)),$$

on the spatial grid points $x_i = i\Delta x$ $(i = 1, \cdots, M)$ *and* $\Delta x = \frac{1}{M}$. *The*

semidiscrete method is $\mathbf{U}'(t) = A\mathbf{U}$, *with*

$$A = \frac{1}{2\Delta x}\begin{pmatrix} 0 & 1 & & & & -1 \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ 1 & & & & -1 & 0 \end{pmatrix}.$$

The eigenvectors of A are the DFT matrix because A is a circulant matrix. The eigenvalues of A are either zero or purely imaginary because A is skew symmetric. (What are the eigenfunctions to the eigenvalue problem $u' = \lambda u, u(0) = u(1)$*?) Forward Euler is not stable for any nonzero time step because its stability region does not contain any nonzero part of the imaginary axis.*

6.9 Stiffness

An IVP is stiff if the step size needed to maintain stability of the forward Euler method is much smaller than the step size required to represent the solution accurately. In Example 6.1, the stability requires $\Delta t \leq \Delta x^2$ while we only need $\Delta t = \mathcal{O}(\Delta x)$ obtain the first order accuracy as $\Delta x \to 0$.

The following is an example in which a time step constraint is derived from accuracy requirement.

Example 6.3. *The harmonic oscillator*

$$u'' + \omega^2 u = 0, \quad u(0) = 1, u'(0) = 0,$$

has the solution $u = \cos(\omega t)$. *If the frequency* ω *is high,* $\omega \gg 1$ *, then the derivatives grow larger and larger, because*

$$\|u^{(p)}\|_\infty = \omega^p$$

The local truncation error of a discretization method of order p is at least

$$\mathcal{O}(\Delta t^p \omega^p).$$

For instance, the centered difference would be second order accurate and the leading term in the local truncation error is $\frac{1}{12}\Delta t^2 u^{(4)} = \frac{1}{12}\Delta t^2 \omega^4 \cos(\omega t)$ *This means that to recover an oscillatory solution* $u(t)$ *accurately, we need to at least require*

$$\Delta t < \frac{1}{\omega^2},$$

regardless of the order of the method. In fact, for $\Delta t > \frac{1}{\omega^2}$*, increasing the order of the method as such is useless.*

6.10 Runge-Kutta methods

High order methods include the linear multistep methods, and Runge-Kutta methods which are one-step methods. First we look at a few examples of Runge-Kutta methods for solving $u'(t) = f(u,t)$ and how they are derived.

Example 6.4. *If we use the midpoint rule for the integral in*

$$u(t_{n+1}) = u(t_n) + \int_{t_{n+1}}^{t_n} f(u,t)\,dt,$$

we get an implicit Runge-Kutta method:

$$U^{n+1} = U^n + \Delta t f\left(\frac{U^{n+1}+U^n}{2}, t_{n+\frac{1}{2}}\right).$$

If we approximate $\frac{U^{n+1}+U^n}{2}$ *by* $U^* = U^n + \frac{1}{2}\Delta t f(U^n, t_n)$*, we obtain the* **explict midpoint method** *(a second order accurate Runge-Kutta method):*

$$\begin{aligned} U^* &= U^n + \frac{1}{2}\Delta t f(U^n, t_n) \\ U^{n+1} &= U^n + \Delta t f(U^*, t_n + \frac{1}{2}\Delta t). \end{aligned}$$

Even tough U^* *is only first order accurate,* U^{n+1} *becomes second order accurate because* $f(U^*, t_n + \Delta t)$ *is multiplied by* Δt *when computing* U^{n+1}*. To check the local truncation error, replace only* U^n *and* U^{n+1} *in the scheme. For convenience, let* u *denote* $u(t_n)$ *and* f *denote* $f(u(t_n), t_n)$*), then we have*

$$U^* = u(t_n) + \frac{1}{2}\Delta t f(u(t_n), t_n) = u + \frac{1}{2}\Delta t f$$

and

$$\begin{aligned} \tau^n =& \frac{u(t_{n+1}) - u(t_n)}{\Delta t} - f(U^*, t_n + \frac{1}{2}\Delta t) \\ =& u'(t_n) + \frac{1}{2}u''(t_n)\Delta t + \frac{1}{6}u'''(t_n)\Delta t^2 \\ & - \left[f + \frac{1}{2}\Delta t\,(f_u f + f_t) + \frac{1}{2}\left(\frac{1}{2}\Delta t\right)^2 (f_{uu}f^2 + 2f_{tu}f + f_{tt})\right]. \end{aligned}$$

Thus $\tau^n = \mathcal{O}(\Delta t^2)$ *because the exact solution* u *satisfies*

$$\begin{aligned} u' &= f \\ u'' &= \frac{\partial}{\partial t} f = f_u u' + f_t. \end{aligned}$$

The obtained explicit midpoint method gives us a first real taste of the original Runge-Kutta idea: a higher order is achieved by repeated function evaluations of f within the interval $[t_n, t_{n+1}]$.

Example 6.5. *If we use the trapezoidal rule for the integral, then we get an implicit scheme*

$$U^{n+1} = U^n + \frac{1}{2}\Delta t f(U^n, t_n) + \frac{1}{2}\Delta t f(U^{n+1}, t_{n+1}).$$

If approximating U^{n+1} by forward Euler, we get the explicit trapezoidal method (another explicit two-stage Runge-Kutta method of order two):

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

Example 6.6. *A very popular fourth order Runge-Kutta method is related to the Simpson's quadrature rule:*

$$u(t_{n+1}) - u(t_n) \approx \Delta t \left[\frac{1}{6} f(u(t_n), t_n) + \frac{4}{6} f(u(t_n + \frac{1}{2}\Delta t), t_n + \frac{1}{2}\Delta t) + \frac{1}{6} f(u(t_{n+1}), t_{n+1})\right].$$

The scheme is given as

$$U^{(1)} = U^n$$
$$U^{(2)} = U^n + \frac{1}{2}\Delta t f(U^{(1)}, t_n)$$
$$U^{(3)} = U^n + \frac{1}{2}\Delta t f(U^{(2)}, t_{n+\frac{1}{2}})$$
$$U^{(4)} = U^n + \Delta t f(U^{(3)}, t_{n+\frac{1}{2}})$$
$$U^{n+1} = U^n + \frac{1}{6}\Delta t \left[f(U^{(1)}, t_n) + 2f(U^{(2)}, t_{n+\frac{1}{2}}) + 2f(U^{(3)}, t_{n+\frac{1}{2}}) + f(U^{(4)}, t_{n+1})\right],$$

where $t_{n+\frac{1}{2}} = t_n + \frac{1}{2}\Delta t$.

6.10.1 Order of accuracy

A general s-stage Runge-Kutta method (which may not be s order accurate) can be written as

$$U^{(i)} = U^n + \Delta t \sum_{j=1}^{s} a_{ij} f(U^{(j)}, t_n + c_j \Delta t)$$
$$U^{n+1} = U^n + \Delta t \sum_{i=1}^{s} b_i f(U^{(i)}, t_n + c_i \Delta t),$$

which can be represented conveniently in a shorthand notation called Butcher tableau:

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}$$

Examples:

- Forward Euler:

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

- One-parameter family of second order methods:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \alpha & \alpha & 0 \\ \hline & 1-\frac{1}{2\alpha} & \frac{1}{2\alpha} \end{array}$$

  For $\alpha = 1$, we have the explicit trapezoidal method, and for $\alpha = \frac{1}{2}$ it is the explicit midpoint method.

- One-parameter families of third order 3-stage methods:

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{2}{3} & \frac{2}{3} & 0 & 0 \\ \frac{2}{3} & \frac{2}{3}\text{-}\frac{1}{4\alpha} & \frac{1}{4\alpha} & 0 \\ \hline & \frac{1}{4} & \frac{3}{4}-\alpha & \alpha \end{array}$$

- The fourth order Runge-Kutta:

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

Consistency requires $\sum_{j=1}^{s} a_{ij} = c_i$ and $\sum_{i=1}^{s} b_i = 1$.

We see that there are s-stage explicit Runge-Kutta methods of order s, at least for $s \leq 4$. One may wonder if it is possible to obtain order $p > s$, and if it is possible to always maintain at least $p = s$. The answers are both negative. An explicit Runge-Kutta method can have at most order s, i.e., $p \leq s$. For explicit s-stage Runge-Kutta method, we have

number of stages	1	2	3	4	5	6	7	8	9	10
attainable order	1	2	3	4	4	5	6	6	7	7

6.10.2 0-stability and convergence

Since Runge-Kutta methods are one-step methods, the discussion of 0-stability will be very similar to forward Euler for explicit Runge-Kutta methods (Implicit Function Theorem is needed for implicit schemes).

Consider solving the nonlinear equation $u' = f(u)$ where f is Lip-conitinuous w.r.t u with Lipschitz constant L. The one-step method takes the form

$$U^{n+1} = U^n + \Delta t \Psi(U^n, t_n, \Delta t),$$

then Ψ is Lip-continuous w.r.t. U^n with Lip-constant L'.

For example, for the explicit midpoint method we have

$$U^{n+1} = U^n + \Delta t f\left(U^n + \frac{1}{2}\Delta t f(U^n)\right),$$

$$u(t_{n+1}) = u(t_n) + \Delta t f\left(u(t_n) + \frac{1}{2}\Delta t f(u(t_n))\right) + \Delta t \tau^n.$$

Subtracting these two equations, we get

$$\begin{aligned}
|e^{n+1}| &\le |e^n| + \Delta t L |e^n + \frac{1}{2}\Delta t \left(f(U^n) - f(u(t_n))\right)| + \Delta t |\tau^n| \\
&\le |e^n| + \Delta t L |e^n| + \Delta t L \frac{1}{2}\Delta t L |e^n| + \Delta t |\tau^n| \\
&= (1 + \Delta t L + \frac{1}{2}\Delta t^2 L^2)|e^n| + \Delta t |\tau^n| \\
&= (1 + \Delta t L')|e^n| + \Delta t |\tau^n|,
\end{aligned}$$

where $L' = L + \frac{1}{2}\Delta t^2 L$. The rest will be similar to Section 6.4.2.

6.10.3 Absolute stability of explicit Runge-Kutta methods

Consider the test equation $u' = \lambda u$ and an explicit s-stage Runge-Kutta method of order p with a Butcher tableau

$$\begin{array}{c|c} \mathbf{c} & A \\ \hline & b^T \end{array}$$

The $s \times s$ matrix A is lower triangular thus $A^s = \mathbf{0}$. Let $\mathbf{U} = \begin{pmatrix} U^{(1)} \\ U^{(2)} \\ \vdots \\ U^{(s)} \end{pmatrix}$, and $\mathbf{e} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$. Then the scheme can be written as

$$\begin{aligned}
\mathbf{U} &= U^n \mathbf{e} + \Delta t \lambda A \mathbf{U}, \\
U^{n+1} &= U^n + \Delta t \lambda \mathbf{b}^T \mathbf{U}.
\end{aligned}$$

Thus we obtain

$$\begin{aligned} U^{n+1} &= U^n + \Delta t\lambda \mathbf{b}^T (I - \Delta t\lambda A)^{-1} U^n \mathbf{e} \\ &= \left[1 + z\mathbf{b}^T (I - zA)^{-1}\mathbf{e}\right] U^n \\ &= \left[1 + z\mathbf{b}^T (I + zA + \cdots + z^{s-1}A^{s-1})\mathbf{e}\right] U^n. \end{aligned}$$

Here we use the fact that $(I - zA)(I + zA + \cdots + z^k A^k) = I - z^{k+1}A^{k+1} = I$ if $k \geq s - 1$. Let $R(z) = 1 + z\mathbf{b}^T(I + zA + \cdots + z^{s-1}A^{s-1})\mathbf{e}$ then

$$R(z) = 1 + z + \frac{z^2}{2!} + \cdots + \frac{z^p}{p!} + \sum_{j=p+1}^{s} z^j \mathbf{b}^T A^{j-1}\mathbf{e}.$$

To prove it, notice that the exact solution satisfies $u(t_{n+1}) = e^{\Delta t\lambda}u(t_n) = e^z u(t_n) = (1 + z + \frac{z^2}{2!} + \cdots + \frac{z^p}{p!} + \cdots)u(t_n)$. And the local truncation error is of order p implies $u(t_{n+1}) - R(z)u(t_n) = \mathcal{O}(\Delta t^{p+1})$ thus we must have $R(\Delta t\lambda) - e^{\Delta t\lambda} = \mathcal{O}(\Delta t^{p+1})$ in an p-th order accurate method.

Remark 6.2. *A byproduct of this discussion is a collection of necessary conditions for constructing an explicit p-th order Runge-Kutta method:*

$$\begin{aligned} \mathbf{b}^T\mathbf{e} = \sum_{i=1}^{s} b_i &= 1 \\ \mathbf{b}^T A\mathbf{e} &= \frac{1}{2} \\ \mathbf{b}^T A^2\mathbf{e} &= \frac{1}{3!} \\ &\vdots \\ \mathbf{b}^T A^{p-1}\mathbf{e} &= \frac{1}{p!} \end{aligned}$$

The stability region is defined by $|R(z)| = 1$. To plot the region, there are at least two methods:

- Find solutions to $R(z) = e^{\mathbb{i}\,\theta}$ for $\theta \in [0, 2\pi]$ which will give the boundary curve of the stability region. We sample θ at uniform N points, i.e., consider $\theta_i = (i-1)\frac{2\pi}{N}$ for $i = 1, 2, \cdots, N$. For $\theta_1 = 0$, the solution to $R(z) = 1$ is $z_1 = 0$. To solve $R(z_j) = e^{\mathbb{i}\,\theta_j}$ $(j \geq 2)$, we can use Newton's method with z_{j-1} as an initial guess. This is an elementary example of *continuation* method.

- Brutal force method: use enough uniform grid points z_{ij} on the complex plane, mark/color the point if $|R(z_{ij})| \leq 1$.

Remark 6.3. *We can also use root function in MATLAB to find all s roots of the polynomial $R(z) - e^{\mathbb{i}\theta}$. If $s >= 5$, there is no explicit polynomial root formula, implied by the Fundamental Theorem of Galois theory. So how can MATLAB still find all roots?*

In particular, if $s = p$ (possible only for $s = 1, 2, 3, 4$), then we get

$$R(z) = 1 + z + \frac{z^2}{2!} + \cdots + \frac{z^s}{s!}.$$

For fixed $s \leq 4$, explicit s-stage methods of order s are not unique but they always have the same stability region! See Figure 6.1 for the stability region of Runge-Kutta methods.

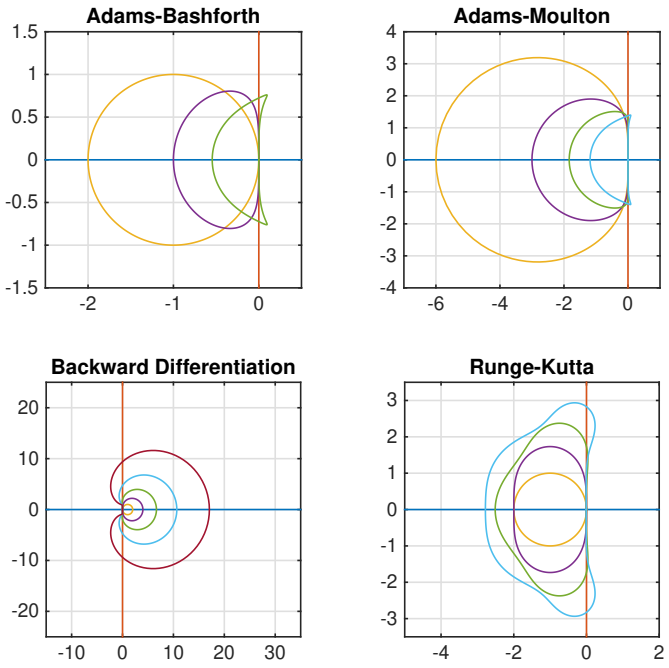


Figure 6.1: Stability region boundary curves for methods of increasing order colored by yellow, violet, green and blue. The Runge-Kutta methods are explicit s-stage with order s. For Adams and Runge-Kutta methods, the region is enclosed by the curve. For backward differentiation formulae methods, the region is outside of the curve. For Adams-Moulton, yellow curve is the third order method. For the other three methods, yellow curve is the first order method (forward or backward Euler).

We can also see that explicit RK methods cannot be A-stable.

6.11 Linear multistep methods

Let F^n denote $f(U^n, t_n)$. For solving the equation $u' = f(u,t)$, a general linear multistep method can be written as

$$\sum_{j=0}^{k} \alpha_j U^{n+1-j} = \Delta t \sum_{j=0}^{k} \beta_j F^{n+1-j}. \tag{6.3}$$

The scheme is linear w.r.t. each f^n (it is nonlinear in Runge-Kutta methods). This is why it is called linear multistep methods.

The normalization $\alpha_0 = 1$ is often assumed to fix the scale factor. The scheme is explicit if $\beta_0 = 0$.

Linear multistep methods typically come in families. The most popular for nonstiff problems is the Adams family and the most popular for stiff problems is the BDF family.

6.11.1 Adams methods

Given ODE $u' = f(u,t)$, we can consider the integral form

$$u(t_{n+1}) = u(t_n) + \int_{t_n}^{t_{n+1}} f(u(t), t)\, dt.$$

The k-step explicit Adams method is obtained by interpolating f through the previous points $t = t_n, t_{n-1}, \cdots, t_{n-k+1}$. Adams methods are the most popular among explicit multistep methods. A simple exercise in polynomial interpolation yields the formulae

$$U^{n+1} = U^n + \Delta t \sum_{j=1}^{k} \beta_j F^{n+1-j},$$

with

$$\beta_j = (-1)^{j-1} \sum_{i=j-1}^{k-1} \binom{i}{j-1} \gamma_i, \quad \gamma_i = (-1)^i \int_0^1 \binom{-s}{i}\, ds,$$

where the binomial coefficients are $\binom{s}{i} = \frac{s(s-1)\cdots(s-i+1)}{i!}$, $\binom{s}{0} = 1$. This is called Adams-Bashforth, in which the local truncation error turns out to be $C_{p+1}\Delta t^p u^{(p+1)}(t_n) + \mathcal{O}(\Delta t^{p+1})$, where $p = k$.

Examples of Adams-Bashforth methods:

$$k = 1: \quad U^{n+1} = U^n + \Delta t F^n$$

$$k = 2: \quad U^{n+1} = U^n + \Delta t(\frac{3}{2}F^n - \frac{1}{2}F^{n-1})$$

$$k = 3: \quad U^{n+1} = U^n + \Delta t(\frac{23}{12}F^n - \frac{16}{12}F^{n-1} + \frac{5}{12}F^{n-2})$$

$$k = 4: \quad U^{n+1} = U^n + \Delta t(\frac{55}{24}F^n - \frac{59}{24}F^{n-1} + \frac{37}{24}F^{n-2} - \frac{9}{24}F^{n-3})$$

Unfortunately, the Adams-Bashforth methods are explicit methods with very small regions of absolute stability. This has motivated the implicit versions of Adams methods, also called Adams-Moulton. The k-step implicit Adams method is derived similarly to the explicit method. The difference is that for this method, the interpolating polynomial is of degree $\leq k$ and it interpolates f at the unknown value t_{n+1} as well:

$$U^{n+1} = U^n + \Delta t \sum_{j=0}^{k} \beta_j F^{n+1-j},$$

The order of the k-step Adams-Moulton method is $p = k + 1$ because $k + 1$ points are used in the underlying polynomial interpolation). An exception is the backward Euler in which $k = 1$ and F^n is not used, yielding $p = k = 1$. A few examples of Adams-Moulton methods:

$$\begin{aligned}
p = 1, k = 1: &\quad U^{n+1} = U^n + \Delta t F^{n+1} \\
p = 2, k = 1: &\quad U^{n+1} = U^n + \Delta t(\frac{1}{2}F^{n+1} + \frac{1}{2}F^n) \\
p = 3, k = 2: &\quad U^{n+1} = U^n + \Delta t(\frac{5}{12}F^{n+1} + \frac{8}{12}F^n - \frac{1}{12}F^{n-1}) \\
p = 4, k = 3: &\quad U^{n+1} = U^n + \Delta t(\frac{9}{24}F^{n+1} + \frac{19}{24}F^n - \frac{5}{24}F^{n-1} + \frac{1}{24}F^{n-2})
\end{aligned}$$

They have much larger stability regions than the Adams-Bashforth methods. But they are implicit.

6.11.2 Backward Differentiation Formulae

The BDF methods are derived by differentiating the polynomial which interpolates past values of u, and setting the derivative at t_{n+1} to $f(U^{n+1}, t_{n+1})$. The k-step BDF method, which has order p = k, has the form

$$\sum_{j=0}^{k} \alpha_j U^{n+1-j} = \Delta t \beta_0 F^{n+1}.$$

The BDF formulae are implicit. A few examples:

$$\begin{aligned}
k = 1: &\quad U^{n+1} = U^n + \Delta t F^{n+1} \\
k = 2: &\quad \frac{3U^{n+1} - 4U^n + U^{n-1}}{2\Delta t} = F^{n+1} \\
k = 3: &\quad \frac{11U^{n+1} - 18U^n + 9U^{n-1} - 2U^{n-2}}{6\Delta t} = F^{n+1}
\end{aligned}$$

6.11.3 Order of accuracy

Given a multistep method (6.3), it is much easier to check its local truncation error by Taylor expansion than Runge-Kutta methods.

6.11.4 Characteristic polynomials

Given a multistep method (6.3), we define characteristic polynomials $\rho(\xi)$ and $\sigma(\xi)$ as

$$\rho(\xi) = \sum_{j=0}^{k} \alpha_j \xi^{k-j},$$

$$\sigma(\xi) = \sum_{j=0}^{k} \beta_j \xi^{k-j}.$$

A linear multistep method (6.3) is consist (LTE goes to zero as $\Delta t \to 0$) if and only if $\rho(1) = 0$ and $\rho'(1) = \sigma(1)$.

6.11.5 0-stability and convergence

The 0-stability can be defined for LMM as for one-step method but it is cumbersome to check. Fortunately, it turns out that it is equivalent to a simple condition on the roots of the characteristic polynomial.

Theorem 6.1. *The linear multistep method is 0-stable if all roots of the characteristic polynomial $\rho(\xi)$ satisfy*

- $|\xi_j| \le 1$, $j = 1, 2, \cdots, k$.
- *If $|\xi_j| = 1$ then ξ_j is a simple root (not repeated).*

If this root condition is satisfied, the method is accurate to order p, and the initial values are accurate to order p, then the method is convergent to order p.

This root condition is derived by checking the 0-stability of (6.3) solving a very simple problem

$$u'(t) = 0, \quad u(0) = 0.$$

The scheme (6.3) becomes a linear difference equation

$$\sum_{j=0}^{k} \alpha_j U^{n+1-j} = 0.$$

Given initial values

$$U^0, U^1, \cdots, U^{k-1},$$

we want to solve this difference equation. We will first construct a solution then show it is the only solution. Assume $U^n = \xi^n$ (on the right hand side, it is ξ to the power n but n is only an index on the left hand side), then

$$\sum_{j=0}^{k} \alpha_j \xi^{n+1-j} = 0.$$

Notice that $\alpha_0 = 1$ thus $\xi \neq 0$. We can divide both sides by ξ^{n-k+1}, then

$$\sum_{j=0}^{k} \alpha_j \xi^{k-j} = 0,$$

which is $\rho(\xi) = 0$. Suppose $\rho(\xi)$ has k distinct roots $\xi_1, \cdots, \xi_k$, then we find k linearly independent solutions to the difference equation. And a general solution to the difference equation can be be written as

$$U^n = c_1 \xi_1^n + c_2 \xi_2^n + \cdots + c_k \xi_k^n,$$

where the coefficients c_i are uniquely determined by the $k \times k$ system

$$\begin{aligned} c_1 + c_2 + \cdots + c_k &= U^0 \\ c_1 \xi_1 + c_2 \xi_2 + \cdots + c_k \xi_k &= U^1 \\ \vdots \quad & \quad \vdots \\ c_1 \xi_1^{k-1} + c_2 \xi_2^{k-1} + \cdots + c_k \xi_k^{k-1} &= U^{k-1}. \end{aligned}$$

So we have just constructed one solution. Next we only need to show there is only one solution to the initial value problem of the difference equation. If U^n and V^n are both solutions, then $W^n = U^n - V^n$ satisfies the difference equation with zero initial values. Thus W^n can only be zero (solve the difference equation with zero initial states, you get only zero).

Suppose we compute the numerical solution U^N at time $T = 1$. If one of the roots has magnitude larger than 1, then $\lim_{\Delta t \to 0} |U^N| = \lim_{N \to +\infty} |U^N| = +\infty$. Therefore all roots must satisfy $|\xi| \leq 1$.

Now suppose we have repeated roots $\xi_1 = \xi_2$, then the solution becomes

$$U^n = c_1 \xi_1^n + c_2 n \xi_1^n + c_3 \xi_3^n + \cdots + c_k \xi_k^n,$$

thus $|\xi_1| = 1$ will also imply divergence.

It turns out that the root conditions is all that is needed for convergence

Theorem 6.2. *For LMMs applied to the initial value problem for $u' = f(u,t)$ where f is Lipschitz continuous w.r.t. u,*

$$\textit{consistency} + \textit{0-stability} \Rightarrow \textit{convergence}.$$

Note that the root condition guaranteeing 0-stability relates to the characteristic polynomial $\rho(\xi)$ alone. Also, for any consistent method the polynomial $\rho(\xi)$ has the root 1. One-step methods have no other roots, which again highlights the fact that they are automatically 0-stable.

Example 6.7. *The LMM*

$$U^{n+1} - 3U^n + 2U^{n-1} = -\Delta t F^{n-1},$$

has an LTE of first order. Apply it to $u' = 0$ then U^n can be explicitly solved in terms of U^0 and U^1. we obtain

$$U^n = 2U^0 - U^1 + 2^n(U^1 - U^0).$$

Unless $U^1 = U^0 = 0$, the approximate solution is never convergent to constant zero.

Example 6.8. *Applying the consistent LMMs*

$$U^{n+1} - 2U^n + U^{n-1} = \frac{1}{2}\Delta t(F^{n+1} - F^{n-1}),$$

to $u' = 0$ gives

$$U^{n+1} - 2U^n + U^{n-1} = 0.$$

The characteristic polynomial is

$$\rho(\xi) = \xi^2 - 2\xi + 1 = (\xi - 1)^2.$$

The general solution is

$$U^n = U^0 + (U^1 - U^0)n.$$

If $U^0 = 0$ and $U^1 = \Delta t$, then $U^N = \Delta t N = T \to T$ as $\Delta t \to 0$.

Example 6.9. *Consider*

$$U^{n+1} = -4U^n + 5U^{n-1} + 4\Delta t F^n + 2\Delta t F^{n-1}.$$

In terms of the local truncation error, this is the most accurate explicit 2-step method. However, $\rho(\xi) = (\xi - 1)(\xi + 5)$. The root condition is violated.

6.11.6 Stability region

Applying (6.3) to the test equation $u' = \lambda u$, we get

$$\sum_{j=0}^{k} (\alpha_j - z\beta_j) U^{n+1-j} = 0.$$

where $z = \Delta t \lambda$. The *stability polynomial* is denoted by

$$\pi(\xi, z) = \rho(\xi) - z\sigma(\xi).$$

The LMM is absolutely stable for a particular value of z if errors introduced in one time step do not grow in future time steps.

Definition 6.4. *The region of absolute stability for the LMM is the set of points z in complex plane for which the polynomial $\pi(\xi, z)$ satisfies the root condition.*

Finding the region of absolute stability is simple for linear multistep methods. Just look for the boundary

$$z = \frac{\rho(e^{\mathbb{i}\,\theta})}{\sigma(e^{\mathbb{i}\,\theta})}, \quad \theta \in [0, 2\pi].$$

Problem 6.1. *Follow the discussions in Section 6.11.5 to show that the root condition of $\pi(\xi, z)$ can guarantee the numerical solution to $u' = \lambda u$ does not blow up (in what sense?) as $\Delta t \to 0$.*

Example 6.10. *Recall that as absolute stability for one-step methods such as Runge-Kutta methods is to require $|U^{n+1}/U^n| \leq 1$ for solving $u' = \lambda u$. The definition of absolute stability region for LMMs is related to the solution of difference equation thus not necessarily the same. Let us consider the second order BDF method as an example:*

$$3U^{n+1} - 4U^n + U^{n-1} = 2\Delta t F^{n+1}.$$

The stability polynomial is given as $3\xi^2 - 4\xi + 1 = 2z\xi^2$. Thus to draw the boundary curve of the stability region, we have

$$z = \frac{3\xi^2 - 4\xi + 1}{\xi^2}, \quad \xi = e^{\mathbb{i}\,\theta}, \theta \in [0, 2\pi].$$

On the other hand, if we have to define something similar to $|U^{n+1}/U^n| \leq 1$, then it is natural to rewrite the scheme in the one-step matrix vector form. For $u' = \lambda u$, the second order BDF method becomes

$$3U^{n+1} - 4U^n + U^{n-1} = 2\Delta t \lambda U^{n+1},$$

thus

$$U^{n+1} = \frac{4}{3 - 2z} U^n - \frac{1}{3 - 2z} U^{n-1}, \quad z = \Delta t \lambda.$$

We can rewrite it as

$$\begin{pmatrix} U^{n+1} \\ U^n \end{pmatrix} = \begin{pmatrix} \frac{4}{3-2z} & \frac{-1}{3-2z} \\ 1 & 0 \end{pmatrix} \begin{pmatrix} U^n \\ U^{n-1} \end{pmatrix}.$$

For computing eigenvalues of the matrix $A = \begin{pmatrix} \frac{4}{3-2z} & \frac{-1}{3-2z} \\ 1 & 0 \end{pmatrix}$, we have

$$det(\xi I - A) = \left(\xi - \frac{4}{3 - 2z}\right)\xi + \frac{1}{3 - 2z} = 0,$$

which becomes

$$3\xi^2 - 4\xi + 1 = 2z\xi^2.$$

The natural choice for defining $|U^{n+1}/U^n| \leq 1$ *in multi-step methods is to require eigenvalues of A to be insied unit circle. Thus we end up with the same equation for drawing a curve derived from the one-step method perspective:*

$$z = \frac{3\xi^2 - 4\xi + 1}{2\xi^2}, \quad \xi = e^{\mathbb{i}\,\theta}, \theta \in [0, 2\pi].$$

6.11.7 Strong stability

Consider (6.3) for the test equation $u' = \lambda u$:

$$\sum_{j=0}^{k} \alpha_j U^{n+1-j} - \Delta t \sum_{j=0}^{k} \beta_j \lambda U^{n+1-j} = 0.$$

The solution to this linear difference equation is related to roots of the polynomial

$$\phi(\xi) = \rho(\xi) - \Delta t \lambda \sigma(\xi).$$

Since the exact solution for $u(0) = 1$ is $u = e^{\lambda t} = (e^{\Delta t \lambda})^n$, we expect one root to approximate $e^{\Delta t \lambda}$, which is required from consistency. That root is called *principal root*. The other roots called *extraneous roots*.

If $Re(\lambda) < 0$, then the exact solution decays and we must prevent any growth in the approximate solution. This is not possible for all such λ if there are extraneous roots of the polynomial $\phi(\xi)$ with magnitude 1. For $\Delta t > 0$ sufficiently small the difference equation must be asymptotically stable in this case. We define a linear multistep method to be

- *strongly stable* if all roots of $\rho(\xi) = 0$ are inside the unit circle except for the root $\xi = 1$.
- *weakly stable* if it is 0-stable but not strongly stable.

Weakly stable methods can be numerically unstable for some problems. Strongly stable k-step methods can have at most order k +1.

Example 6.11. *The leapfrog method*

$$\frac{U^{n+1} - U^{n-1}}{2\Delta t} = F^n$$

has characteristic polynomial $\rho(\xi) = \xi^2 - 1$ *with two roots* $\xi = \pm 1$ *on the unit circle thus it is weakly stable. The stability region boundary curve is determined by*

$$z = \frac{\rho(e^{\mathbb{i}\,\theta})}{2\sigma(e^{\mathbb{i}\,\theta})} = \frac{(e^{\mathbb{i}\,\theta})^2 - 1}{2e^{\mathbb{i}\,\theta}} = \frac{1}{2}(e^{\mathbb{i}\,\theta} - e^{-\mathbb{i}\,\theta}) = \mathbb{i}\sin\theta.$$

Therefore the stability region boundary is an interval $[-\mathbb{i}, \mathbb{i}]$ *on the imaginary axis. To determine the stability region, we can check the root conditions for the polynomial*

$$\pi(\xi, z) = \xi^2 - 1 - 2z\xi, \quad z \in [-\mathbb{i}, \mathbb{i}].$$

If $z = \mathbb{i}$*, then* $\pi(\xi, \mathbb{i})$ *has two same roots* $\xi = \mathbb{i}$ *thus the root condition is not satisfied thus not absolutely stable. Similarly, we do not have the absolute stability for* $z = -\mathbb{i}$*. For* $z \in (-\mathbb{i}, \mathbb{i})$*, the roots of* $\pi(\xi, z)$ *are* $\xi = z \pm \sqrt{z^2 + 1}$*. The stability region is the interval* $(-\mathbb{i}, \mathbb{i})$*.*

Therefore the leapfrog method is not numerically stable for solving problems like diffusion, e.g., Example 6.1, which involves $u' = \lambda u$ with real negative λ. On the other hand, it is well suited for problems involving only purely imaginary λ, e.g., hyperbolic problems like Example 6.2. The very popular FDTD (finite-difference time-domain or Yee's method) for Maxwell's equations uses centered difference in both space and time to achieve second order accuracy, which is the leapfrog method in time on staggered grids.

7

Finite difference schemes for linear time-dependent problems

7.1 Basic concepts, definitions and notation

Consider a general initial value problem for linear partial differential equations:

$$\begin{aligned} u_t(x,t) &= \mathcal{P}\left(x,t,\frac{\partial}{\partial x}\right)u(x,t), \\ u(x,0) &= f(x), \end{aligned} \tag{7.1}$$

where x is a vector of s components: $x = (x_1, \cdots, x_s)$, u is a vector of p components: $u(x,t) = (u_1(x,t), \cdots, u_p(x,t))$ and $\mathcal{P}$ is a polynomial of $\frac{\partial}{\partial x}$. If the highest order time derivative in a linear partial differential equation is $\frac{\partial^m}{\partial t^m}u$, then we can always rewrite it in the form of (7.1) as a system for a unknown vector $[u, \frac{\partial}{\partial t}u, \cdots, \frac{\partial^{m-1}}{\partial t^{m-1}}u]^T$. For instance, the two way wave equation $u_{tt} = u_{xx}$ can be written as

$$\frac{\partial}{\partial t}\begin{pmatrix} u \\ u_t \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{\partial^2}{\partial x^2} & 0 \end{pmatrix}\begin{pmatrix} u \\ u_t \end{pmatrix}. \tag{7.2}$$

Remark 7.1. *Let $(v_1, v_2)^T$ denote the unknown functions in* (7.2) *and take the Fourier transform, then we get*

$$\frac{\partial}{\partial t}\begin{pmatrix} \hat{v}_1(t) \\ \hat{v}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega^2 & 0 \end{pmatrix}\begin{pmatrix} \hat{v}_1(t) \\ \hat{v}_2(t) \end{pmatrix}.$$

Notice that $\begin{pmatrix} 0 & 1 \\ -\omega^2 & 0 \end{pmatrix}$ *is diagonalizable with eigenvalues $\pm \mathbb{i}\,\omega$, thus* (7.2) *is the convection to two directions even though $\frac{\partial^2}{\partial x^2}$ is the only spatial differential operator in* (7.2).

For computational convenience we will restrict the domain of the solution $u(x,t)$ of (7.1) to a bounded region, even though it might be defined for all x and all $t > 0$. On this bounded region we construct a grid of points, discretizing both space (in each of the space coordinates) and time. For this purpose, we specify the step sizes Δt and Δx_i for $i = 1, \cdots, s$ and define the grid points as points of the form $(x_1, \cdots, x_s, t_n)$, where:

$$t_n = n\Delta t$$

$$x_{j_i} = j_i \Delta x_i, i = 0, \cdots, N_i.$$

Although in many practical applications it is preferable to define suitable varying step sizes, we have chosen here constant ones for simplicity of notation. However, the concepts and properties discussed in this chapter can be readily generalized to the variable step-size case.

The main idea of any finite difference scheme attempting to approximate the values of $u(x,t)$ by computer methods is to construct a vector of p components for given integers n and $j_1, \cdots, j_s$ with $0 \leq j_i \leq N_i$, which we call $U^n_{j_1,\cdots,j_s}$ and which "approximates" the value of $u(x_{j_1}, \cdots, x_{j_s}, n\Delta t)$.

For fixed n, $U^n_{j_1,\cdots,j_s}$ is therefore a vector-valued function of the set of integers $\{j_i = 0, \cdots, N_i; 0 \leq i \leq s\}$. For such functions, we define the k-th shift operator E_k to be the operator that shifts the index j_k to its right $(j_k + 1)$, that is:

$$E_k U^n_{j_1,\cdots,j_k,\cdots,j_s} = U^n_{j_1,\cdots,j_k+1,\cdots,j_s}, \quad 1 \leq k \leq s.$$

Definition 7.1. *A finite difference scheme is a recursion formula of the form:*

$$B_0(E_1, \cdots, E_s)V^{n+1}_\alpha = B_1(E_1, \cdots, E_s))V^n_\alpha \tag{7.3}$$

where $\alpha = j_1, \cdots, j_s$ is a multi-index, and $B_0(E_1, \cdots, E_s)$ and $B_1(E_1, \cdots, E_s)$ are functions of the operators E_i, $1 \leq i \leq s$. If B_0 is the identity operator, we say the scheme is explicit. Otherwise it is called an implicit scheme.

We now give some examples to illustrate our notation.

Example 7.1. *Consider the two-dimensional problem:*

$$u_t = u_x + u_y,$$

where $u(x,y,t)$ is a real valued function. Let Δt, Δx and Δy be positive, fixed quantities. One possible finite difference scheme is given by:

$$\begin{aligned} U^{n+1}_{i,j} = &\frac{1}{4}\left(U^n_{i+1,j+1} + U^n_{i-1,j+1} + U^n_{i+1,j-1} + U^n_{i-1,j-1}\right) \\ &+ \frac{\Delta t}{2\Delta x}(U^n_{i+1,j} - U^n_{i-1,j}) + \frac{\Delta t}{2\Delta y}(U^n_{i,j+1} - U^n_{i,j-1}) \end{aligned}$$

which in terms of the shift operators E_1 and E_2, can be written in the form:

$$U_{i,j}^{n+1} = \left(\frac{1}{4}(E_1 + E_1^{-1})(E_2 + E_2^{-1}) + \frac{\Delta t}{2\Delta x}(E_1 - E_1^{-1}) + \frac{\Delta t}{2\Delta y}(E_2 - E_2^{-1})\right) U_{i,j}^{n}.$$

Thus $V^n = U^n$ in this case.

Example 7.2. *Conside the Leapfrog scheme for the one-way wave equation:*

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

In order to write down this scheme in the form (7.3)*, we define the two dimensional vector:*

$$V_j^n = \begin{pmatrix} U_j^n \\ U_j^{n-1} \end{pmatrix},$$

and express $V_j^{n+1} = B_1(E)V_j^n$, where now $B_1(E)$ is a 2×2 matrix depending on the shift operator E. In fact, since:

$$\begin{pmatrix} U_j^{n+1} \\ U_j^n \end{pmatrix} = \begin{pmatrix} \frac{\Delta t}{\Delta x}(U_{j+1}^n - U_{j-1}^n) + U_j^{n-1} \\ U_j^n \end{pmatrix},$$

then

$$V_j^{n+1} = \begin{pmatrix} \frac{\Delta t}{\Delta x}(E - E^{-1}) & 1 \\ 1 & 0 \end{pmatrix} V_j^n$$

Example 7.3. *For the same equation $u_t = u_x$, consider the scheme:*

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

This can be written in terms of the shift operator E in the following way:

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

For implicit schemes like the example above, to find U_j^{n+1}, we need to solve a globally coupled linear sytem. We shall assume that the operator B_0^{-1} exists and it is bounded, so the finite difference scheme (7.3) can always be written in matrix form as:

$$V^{n+1} = C(\Delta t, \Delta x, \bar{x}, t)V^n$$

where we are taking all the components $\{V_\alpha^n : \alpha = (j_1, ..., j_x); j_i = 0, ..., N_i\}$ ordered to form a vector (e.g., the $vec(X)$ operation in Chapter 2); $\Delta x = (\Delta x_1, \cdots, \Delta x_s)$, and $\bar{x} = \{x_{j_i} : j_i = 0, ..., N_i, i = 1, ..., s\}$. We will assume that $\Delta x_i = h_i(\Delta t)$ for some functions h_i of the parameter Δt, for all space coordinates $i = 0, ..., s$. If the operator $\mathcal{P}$ in (7.1) does not depend on time,

then it is reasonable to limit our study to the case where C does not depend on time either. We will refer to this situation as the autonomous equation. If C does not depend on $\bar{x}$ either, we call the scheme a constant coefficient scheme, which we shall study in detail later. For the remainder of this chapter we shall simply write $C(\Delta t)$, keeping in mind that it may depend on t and $\bar{x}$ as well. We will therefore analyze the finite difference scheme (7.3) in its equivalent form:

$$V^{n+1} = C(\Delta t)V^n \tag{7.4}$$

where now $C(\Delta t)$ is an $N \times N$ matrix, with:

$$N = \prod_{i=1}^{s}(N_i + 1).$$

Recall that, N_i depends on Δx_i which is a function of Δt, thus the dimension of the matrix $C(\Delta t)$ depends on Δt.

Definition 7.2. *Let $\Delta x = (\Delta x_1, \cdots, \Delta x_{N_s})$, then for any fixed real number $t \geq 0$, we define the operator $Q_{\Delta x}$ by:*

$$Q_{\Delta x}u(x,t) = \{u(x_{j_1}, \cdots, x_{j_s}, t), \quad j = 0, \cdots, N_i, i = 0, \cdots, s.\}$$

So given a function $u(x,t)$, $Q_{\Delta x}u(x,t)$ is a vector with $N = \prod_{i=1}^{s}(N_i + 1)$ components, each of them representing a vector (recall that $u(x,t) = (u_1(x,t), ..., u_p(x,t))$ is a vector of p components).

At any fixed time t, $Q_{\Delta x}$ is an operator which "looks" at the values that $u(x,t)$ attains at the space grid points. In some cases it is more appropriate to specify projection operators which assign some values between the grid points. The space where we "project" the solution $u(x,t)$ via $Q_{\Delta x}$ is the same space where we are to construct the numerical solution, in accordance with (7.4). We are interested in studying the behavior of the collection of vectors in (7.4) for "small" values of Δt. We shall assume that:

$$\lim_{\Delta t \to 0} \Delta x_i = \lim_{\Delta t \to 0} h_i(\Delta t) = 0.$$

We now want to give a precise meaning to the statement *as Δt becomes smaller, the numeical solution gets closer to the analytical solution at any given time $t = n\Delta t$ held fixed.* Specifically, we want to compare the limit of V_i^n as $\Delta t \to 0$ and $n \to \infty$ such that $t = n\Delta t$ is constant, with the corresponding limit of $Q_{\Delta x}u$. This involves the concept of norms on the euclidean space $\mathbb{R}^N$ when the dimensjon N grows as $\Delta t \to 0$.

Definition 7.3. *For any vector $V = (V_1, ..., V_N)$, we define the norm $|V|_N$ by:*

$$|V|_N^2 = \frac{1}{N}\sum_{j=1}^{N}|V_j|^2$$

where, if each component V, is itself a vector, $|V_j|$ denotes the usual vector norm.

By our notations, (7.4) can denote a "one-step" method if $V^n = U^n$ where U^n approximates u at t_n, or a "k-step" method if

$$V^n = \begin{pmatrix} U^n \\ U^{n-1} \\ \vdots \\ U^{n-k} \end{pmatrix}. \tag{7.5}$$

7.2 Properties of Finite Difference Schemes

Throughout this section, we shall consider $u(x,t)$ to be the solution of a well posed initial value problem. That is, calling $S(t,t_0)$ the solution operator, the function u is specified by:

$$u(x,t) = S(t,t_0)u(x,t_0),$$

thus in particular:

$$u(x,(n+1)\Delta t) = S((n+1)\Delta t, n\Delta t)u(x,n\Delta t).$$

If the problem is *autonomous*,that is, the operator $\mathcal{P}$ in (7.1) is independent of time, then S is a function of the elapsed $(t-t_0)$ and we can simply write $S(t-t_0)$ and $S(\Delta t)$ in the above expressions.

Definition 7.4. *We say that the scheme $U^{n+1} = C(\Delta t)U^n$ is accurate of degree (or order) q_1 in space and q_2 in time, or more shortly, accurate (or consistent) of order (q_1,q_2) if for any fixed $t = n\Delta t$ and a very smooth solution $u(x,t)$:*

$$\left|\left[C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(t+\Delta t,t)\right]u(x,t)\right|_N \le K(t)\Delta t(|\Delta x|^{q_1} + \Delta t^{q_2}) \tag{7.6}$$

where

$$|\Delta x| = \sqrt{\sum_{i=1}^{s}(\Delta x_i)^2}.$$

If the system is autonomous, we can write (7.6) *in the form*

$$\left|\left[C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)\right]u(x,t)\right|_N \le K(t)\Delta t(|\Delta x|^{q_1} + \Delta t^{q_2}).$$

For a k-step method (7.4) *with* (7.5)*, accuracy of order (q_1,q_2) means*

$$\left|\left[C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)\right]\begin{pmatrix} u(x,t) \\ u(x,t-\Delta t) \\ \vdots \\ u(x,t-k\Delta t) \end{pmatrix}\right|_N \le K(t)\Delta t(|\Delta x|^{q_1} + \Delta t^{q_2}). \tag{7.7}$$

In most of the cases, it is desirable to have the same degree of accuracy in space and time, $q_1 = q_2 = q$, and when this happens, if no confusion arises, we will say that the scheme is accurate of degree q, or q-th order accurate. In some situations, however, we will work with accurate schemes for which $q_1 \neq q_2$.

This definition of accuracy (7.7) is simply an abstract description of the following local truncation error.

Definition 7.5. *Rewrite the scheme $V^{n+1} = C(\Delta t)V^n$ for solving 7.1 in the form approaching $u_t(x,t) - \mathcal{P}\left(x, t, \frac{\partial}{\partial x}\right) u(x,t) = 0$ as $\Delta t \to 0$ and $\Delta x \to 0$. The local truncation error is the residue of replacing numerical solutions by a smooth exact solution in the scheme of this form. The scheme is accurate of degree (or order) q_1 in space and q_2 in time if the local truncation error is equal to $\mathcal{O}(|\Delta x|^{q_1}) + \mathcal{O}(\Delta t^{q_2})$.*

We give now some examples of different schemes for the problem $u_t = u_x$.
Scheme 1:

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

This scheme is useless since it will never be stable as we have seen in Example 6.2. Nonetheless let us consider its accuracy. We denote by u_j^n, the true solution at the grid points: $u_j^n = u(j\Delta x, n\Delta t)$. Then

$$[C(\Delta t)Q_{\Delta x}u(x, n\Delta t)]_j = u_j^n + \frac{\Delta t}{2\Delta x}(u_{j+1}^n - u_{j-1}^n)$$

is the j-th component of the vector $C(\Delta t)Q_{\Delta x}u(x, n\Delta t)$. By definition,

$$S(\Delta t)u(x, n\Delta t) = u(x, (n+1)\Delta t),$$

and therefore:

$$[Q_{\Delta x}S(\Delta t)u(x, n\Delta t)]_j = u_j^{n+1}.$$

By the Taylor's expansion around (x_j, t_n), and the fact $u_t = u_x$, we get

$$\begin{aligned}
&|C(\Delta t)Q_{\Delta x}u(x, n\Delta t) - Q_{\Delta x}S(\Delta t)u(x, n\Delta t)|_j \\
=&|u_j^{n+1} - u_j^n - \frac{\Delta t}{2\Delta x}(u_{j+1}^n - u_{j-1}^n)| \\
=&|\Delta t(u_t)_j^n + \frac{1}{2}\Delta t^2(u_{tt})_j^n - \frac{\Delta t}{2\Delta x}(2\Delta x(u_x)_j^n + 2\frac{1}{6}\Delta x^3(u_{xxx})_j^n)| \\
=&|\frac{1}{2}\Delta t^2(u_{tt})_j^n - \frac{1}{6}\Delta t\Delta x^2(u_{xxx})_j^n| \\
\leq&\frac{1}{2}\max\left\{\max_x |u_{tt}(x, n\Delta t)|, \frac{1}{3}\max_x |u_{xxx}(x, n\Delta t)|\right\}\Delta t(\Delta t + \Delta x^2).
\end{aligned}$$

Assume there is a very smooth solution $u(x,t)$ s.t.

$$\max\left\{\max_x |u_{tt}(x, n\Delta t)|, \frac{1}{3}\max_x |u_{xxx}(x, n\Delta t)|\right\} \leq K,$$

then the scheme is accurate of order $(2,1)$. The scheme can be rewritten in the form approaching $u_t - u_x = 0$:

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

As an alternative way to check accuracy, we can compute the local truncation error as:

$$\tau_j^n = \frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = u_t(x_j, t_n) + \mathcal{O}(\Delta t) - u_x(x_j, t_n) + \mathcal{O}(\Delta x^2)$$

and using now the equation $u_t = u_x$, satisfied by $u(x,t)$, we conclude that this scheme is accurate of second order in space and first order in time.
Scheme 2: Lax-Friedrich's Scheme:

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

This scheme is a first order accurate scheme, that is, $q_1 = q_2 = 1$.
Scheme 3: Upwind Scheme: Consider the one-sided difference for the spatial derivative:

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

This is a first order accurate scheme.
Scheme 4: Downwind Scheme: Consider the one-sided difference for the spatial derivative:

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

This is a first order accurate scheme. However this scheme is also useless. The domain of dependence (the exact solution is a wave travelling to the left thus $u(x_j, t_n + \Delta t) = u(x_j - \Delta t, t_n)$ thus U_j^{n+1} depends on values of U^n to the right of x_j) is not included in the scheme stencil (U_j^{n+1} is based on U_j^n and U_{j-1}^n) therefore such a scheme is unstable.
Scheme 5: Leapfrog Scheme: If we use the centered difference for both time and spatial derivatives, we get

$$U_j^{n+1} = U_j^{n-1} + \frac{\Delta t}{\Delta x}(U_{j+1}^n - U_{j-1}^n). \tag{7.8}$$

To find its accuracy, rewrite it as

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

and compute

$$\tau_j^n = \frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} - \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = u_t + \mathcal{O}(\Delta t^2) - u_x + \mathcal{O}(\Delta x^2).$$

So this is a second order accurate scheme. It should be noticed that in order to implement this scheme it is not enough to specify initial conditions U^0 since it involves two time steps. To obtain U^1 for initiate the computation, there are many different ways. For instance, we can use a one-step method to approximate U^1.

Scheme 5: Lax-Wendroff Scheme: This scheme was developed around 1960 - 1964 and it is very frequently used. It is based on the Taylor series expansion for $u(x,t)$ given by:

$$u(x,t+\Delta t) = u(x,t) + \Delta t u_t(x,t) + \frac{1}{2}\Delta t^2 u_{tt}(x,t) + \mathcal{O}(\Delta t^3),$$

which, using $u_t = u_x$, reduces to:

$$u(x,t+\Delta t) = u(x,t) + \Delta t u_x(x,t) + \frac{1}{2}\Delta t^2 u_{xx}(x,t) + \mathcal{O}(\Delta t^3).$$

Using the centered difference, we obtain a scheme with second order accuracy in both time and space:

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

Scheme 5: Cranck-Nicholson Scheme: This is a second order accurate implicit scheme

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

The mere fact that a scheme is accurate does not imply that it provides useful results. Therefore we would like to compare the behavior of the numerical solution with the true solution, and not only the discrepancies resulting from one step of the time iterations. This comparison is the underlying concept of *convergence*.

Definition 7.6. *For a scheme $V^{n+1} = C(\Delta t)V^n$ in which U^n approximates u at $n\Delta t$, we say that the scheme converges if for arbitrary fixed $t > 0$ we have, for all n, Δt such that in $n\Delta t = t$:*

$$\lim_{\Delta t \to 0, \Delta x \to 0} |U^n - Q_{\Delta x} u(x, n\Delta t)|_N = 0.$$

Notice that the number N of points in the space grid becomes larger as $\Delta t \to 0$. If the initial condition of the original problem is $u(x,0) = f(x)$, then we can write:

$$u(x,t) = S(t)f(x) = S(t-t_1)S(t_1)f(x),$$

for any intermediate time $0 \leq t_1 \leq t$. In general we have:

$$u(x, n\Delta t) = S(\Delta t)^n f(x),$$

and analogously, a scheme in the form of $U^{n+1} = C(\Delta t)U^n$ can also be written as:

$$U^{n+1} = C(\Delta t)^n U^0$$

where $U^0 = Q_{\Delta x} f(x)$. In this notation, the convergence condition reads as follows:

$$\lim_{\Delta t \to 0, \Delta x \to 0} |(C(\Delta t)^n Q_{\Delta x} - Q_{\Delta x} S(\Delta t)^n) f(x)|_N = 0.$$

It should be clear now that convergence involves the difference between the values predicted by the numerical solution itself and those of the true solution "projected" at the grid points. On the other hand, to establish the accuracy of the scheme, we only need to check how the operator $C(\Delta t)$ changes the value of the true solution during only one time step, as compared to the true solution Δt units of time later. Convergence is the most important property of a numerical method. However, it cannot be established directly, since the true solution $u(x,t)$ is not known. We therefore look for ways to determine convergence indirectly, using only the partial differential equation and properties of the scheme that do not involve explicit knowledge of the function u that we want to approximate.

Definition 7.7. *We say that scheme $V^{n+1} = C(\Delta t)V^n$ is stable if for any fixed $t > 0$, there exist constants K and a such that:*

$$\|C(\Delta t)^n\| \leq K e^{\alpha n \Delta t},$$

for all n and Δt such that $n\Delta t = t$.. Here $\|C\|$ is the spectral norm for the matrix C.

Notice, first of all, that stability is indeed the discrete analog of well posedness. Recall from Chapter 5 that well posedness of the problem, in terms of the solution operator is equivalent to

$$|S(t,t_0)| \leq K e^{a(t-t_0)}, \quad \forall t \geq t_0,$$

where $|S|$ denotes the "operator norm". Therefore, for an autonomous system, where $S(n\Delta t) = S(\Delta t)^n$, we have:

$$|S(\Delta t)^n| \leq K e^{an\Delta t},$$

which is almost identical to the stability condition, except that here S is an operator acting on functions (generally belonging to some Hilbert space), and consequently the above norm refers to the corresponding operator norm, whereas in $|C(\Delta t)^n|$ is the matrix norm.

Theorem 7.1. Lax Equivalence Theorem. *Let $u(x,t)$ be a classical solution of the well posed linear problem* (7.1) *and let the finite difference scheme $V^{n+1} = C(\Delta t)V^n$ be accurate of order (q_1, q_2), i.e., the scheme satisfies* (7.6)*. If the scheme is stable, then for any T, there exists a bounded function $G(t)$ such that for all $t \in [0,T]$ and $n\Delta t = t$, the following holds*

$$|U^n - Q_{\Delta x}u(x, n\Delta t)|_N \leq G(t)(|\Delta x|^{q_1} + \Delta t^{q_2}).$$

Remark 7.2. *The theorem states not only the convergence but also the rate of convergence for a smooth solution of a wellposed initial value problem of any linear PDEs.*

Proof. For simplicity, we consider the one-step method $U^{n+1} = C(\Delta t)U^n$ and the extension to the k-step case is straightforward. Let

$$\delta^n = [C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)]u(x,t).$$

The actual error that we want to control to prove the convergence is

$$\begin{aligned}\varepsilon^{n+1} &= U^{n+1} - Q_{\Delta x}u(x,(n+1)\Delta t)\\ &= C(\Delta t)[U^n - Q_{\Delta x}u(x,n\Delta t)] + [C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)]u(x,t)\\ &= C(\Delta t)\varepsilon^n + \delta^n\end{aligned}$$

By solving $\varepsilon^{n+1} = C(\Delta t)\varepsilon^n + \delta^n$ and $\varepsilon^0 = 0$ (because we have $U^0 = Q_{\Delta x}u(x,0)$), we get

$$\varepsilon^n = \sum_{k=0}^{n-1} C(\Delta t)^{n-k-1}\delta^k,$$

thus

$$|\varepsilon^n|_N \leq \sum_{k=0}^{n-1} \|C(\Delta t)^{n-k-1}\||\delta^k|_N.$$

By stability, $\|C(\Delta t)^{n-k-1}\| \leq Ke^{a(n-k-1)\Delta t} \leq K_1$ for some constant K_1 and all k s.t. $0 \leq k \leq n-1$, and using accuracy on $|\delta^n|_N$:

$$|[C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)]\,u(x,t)|_N \leq K(t)\Delta t(|\Delta x|^{q_1} + \Delta t^{q_2}),$$

we get

$$|\varepsilon^n|_N \leq K_1 n\Delta t K(n\Delta t)(|\Delta x|^{q_1} + \Delta t^{q_2}),$$

which is the desired result, upon letting $G(t) = K_1 t K(t)$. □

Remark 7.3. *The Lax Equivalence Theorem applies to any linear scheme (for solving a linear PDE) in the form of $V^{n+1} = C(\Delta t)V^n$. For instance, in a finite element method in the form $U^{n+1} = C(\Delta t)U^n$ solving (7.1), U^n denotes the finite element basis coefficients (in contrast to point values in a finite difference method) and $Q_{\Delta x}u(x,t)$ denotes the projection of the exact solution onto the finite element space, then the same proof is still valid.*

In the Lax Equivalence Theorem, the assumption that $u(x,t)$ is a classical solution amounts to assume that the initial data $f(x)$ is a function with r continuous derivatives - where r is the degree of the polynomial $\mathcal{P}$ and with compact support, which we denote by $f(x) \in C_0^r$. Indeed, the assumption $f(x) \in C_0^r$ together with well posedness is equivalent to stating that $u(x,t)$ is a classical solution. However, the theorem can be generalized for the case where the initial function is not in C_0^r, provided that we can approximate this function in the L^2-sense by functions in C_0^r. To see this, assume that there exists a sequence $f_l(x) \in C_0^r, l = 1, 2, \cdots$ satisfying $\lim_{l\to+\infty} \|f - f_l\|^2 = 0$ where the norm is the L^2 norm (for example, $f(x)$ can be a step function multiplying a Gaussian, then $f(x)$ is not even continuous but can be approximated by a sequence of functions in C_0^r). Let $S(t - t_0)$ be the solution operator for the problem and define the sequence $u_l(x,t)$ as the corresponding solution with initial value $f_l(x)$, that is:

$$u_l(x,t) = S(t)f_l(x).$$

Since for any given $t \geq 0$, $S(t)$ is a bounded operator on L^2, it follows by convergence of f_l to f that the sequence of functions $u_l(x,t)$ for fixed t, converges in L^2 to some limit function $u(x,t)$ (which, however, may lack smoothness). Using the Lax Equivalence Theorem for each integer l we have:

$$|U_l^n - Q_{\Delta x}u_l(x, n\Delta t)|_N \leq G_l(t)(|\Delta x|^{q_1} + \Delta t^{q_2}), \tag{7.9}$$

where the U_l^n are defined for each l using scheme $U^{n+1} = C(\Delta t)U^n$ with initial value $U^0 = Q_{\Delta x}f_l(x)$. Therefore:

$$|U_l^n - U_m^n|_N \leq \|C(\Delta t)^n\||Q_{\Delta x}(f_l - f_m)|_N.$$

As a consequence of the L^2 convergence of f_l, it follows that for sufficiently large N, $|Q_{\Delta x}(f_l - f_m)|_N \to 0$ as $l, m \to \infty$, implying that the sequences $\{U_l^n : l \geq 1\}$ are Cauchy sequences for each n. This ensures the existence of the limiting vectors:

$$U^n = \lim_{l\to\infty} U_l^n \quad \text{for each} \quad n \geq 1.$$

Now to prove convergence we express the difference:

$$\begin{aligned} |Q_{\Delta x}u(x, n\Delta t) - U^n|_N &= |Q_{\Delta x}[u(x, n\Delta t) - u_l(x, n\Delta t)] + Q_{\Delta x}u_l(x, n\Delta t) - U_l^n + (U_l^n - U^n)|_N \\ &\leq |Q_{\Delta x}(u - u_l)|_N + |Q_{\Delta x}u_l - U_l^n|_N + |U_l^n - U^n|_N \end{aligned}$$

The first and third terms of this last inequality tend to zero as l increases, due to the definitions of u and U^n. The middle term satisfies (7.9), so all these facts together yield the convergence result for more general initial conditions. Notice that we loose information on the rate of convergence, since we do not know how the functions $G_l(t)$ in (7.9) behave with increasing l. Even if we know the rates for each l, the above inequality involves two limit processes.

Theorem 7.2. Kreiss Perturbation Theorem. *Suppose that the scheme:*

$$V^{n+1} = C(\Delta t)V^n$$

is stable. Then the perturbed scheme:

$$V^{n+1} = [C(\Delta t) + \Delta t D(\Delta t)]V^n$$

is stable, provided that $|D(\Delta t)| \leq H$, for some constant $H \geq 0$.

Before we give the proof of Kreiss perturbation theorein, we shall illustrate its usefulness.

Example 7.4. *Consider the partial differential equation:*

$$u_t = u_x - \beta u$$

and the scheme:

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

The Kreiss Perturbation Theorem states that it is enough to check stability for the leapfrog scheme (7.8)*, since $D(t) = -2\beta I$.*

Proof. We will just prove the "one-step" version, i.e., the case when $V^n = U^n$. The multi-step case is similar. Define the vectors W^n by the transformation:

$$W^n = e^{-n\Delta t\beta}U^n$$

where $\beta > 0$ is a constant to be determined later. The perturbed scheme becomes

$$W^{n+1} = e^{-(n+1)\Delta t\beta}[C(\Delta t)+\Delta t D(\Delta t)]e^{n\Delta\beta}W^n = e^{-\Delta t\beta}[C(\Delta t)+\Delta t D(\Delta t)]W^n,$$

so we get:

$$W^{n+1} = e^{-\Delta t\beta}C(\Delta t)W^n + \Delta t\bar{D}(\Delta t)W^n$$

where $\bar{D}(\Delta t) = e^{-\Delta t\beta}D(\Delta t)$. Let $\delta_n = \Delta t\bar{D}(\Delta t)W^n$, then the analog of the Duhamel principle for the finite difference equation:

$$W^{n+1} = e^{-\Delta t\beta}C(\Delta t)W^n + \delta_n$$

is given by

$$W^n = [e^{-\Delta t\beta}C(\Delta t)]^n W^0 + \sum_{k=0}^{n-1}[e^{-\Delta t\beta}C(\Delta t)]^{n-k-1}\delta_k$$

thus

$$W^n = [e^{-\Delta t\beta}C(\Delta t)]^n W^0 + \sum_{k=0}^{n-1}[e^{-\Delta t\beta}C(\Delta t)]^{n-k-1}\Delta t\bar{D}(\Delta t)W^k$$

By stability of $C(\Delta t)$ and boundedness of $D(\Delta t)$, there is a constant C_1 such that:

$$|C(\Delta t)^{n-k-1}||D(\Delta t)| \leq C_1$$

for all integers k with $0 \leq k \leq n-1$. Thus:

$$|W^n|_N \leq |C(\Delta t)|^n e^{-n\Delta t\beta}|W^0|_N + \left(C_1\Delta t\sum_{k=0}^{n-1} e^{-\Delta t\beta(n-k)}\right)\max_{0\leq k\leq n-1}|W^k|_N.$$

Let $z = e^{-\Delta t\beta}$, then $0 \leq z \leq 1$ and:

$$\sum_{k=0}^{n-1} e^{-\Delta t\beta(n-k)} = \sum_{k=0}^{n-1} e^{-\Delta t\beta k} = \sum_{k=0}^{n-1} z^k = \frac{1-z^n}{1-z} = \frac{1-e^{-n\Delta t\beta}}{1-e^{-\Delta t\beta}}$$

and since $1 - e^{-\Delta t\beta} \approx \beta\Delta t + \mathcal{O}(\Delta t^2)$, we may now pick β large enough so that the quantity:

$$C_1\Delta t\sum_{k=0}^{n-1} e^{-\Delta t\beta(n-k)} = C_1\frac{1-e^{-n\Delta t\beta}}{\beta + \mathcal{O}(\Delta t)}$$

is bounded by some constant $\gamma < 1/2$ for all integers n and all $\Delta t > 0$. Since $|C(\Delta t)|^n \leq C_2 e^{an\Delta t}$ for some constants C_2 and a, and using $U^0 = W^0$, it follows that:

$$|W^n|_N \leq C_2 e^{(a-\beta)n\Delta t}|U^0|_N + \gamma\max_{0\leq k\leq n-1}|W^k|_N.$$

We may assume that $a - \beta < 0$, for if this is not the case, we just increase the value of β and we will still have $\gamma < 1/2$. Then $e^{(a-\beta)\Delta t} \leq 1$ for all n, Δt, and:

$$|W^n|_N \leq C_2|U^0|_N + \gamma\max_{0\leq k\leq n-1} W|W^k|_N,$$

where C_2 does not depend on n. Now for any arbitrary large integer M, we take the maximum on both sides over $0 \leq n \leq M$:

$$\max_{0\leq n\leq M}|W^n|_N \leq C_2|U^0|_N + \max_{0\leq n\leq M}\gamma\max_{0\leq k\leq n-1}|W^k|_N,$$

thus

$$\max_{0\le n\le M}|W^n|_N \le C_2|U^0|_N + \gamma \max_{0\le n\le M}|W^k|_N.$$

Therefore,

$$(1-\gamma)|W^n|_N \le (1-\gamma)\max_{0\le n\le M}|W^n|_N \le C_2|U^0|_N,$$

for all $0 \le n \le M$. We finally get

$$|U^n|_N \le \frac{C_2}{1-\gamma}e^{\beta n\Delta t}|U^0|_N.$$

Stability follows from the fact that $U^n = C(\Delta t)^n U^0$. □

As can be deducted from the proof, it is extremely important that the perturbation be of the order Δt, more specifically, that it has the form $\Delta t D(\Delta t)$. In many practical situations one has to be careful in applying the result of this perturbation theorem, always checking first if the assumptions are indeed satisfied. The following is an example in which the perturbation has apparently the form $\Delta t D(\Delta t)$, yet it gives rise to an unstable scheme.

Example 7.5. *For the equation:*

$$u_t = u_{xx} + u_x$$

the term u_x is a lower order term. One possible scheme is the following:

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

The last term corresponds to the "perturbation" and we want to know whether we can neglect this term by applying Kreiss Perturbation Theorem, in order to check stability. Although it appears that the perturbation is of order Δt, this may not be the case, for if we choose $\Delta t/\Delta x^2$ constant to achieve stability of the unperturbed scheme, then the perturbation is really of order $\sqrt{\Delta t}$ and the theorem is not applicable in this case.

7.3 Basic definitions and notations for stability

Next we will consider the stability for constant coefficient schemes. As already mentioned, stability of a finite difference scheme is the discrete analog of the concept of well posedness of a partial differential equation. We present the basic results and tools that allow us establish the stability of a constant coefficient finite difference scheme. Examples are inserted throughout the rest of this chapter in order to introduce and illustrate the concepts involved in the problem.

Example 7.6. *Consider*

$$u_t(x,t) = u_x(x,t), \quad x \in [0, 2\pi],$$

and we assume 2π-periodicity of the solution. We construct a grid of points with constant spacing Δx in space and Δt in time, such that:

$$\frac{\Delta t}{\Delta x} = \lambda \leq 1; \quad x_j = j\Delta x, j = 0, \cdots, N-1, \Delta x = \frac{2\pi}{N}.$$

Let us focus on the upwind scheme:

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

with the appropriate boundary conditions given by the periodicity requirement:

$$U_{-1}^n = U_{N-1}^n, U_N^n = U_0^n,$$

which holds for all $n > 0$. This upwind scheme can be written in the matrix form as:

$$U^{n+1} = C(\Delta t)U^n$$

where each U^n is a vector of N components:

$$\begin{pmatrix} U_0^{n+1} \\ U_1^{n+1} \\ \vdots \\ U_{N-2}^{n+1} \\ U_{N-1}^{n+1} \end{pmatrix} = \begin{pmatrix} 1-\lambda & \lambda & \cdots & 0 & 0 \\ 0 & 1-\lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \vdots & 1-\lambda & \lambda \\ \lambda & 0 & \vdots & 0 & 1-\lambda \end{pmatrix} \begin{pmatrix} U_0^n \\ U_1^n \\ \vdots \\ U_{N-2}^n \\ U_{N-1}^n \end{pmatrix}.$$

The dimension of the matrix $C(\Delta t)$ depends on Δt itself, through the dependence of N on Δt. In general, this makes it hard to check directly the stability of the scheme, that is, to find constants K and α such that:

$$\|C(\Delta t)^n\| \leq Ke^{\alpha t},$$

for all n and Δt such that $t = n\Delta t$ is held fixed.

Before we analyze the stability, let us recall several matrix notations:

- A^T is the transpose of A. A^* is the complex conjugate transpose.
- Eigenvalues: $\text{eig}_i(S)$ denotes the eigenvalue of S with i-th largest magnitude.
- Jordan Normal Form: any matrix S can be decomposed as $S = P\Lambda P^{-1}$ where Λ is upper-triangular and the diagonal entries are eigenvalues of S.

- Singular Value: the i-th largest one is denoted as $\sigma_i(S) = \sqrt{\text{eig}_i(SS^*)} = \sqrt{\text{eig}_i(S^*S)}$.
- Normal Matrix: a matrix A is called normal if $AA^* = A^*A$.
- The following are equivalent:
  1. A is normal.
  2. A is diagonalizable by a unitary matrix, i.e., $A = P\Lambda P^*$, Λ is a diagonal matrix and $PP^* = I$.
  3. $\sigma_i(A) = |\,\text{eig}_i(A)|$.

For the particular case in Example 7.6, we can find $\|C(\Delta t)^n\|$ since $C(\Delta t)$ is circulant thus can be diagonalized by the DFT matrix, which is a unitary matrix. By multiplyting $C(\Delta t)$ to the vector obtained by sampling $e^{\mathbb{i}\,nx}$ at $x = (0, \Delta x, 2\Delta x, \cdots, (N-1)\Delta x)^T$, we can get the eigenvalues as $\lambda_k = 1 - \lambda + \lambda e^{\mathbb{i}\,k\Delta x}$. Let T be the DFT matrix then $C(\Delta t) = T\Lambda T^*$ where the diagonal matrix Λ has diagonal entries λ_k $(k = 0, 1, \cdots, N-1)$. Since $C(\Delta t)^n = T\Lambda^n T^*$ (so $C(\Delta t)^n$ is a normal matrix), thus $C(\Delta t)^n[C(\Delta t)^n]^* = T[\Lambda\Lambda^*]^n T^*$, we get the singular values of $C(\Delta t)^n$ as

$$|\lambda_k|^n = [(1-\lambda)^2 + \lambda^2 + 2\cos(k\Delta x)\lambda(1-\lambda)]^{\frac{n}{2}}.$$

Next we use an easier alternative method instead of looking directly at the matrix $C(\Delta t)$. In Example 7.6, we can consider the discrete Fourier transform (4.8) and the inverse discrete Fourier transform (4.9) for U_j^n. Assume N is even, we use a normalized (also index shifted) version of (4.8) and (4.9):

$$\hat{U}_k^n = \sum_{j=0}^{N-1} e^{-\mathbb{i}\,kj\Delta x} U_j^n, \quad k = 0, \cdots, N-1.$$

$$U_j^n = \frac{1}{N} \sum_{k=0}^{N-1} e^{\mathbb{i}\,kj\Delta x} \hat{U}_k^n, \quad j = 0, \cdots, N-1,$$

We also have the Parseval's identity for the discrete Fourier transform above:

$$\sum_{j=0}^{N-1} |U_j^n|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |\hat{U}_k^n|^2.$$

Replace U^n by its inverse discrete Fourier transform in the upwind scheme, we get

$$\frac{1}{N} \sum_{k=0}^{N-1} e^{\mathbb{i}\,kj\Delta x} \hat{U}_k^{n+1} = \frac{1}{N} \sum_{k=0}^{N-1} e^{\mathbb{i}\,kj\Delta x} \hat{U}_k^n + \lambda \Big(\frac{1}{N} \sum_{k=0}^{N-1} e^{\mathbb{i}\,k(j+1)\Delta x} \hat{U}_k^n - \frac{1}{N} \sum_{k=0}^{N-1} e^{\mathbb{i}\,kj\Delta x} \hat{U}_k^n \Big),$$

thus

$$\frac{1}{N}\sum_{k=0}^{N-1} e^{\mathbb{i}\, kj\Delta x}\left[\hat{U}_k^{n+1} - \hat{U}_k^n - \lambda(e^{\mathbb{i}\, k\Delta x}\hat{U}_k^n - \hat{U}_k^n)\right] = 0.$$

which means that the inverse discrete Fourier transform of $\hat{U}_k^{n+1} - \hat{U}_k^n - \lambda(e^{\mathbb{i}\, k\Delta x}\hat{U}_k^n - \hat{U}_k^n)$ is equal to zero. Therefore, we get

$$\hat{U}_k^{n+1} - \hat{U}_k^n - \lambda(e^{\mathbb{i}\, k\Delta x}\hat{U}_k^n - \hat{U}_k^n) = 0,$$

which can be written as

$$\hat{U}_k^{n+1} = g(k)\hat{U}_k^n, \quad g(k) = 1 + \lambda(e^{\mathbb{i}\, k\Delta x} - 1). \tag{7.10}$$

Then we have

$$\begin{aligned}\sum_{j=0}^{N-1} |U_j^{n+1}|^2 &= \frac{1}{N}\sum_{k=0}^{N-1} |\hat{U}_k^{n+1}|^2 \\ &= \frac{1}{N}\sum_{j=0}^{N-1} |g(k)|^2 |\hat{U}_k^n|^2 \\ &\leq \max_k |g(k)|^2 \sum_{j=0}^{N-1} |U_j^n|^2.\end{aligned}$$

Thus if $\max_k |g(k)|^2$ is bounded for all possible values of $k\Delta x$, we have a bound for $\|C(\Delta t)\|$, which yields stability. The main idea is therefore to study the functions $g(k)$ instead of working with the matirx $C(\Delta t)$, even though these two methods are essentially equivalent. Recall that we have used the DFT matrix to diagonalize the circulant matrix and the DFT matrix represents precisely the discrete Fourier transform. Notice that $g(k)$ are exactly the eigenvalues of $C(\Delta t)$. Notheless, the second method is easier, because we can obtain (7.10) simply by asserting an ansatz $U_j^n = \hat{U}_k^n e^{\mathbb{i}\, kx_j}$ into the scheme.

Example 7.7. *We consider now a generalization of the previous example. Let A be a constant $p \times p$ matrix and and $u(x,t) = (u_1(x,t), \cdots, u_p(x,t))^T$ satisfying:*

$$u_t = Au_x, \quad x \in [0, 2\pi],$$

and we also assume 2π-periodicity of $u(x,t)$. Consider a naive extension of the upwind scheme:

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

with boundary conditions

$$U_{-1}^n = U_{N-1}^n, U_N^n = U_0^n.$$

Now let us use the ansatz $U_j^n = \hat{U}_k^n e^{\mathbb{i}\, kj\Delta x}$, *which is equivalent to apply the discrete Fourier transform to the scheme or use the DFT matrix to "diagonalize"* $C(\Delta t)$. *We get*

$$\hat{U}_k^{n+1} e^{\mathbb{i}\, kj\Delta x} = \hat{U}_k^n e^{\mathbb{i}\, kj\Delta x} + \lambda A(\hat{U}_k^n e^{\mathbb{i}\, k(j+1)\Delta x} - \hat{U}_k^n e^{\mathbb{i}\, kj\Delta x}),$$

thus

$$\hat{U}_k^{n+1} = \mathcal{G}(k)\hat{U}_k^n, \quad \mathcal{G}(k) = I + \lambda A(e^{\mathbb{i}\, k\Delta x} - 1).$$

where $\mathcal{G}(k)$ *is a* $p \times p$ *matrix. Notice that* $C(\Delta t)$ *is a* $Np \times Np$ *matrix with* $N \to 0$ *while the size of* $\mathcal{G}(k)$ *is fixed.*

We now generalize the concepts introduced in the examples given above. As we recall from Chapter 5, the general form of a partial differential equation with constant coefficients is given by

$$\begin{aligned} u_t &= \mathcal{P}(\partial \,/\, \partial\, x)u, \\ u(x,0) &= f(x), \end{aligned} \tag{7.11}$$

where $u(x,t) = (u_1(x,t), \cdots, u_n(x,t))^T$ is a function of $x = (x_1, \cdots, x_s)$ and time t. In an analogous way, we can define a finite difference scheme with constant coefficients in general form:

Definition 7.8. *Let* Δt *and* Δx_i *be any given step sizes, for* $i = 1, \cdots, s$, *and denote by* $X = \{x_{j_i} : j_i = 0, \cdots, N_i; i = 1, \cdots, s\}$ *the collection of all grid points in the space coordinates. A scheme of the form:*

$$V^{n+1} = C(\Delta t, X, t)V^n$$

is called a constant coefficient scheme if the matrix $C(\Delta t, X, t)$ *does not depend on* X *and* t, *so we can write:*

$$V^{n+1} = C(\Delta t)V^n.$$

Consider the constant coefficient scheme:

$$U^{n+1} = C(\Delta t, X, t)U^n.$$

For each multiindex $j = (j_1, \cdots, j_i)$ with $j_i = 0, \cdots, N_i$ and $i = 1, \cdots, s$, U_j^n is a vector of p components approximating the value of the true solution at the grid points $u(j_1\Delta x_1, \cdots, j_s\Delta x_s, n\Delta t)$. The discrete Fourier transform is now given by

$$\hat{U}_k^n = \sum_{j \in \mathcal{J}} e^{-\,\mathbb{i}\langle k, x_j\rangle} U_j^n,$$

$$U_j^n = \frac{1}{N} \sum_{k \in \mathcal{K}} e^{\mathbb{i}\langle k, x_j\rangle} \hat{U}_k^n.$$

where $N = \Pi_{i=1}^{s} N_i$, $k = (k_1, \cdots, k_s)$ is a multi-index, the sets $\mathcal{J}$ and $\mathcal{K}$ are the set of multiindex j and k so that $0 \leq k_i \leq N_i - 1$ and $0 \leq j_i \leq N_i - 1$. Using the discrete Fourier transform of the scheme yields the difference equations in the Fourier space:

$$\hat{U}_k^{n+1} = \mathcal{G}(\Delta t, k)\hat{U}_k^n.$$

We call the matrix $\mathcal{G}(\Delta t, k)$ the **amplification matrix**. If the problem is scalar (p = 1), then we write $g(\Delta t, k)$ or sometimes just $g(k)$, and usually call it the *amplification factor*.

7.4 von Neumann stability

Stability of the scheme $V^{n+1} = C(\Delta t)V^n$ can be written in terms of the amplification matrix as the following condition: given $t > 0$, there exist constants K and α such that for all multi-index k and all n such that $n\Delta t = t$,

$$\|\mathcal{G}(\Delta t, k)^n\| \leq Ke^{\alpha t}.$$

The condition must be satisfied for all multi-index k in order to establish stability of the scheme. This condition involves an infinite number of matrices being uniformly bounded, yet in practice it turns out to be remarkably easier to deal with the amplification matrices treating k as a parameter, than it is to study stability working directly with $C(\Delta t)$, whose dimension depends on the chosen discretization of space and time. Our first result presents a necessary although not sufficient condition for stability.

Theorem 7.3. The von Neumann Condition *The amplification matrix of a stable scheme satisfies the condition:*

$$\rho[\mathcal{G}(\Delta t, k)] \leq e^{\gamma \Delta t} = 1 + \mathcal{O}(\Delta t),$$

where $\rho[\mathcal{G}(\Delta t, k)]$ denotes the spectral radius (largest magnitude of eigenvalues) of the matrix $\mathcal{G}(\Delta t, k)$.

The von Neumann stability condition is necessary but not sufficient for stability. In most practical applications, turns out to be easily checked whether this condition holds or not, as we shall examplify later on. When determining stability of a scheme, our first step shall always be verifying whether this condition holds or not.

Proof. If the scheme is stable, then

$$\|\mathcal{G}^n\| \leq Ke^{\alpha t},$$

where $t = n\Delta t$. We need a fact for the spectral radius

$$\rho(A)^n \leq \|A^n\|.$$

To see why this is true, let v and λ be eigenvectors and eigenvalues of A, then

$$|\lambda|^k\|v\| = \|\lambda^k v\| = \|A^k v\| \leq \|A^k\| \cdot \|v\| \Rightarrow |\lambda|^k \leq \|A^k\|.$$

So we have

$$\rho[\mathcal{G}(\Delta t, k)] \leq \|\mathcal{G}^n\|^{\frac{1}{n}} \leq K^{\frac{1}{n}} e^{\alpha \Delta t}.$$

Since $t = n\Delta t$ is held fixed at a constant value, $K^{\frac{1}{n}} = K^{\frac{\Delta t}{t}}$. Let $\beta = \log K$, then

$$\rho(\mathcal{G}) \leq e^{\beta \Delta t/t} e^{\alpha \Delta t} = e^{(\beta/t+\alpha)\Delta t} = e^{\gamma \Delta t}$$

where $\gamma = \beta/t + \alpha$ is a positive constant for all n and Δt such that $t = n\Delta t$ is constant, yielding the von Neumann condition. □

Remark 7.4. *The von Neumann condition is also sufficient for stability in the following two cases:*

- *If $\mathcal{G}$ is a normal matrix (the scalar case $\mathcal{G} = g$ is a special case), then so is $\mathcal{G}^n$ thus $\|\mathcal{G}^n\| = \rho[\mathcal{G}^n]$.*
- *If $\mathcal{G}$ is diagonalizable $\mathcal{G}(\Delta t, k) = T\Lambda T^{-1}$ with $\|T\|\|T^{-1}\| \leq K$ for all Δt and k, then $\mathcal{G}^n = T\Lambda^n T^{-1}$ thus $\|\mathcal{G}\| \leq \|T\|\|\Lambda^n\|\|T^{-1}\| = \|T\|\rho[\mathcal{G}^n]\|T^{-1}\|$.*

7.5 The leapfrog scheme

7.5.1 The one way wave equation

In this section we first study in detail the leap frog scheme for the one dimensional scalar equation $u_t = u_x$ to understand the stability we have defined for finite difference schemes. The scheme is:

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

and we impose the periodic boundary conditions through the usual periodicity requirement:

$$U_{-1}^n = U_{N-1}^n, \quad U_0^n = U_N^n$$

Define the vector

$$V_j^n = \begin{pmatrix} U_j^n \\ U_j^{n-1} \end{pmatrix},$$

then we can rewrite the scheme into the form $V^{n+1} = C(\Delta t)V^n$. Let $\lambda = \frac{\Delta t}{\Delta x}$, then the scheme becomes

$$V_j^{n+1} = \begin{pmatrix} \lambda(E - E^{-1}) & 1 \\ 1 & 0 \end{pmatrix} V_j^n,$$

where E and E^{-1} are the shift operations, as introduced in Section 7.1. Therefore, although the original problem is a scalar one, here V is a 2-component vector: we have considered a "fictitious" component to be able to represent the scheme by $V^{n+1} = C(\Delta t)V^n$. Plugging in the ansatz $V_j^n = \hat{V}_k^n e^{\mathbb{i}\, kj\Delta x}$ (which is equivalent to plugging in the discrete Fourier transform of V^n), we get

$$\hat{V}_k^{n+1} e^{\mathbb{i}\, kj\Delta x} = \begin{pmatrix} \lambda(E - E^{-1}) & 1 \\ 1 & 0 \end{pmatrix} \hat{V}_k^n e^{\mathbb{i}\, kj\Delta x}.$$

Notice that the shift operators act only on the functions of $x_j = j\Delta x$, we have

$$E\hat{e}^{\mathbb{i}\, kj\Delta x} V_k^n = e^{\mathbb{i}\, k\Delta x} e^{\mathbb{i}\, kj\Delta x} \hat{V}_k^n,$$
$$E^{-1}\hat{e}^{\mathbb{i}\, kj\Delta x} V_k^n = e^{-\mathbb{i}\, k\Delta x} e^{\mathbb{i}\, kj\Delta x} \hat{V}_k^n,$$

thus

$$\hat{V}_k^{n+1} = e^{-\mathbb{i}\, kj\Delta x} \begin{pmatrix} \lambda(e^{\mathbb{i}\, k\Delta x} - e^{-\mathbb{i}\, k\Delta x}) & 1 \\ 1 & 0 \end{pmatrix} e^{\mathbb{i}\, kj\Delta x} \hat{V}_k^n = \begin{pmatrix} 2\,\mathbb{i}\,\lambda \sin(k\Delta x) & 1 \\ 1 & 0 \end{pmatrix} \hat{V}_k^n.$$

Therefore we have the explicit expression for the amplification matrix:

$$\mathcal{G}(\Delta x, k) = \begin{pmatrix} 2\,\mathbb{i}\,\lambda \sin(k\Delta x) & 1 \\ 1 & 0 \end{pmatrix}$$

The variable $k\Delta x$ appears in the expression of the amplilication matrix as the argument of a trigonometric function. This is in general true, and in order to analyze the amplification matrix in terms of its arguments, it is enough to consider the variable $\xi = k\Delta x$ restricted to $0 \leq \xi < 2\pi$. Throughout the rest of this text, we shall often write $\xi = k\Delta x$ without further mentioning that we actually consider ξ to be restricted to the interval $[0, 2\pi)$.

The eigenvalues of the amplification matrix $\mathcal{G}(\Delta x, k)$ can be calculated as:

$$\mu_1(\xi) = \mathbb{i}\,\lambda \sin\xi + \sqrt{1 - \lambda^2 \sin^2\xi},$$
$$\mu_1(\xi) = \mathbb{i}\,\lambda \sin\xi - \sqrt{1 - \lambda^2 \sin^2\xi},$$

We will check now the von Neumann condition as well as the conditions for stability of the leap frog scheme under study.

Case I: If $\lambda^2 > 1$, then for those values of k such that $\xi = k\Delta x = \frac{pi}{2}$ we have:

$$\mu_1(\pi/2) = \mathbb{i}(\lambda + \sqrt{\lambda^2 - 1}),$$

so $|\mu_1(\pi/2)| > 1$, yielding that the von Neumann stability condition is not satisfied by the amplification matrix. We conclude that the leap frog scheme is unstable when $\lambda > 1$.

Case II: If $\lambda^2 \leq 1$, then

$$|\mu_i(\xi)|^2 = \lambda^2 \sin^2 \xi + 1 - \lambda^2 \sin^2 \xi = 1,$$

for $i = 1, 2$ which holds for any value of ξ. Therefore $\rho[\mathcal{G}] = 1$ and the von Neumann condition is satisfied. Nonetheless, this does not imply that the scheme is stable for $\lambda \leq 1$. Indeed we will show that the scheme is actually unstable for $\lambda = 1$.

To see this, recall that stability requires that the family of matrices $\mathcal{G}(\Delta, k)$ be uniformly bounded by $Ke^{\alpha t}$ for all values of k. Consider now $\lambda = \Delta t/\Delta x = 1$, then for all n with $n\Delta t$ fixed, stability would certainly imply the uniform bound in $\|\mathcal{G}^n(\Delta, k)\|$ as $n \to \infty$ for all possible values of k. Notice that $\lambda = 1$ is also fixed. In order to prove our claim that this case is unstable, it suffices to show that for one particular value of $\xi = k\Delta x$, $\|\mathcal{G}^n(\Delta, k)\|$ is not bounded as $n \to \infty$. Let $\xi = \pi/2$ and k_0 denote the modes for which $k_0\Delta x = \frac{\pi}{2}$(modulo 2π), then

$$\mathcal{G}(\Delta t, k_0) = \begin{pmatrix} 2\,\mathbb{i} & 1 \\ 1 & 0 \end{pmatrix}.$$

Notice that $\mathcal{G}(\Delta t, k_0)$ has one repeated eigenvalue $\mu_1 = \mu_2 = \mathbb{i}$ and it is not diagonalizable (because the eigenspace is one-dimensional). Let v_1 the be one eigenvector and v_2 be one generalized eigenvector. Let $T = [v_1, v_2]$, then the Jordan form of this matrix can be written as

$$\mathcal{G}(\Delta t, k_0) = T \begin{pmatrix} \mathbb{i} & 1 \\ 0 & \mathbb{i} \end{pmatrix} T^{-1}.$$

Therefore,

$$\mathcal{G}^n(\Delta t, k_0) = T \begin{pmatrix} \mathbb{i} & 1 \\ 0 & \mathbb{i} \end{pmatrix}^n T^{-1} = T \begin{pmatrix} \mathbb{i}^n & n\,\mathbb{i}^{n-1} \\ 0 & \mathbb{i}^n \end{pmatrix} T^{-1}.$$

Obviously $\left\| \begin{pmatrix} \mathbb{i}^n & n\,\mathbb{i}^{n-1} \\ 0 & \mathbb{i}^n \end{pmatrix} \right\| \to \infty$ as $n \to \infty$, thus $\|\mathcal{G}^n(\Delta t, k_0)\| \to \infty$ as $n \to \infty$. Therefore, the leap frog scheme is unstable for $\lambda = 1$, although the von Neumann condition is satisfied.

Lemma 7.1. *The leap frog scheme for $u_t = u_x$ is stable for $\lambda < 1$.*

Proof. Let $\lambda < 1$. Then the two eigenvalues $\mu_1(\xi)$ and $\mu_2(\xi)$ are distinct thus $\mathcal{G}$ is diagonalizable. Let T be the eigenvector matrix then we have

$$\mathcal{G} = T \begin{pmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{pmatrix} T^{-1},$$

thus

$$\mathcal{G}^n = T \begin{pmatrix} \mu_1^n & 0 \\ 0 & \mu_2^n \end{pmatrix} T^{-1},$$

and

$$\|\mathcal{G}^n\| \leq = \|T\| \left\| \begin{pmatrix} \mu_1^n & 0 \\ 0 & \mu_2^n \end{pmatrix} \right\| \|T^{-1}\|$$

The spectral norm of $\begin{pmatrix} \mu_1^n & 0 \\ 0 & \mu_2^n \end{pmatrix}$ is equal to $\max_i |\mu_i|^n = 1$ (recall that $\|\mu_i\| = 1$) because we have (the singular values of A are square roots of eigenvalues of AA^*)

$$\begin{pmatrix} \mu_1^n & 0 \\ 0 & \mu_2^n \end{pmatrix} \begin{pmatrix} \bar{\mu}_1^n & 0 \\ 0 & \bar{\mu}_2^n \end{pmatrix} = \begin{pmatrix} |\mu_1|^{2n} & 0 \\ 0 & |\mu_2|^{2n} \end{pmatrix}.$$

Therefore $\|\mathcal{G}^n\| \leq \|T\|\|T^{-1}\|$. To conclude the uniform boundedness of $\|\mathcal{G}^n\|$ as $n \to \infty$, we still need to show $\|T\|\|T^{-1}\|$ are bounded as $n \to \infty$. This is true since T depends on only ξ and λ. The eigenvectors of $\mathcal{G}$ can be explicitly computed. For instance, we can take

$$T = \begin{pmatrix} \mu_1 & \mu_2 \\ 1 & 1 \end{pmatrix}, \quad T^{-1} = \frac{1}{\mu_1 - \mu_2} \begin{pmatrix} 1 & -\mu_2 \\ -1 & \mu_1 \end{pmatrix}.$$

We have

$$T^*T = \begin{pmatrix} 2 & \mu_1^*\mu_2 + 1 \\ \mu_1\mu_2^* + 1 & 2 \end{pmatrix} = \begin{pmatrix} 2 & -(\mu_1 + \mu_2)\mu_2 \\ -(\mu_1 + \mu_2)\mu_1 & 2 \end{pmatrix}$$

whose eigenvalues are bounded at least by 4, yielding $\|T\| \leq 2$. Similarly,

$$(T^{-1})^*T^{-1} = \frac{1}{(\mu_1 - \mu_2)^2} \begin{pmatrix} 2 & -(\mu_1 + \mu_2) \\ 0 & \mu_1(\mu_1 - \mu_2) \end{pmatrix},$$

whose eigenvalues are also bounded. Indeed, since $\frac{1}{(\mu_1-\mu_2)^2} \leq \frac{1}{4(1-\lambda^2)}$, we have $\|T^{-1}\|^2 \leq \frac{C}{1-\lambda^2}$ for some constant C.

This implies that $\|\mathcal{G}^n(\Delta t, k)\|$ is bounded for all values of Δt, k and n such that $t = n\Delta t$ is held fixed, which proves the assertion. □

Example 7.8. *Now we use a numerical example to understand the stability condition $\lambda < 1$ that we have just derived. Consider using the leapfrog scheme to solve $u_t = u_x$ with periodic boundary conditions on the inverval $x \in [-1, 1]$ and initial condition $f(x) = \frac{1}{a}e^{-x^2/a^2}$ with $a = 0.02$.*

First we consider the exact initial conditions, i.e, suppose we take $U^0 = f(x)$ and $U^1 = f(x+\Delta t)$. See Figure 7.1 for three cases $\frac{\Delta t}{\Delta x} = \lambda = 0.9, \lambda = 1$, and $\lambda = 1.1$ at time $t = 0.8$. We can observe that:

1. *The numerical solution blows up for $\lambda = 1.1$, as expected.*

2. *The case $\lambda = 1$ gives the best solution. This is actually not a surprise because the numerical stencil happens to coincide with the characteristic lines of $u_t = u_x$. In other words, the numerical scheme produces the exact solution in this case. For instance, the exact solution at $(x_j, 2\Delta t)$ is $f(x_j + 2\Delta t)$, while the leapfrog scheme gives $U_j^2 = U_j^0 + (U_{j+1}^1 - U_{j-1}^1) = U_{j+1}^1 = f(x_{j+1} + \Delta t) = f(x_j + 2\Delta t)$, where we use facts that $U_j^0 = U_{j-1}^1$ and $U_{j+1}^1 = f(x_{j+1} + \Delta t)$ (both are due to exact initial conditions).*

3. *There are some oscillations in the case $\lambda = 0.9$. There is nothing contradictory to the stability $\|\mathcal{G}^n\| \leq Ke^{\alpha t}$ because this stability is 0-stability, similar to what we defined for ODE solvers in Chapter 6. These oscillations imply the error at this specific grid is large. On the other hand, if we refine the mesh ($\Delta x \to 0$), these errors will go away in a second order rate since this is a smooth solution. In other words, the oscillations in Figure 7.1 (d) are accuracy issues rather than stability issues.*

It is counterintuitive that an unstable scheme $\lambda = 1$ can produce a very nice solution. Actually it produces the exact one, which cannot be better. However, we have used the exact initial conditions. Now let us see what will happen if using inexact initial conditions to initiate the leapfrog scheme. We consider the following consistent perturbed initial conditions:

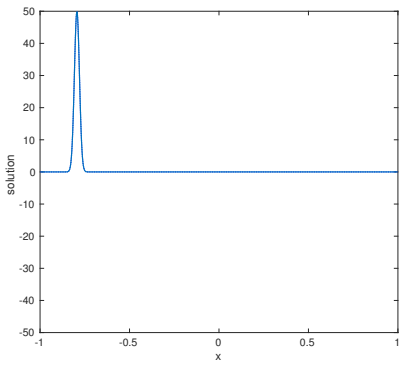
```
1  U_0=f(x);
2  U_1=f(x+dt)+dt*10*randn(size(x));
```

See Figure 7.2 for numerical solutions of $\lambda = 0.9$ and $\lambda = 1$ at a longer time $t = 2.8$. We can see that both stable and unstable schemes produce oscillations. However, the oscillations reduce when we refine the mesh in the stable scheme ($\lambda = 0.9$) while the oscillations increase when we refine the mesh in the unstable scheme ($\lambda = 1$). This is preciously what will happen for unstable schemes: we lose convergence (as $\Delta x \to 0, \Delta t \to 0$).

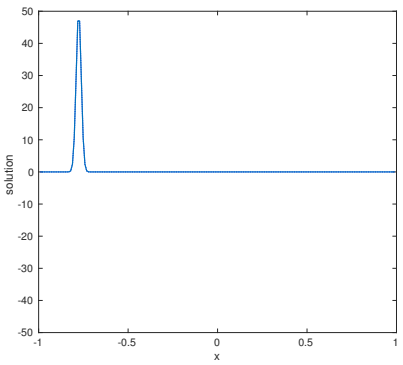
We have two interesting observations in this example:

- *An unstable scheme does not necessarily produce blow-ups. It is not enough to assert a scheme designed/implemented is stable if we only see the numerical solution on a coarse grid fits the reference solution well. It is necessary to validate the convergence by refining the mesh. For a linear problem, if there is no convergence (error stops to decrease when refining meshes), then there is no stability.*

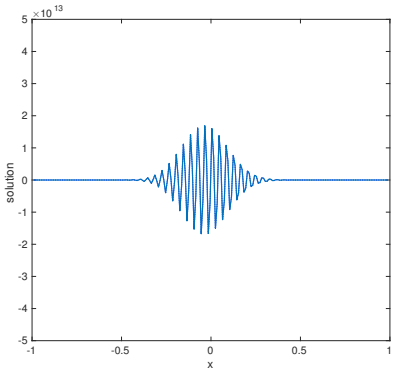
- *On some grid, an unstable scheme may produce better solutions, which does not imply any of its usefulness though.*



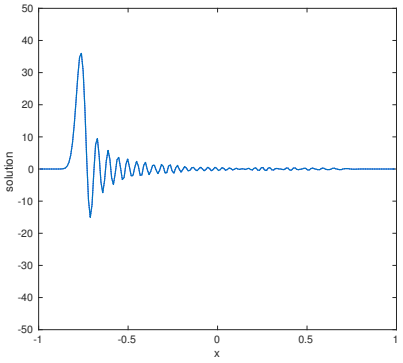
(a) Reference Solution.



(b) $\lambda = 1$ with exact initial conditions on 200 grid points.

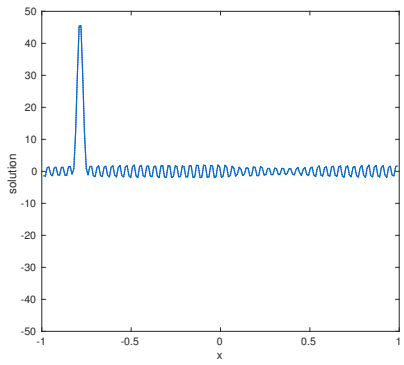


(c) $\lambda = 1.1$ with exact initial conditions on 200 grid points.

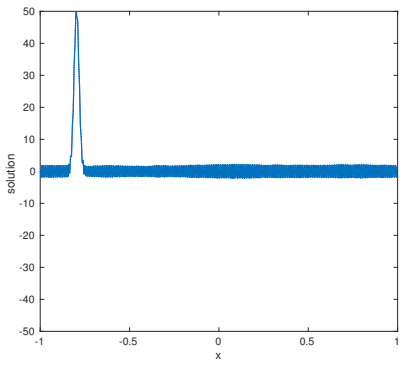


(d) $\lambda = 0.9$ with exact initial conditions on 200 grid points.

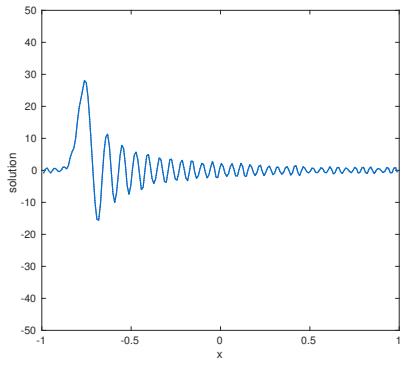
Figure 7.1: The leapfrog scheme for $u_t = u_x$ with exact initial conditions at time $t = 0.8$.



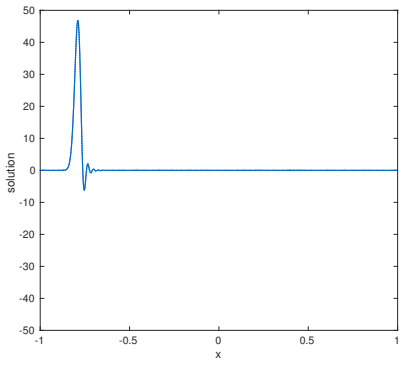
(a) $\lambda = 1$ on 200 grid points.



(b) $\lambda = 1$ on 800 grid points.

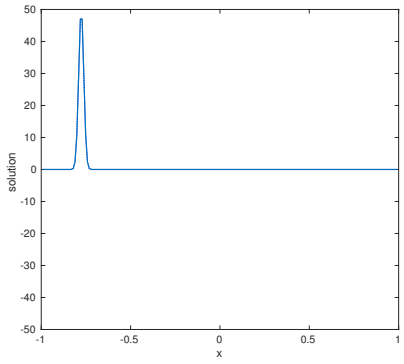


(c) $\lambda = 0.9$ on 200 grid points.

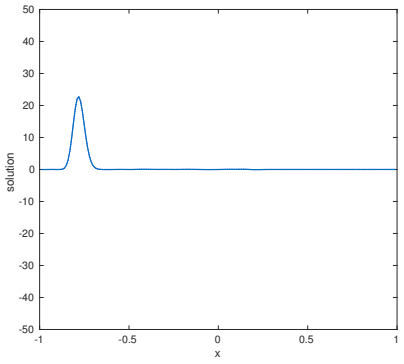


(d) $\lambda = 0.9$ on 800 grid points.

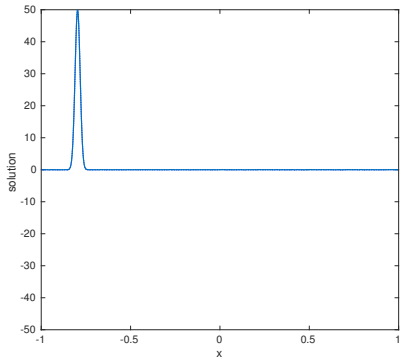
Figure 7.2: The leapfrog scheme for $u_t = u_x$ with consistent perturbed initial conditions at time $t = 2.8$. The oscillatory perturbation in the initial condition will vanish as $\Delta t \to 0$. However, the oscillations in the unstable scheme do not vanish as mesh refines.



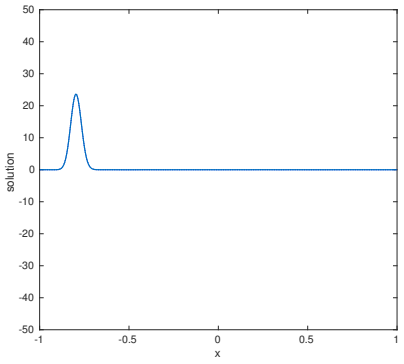
(a) $\lambda = 1$ on 200 grid points at time $t = 0.8$. Exact initial conditions.



(b) $\lambda = 0.9$ on 200 grid points at time $t = 0.8$. Exact initial conditions.



(c) $\lambda = 1$ on 800 grid points at time $t = 2.8$. Perturbed initial conditions.



(d) $\lambda = 0.9$ on 800 grid points at time $t = 2.8$. Perturbed initial conditions.

Figure 7.3: The upwind scheme for $u_t = u_x$.

Finally as a comparison, consider the first order accurate upwind scheme

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

Plugging in the ansatz $U_j^{n+1} = \hat{U}_k^{n+1} e^{\mathbb{i}\, kj\Delta x}$, *we get*

$$\hat{U}_k^{n+1} e^{\mathbb{i}\, kj\Delta x} = \hat{U}_k^n e^{\mathbb{i}\, kj\Delta x} + \lambda(\hat{U}_k^n e^{\mathbb{i}\, k(j+1)\Delta x} - \hat{U}_k^n e^{\mathbb{i}\, kj\Delta x}),$$

thus the amplification factor is $g(k) = 1 - \lambda + \lambda e^{\mathbb{i}\, k\Delta x}$. *We have*

$$|g^n| = |g|^n = \left[(1-\lambda)^2 + \lambda^2 + 2\lambda(1-\lambda)\cos\xi\right]^{\frac{n}{2}}.$$

For stability, we need $|g^n|$ *to be uniformly bounded as* $n \to \infty$, *which holds if and only if*

$$(1-\lambda)^2 + \lambda^2 + 2\lambda(1-\lambda)\cos\xi \le 1,$$

i.e.,

$$2(1-\cos\xi)\lambda(\lambda - 1) \le 0.$$

So the upwind scheme is stable if and only if $\lambda \le 1$. *See Figure 7.3 for the performance of the upwind scheme with exact and similarly perturbed initial conditions. We can observe that*

- *The upwind scheme with* $\lambda = 1$ *also produces the exact solution with exact initial conditions, and it is stable.*

- *If we compare Figure 7.3 (b) with Figure 7.1 (d), then it may seem that the upwind scheme gives a better solution in some sense (less oscillatory), which is not contradictory to the fact that the leapfrog scheme is a more accurate scheme. Recall that we define the order of accuracy for* $\Delta x \to 0$ *for smooth solutions. In this example, the solution is smooth, but obviously it is underresovled on the 200-point mesh. In other words, comparison of accuracy of numerical schemes makes little sense (if there is any) on this mesh because even the sampling error (representing the initial data on 200 grid points) is huge. Recall that sampling in space is equivalent to periodization in frequency. Also see Shannon Sampling Theorem in Chapter 4.*

Finally, let us try to understand the stability and "oscillations" in Figure 7.1 (d) from the perspective of stability region of ODE solvers. Recall that in Section 6.11.6 we defined the absolute stability for the linear multistep methods. In Example 6.11, we found the stability region of the leagfrog method is the inverval $(-\mathbb{i}, \mathbb{i})$ on the imagnary axis. In particular, consider solving Example 6.2 by the leapfrog method. Namely, we solve the

semidiscrete scheme $\mathbf{U}'(t) = A\mathbf{U}$, with

$$A = \frac{1}{2\Delta x}\begin{pmatrix} 0 & 1 & & & & -1 \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ 1 & & & & -1 & 0 \end{pmatrix},$$

by centered difference for the time derivative. Recall that A is circulant so DFT matrix diagonalizes it thus it is easy to find eigenvalues. The matrix A has purely imagnary eigenvalues because it is skew-symmetric. The eigenvalues are $\mathbb{i}\sin(k\Delta x)/\Delta x, k = 0, \cdots, N-1$. Since $\Delta x = \frac{2\pi}{N}$, the largest magnitude of the eigenvalues are $\mathbb{i}/\Delta x$ when $k\Delta x = \frac{\pi}{2}$. Thus to ensure the absolute stability, we need to take the time step to satisfy $\Delta t/\Delta x < 1$ (notice that $\lambda = 1$ will be on the outside of the stability region).

The "oscillations" in Figure 7.1 (d) do not "grow" in time. On the other hand, we need to see why we still have "oscillations" with absolute stability ensured. The absolute stability for a multistep method means the set of points z in complex plane so that the polynomial $\pi(\xi, z) = \rho(\xi) - z\sigma(\xi)$ satisfies the root condition. The root condition is derived from the initial value problem for the difference equation (for the leapfrog method solving $u' = au$),

$$U^{n+1} = U^{n-1} + 2\Delta t a U^n.$$

If $z = \Delta t a \in (-\mathbb{i}, \mathbb{i})$, then $\pi(\xi, z) = \rho(\xi) - z\sigma(\xi) = \xi^2 - 2z\xi - 1$ has two distinct roots ξ_1 and ξ_2 satisfying $|\xi_i| \leq 1$. The solution to this IVP can be written as

$$U^n = c_1\xi_1^n + c_2\xi_2^n.$$

Even though, $\|\xi_1^n\| \leq 1$ and $\|\xi_2^n\| \leq 1$, obviously we do not necessarily have $\|U^{n+1}\| \leq \|U^n\|$, which explains the "oscillations" in Figure 7.1 (d). However, the "energy" of U^n does not grow for fixed c_1 and c_2. In other words, the "oscillations" in Figure 7.1 (d) will not grow as time evolves.

7.5.2 The two way wave equation

The leapfrog method (second order centered difference for time and space derivatives) for the two-way wave equation $u_{tt} = u_{xx}$ is

$$\frac{U_j^{n+1} - 2U_j^n + U_j^{n-1}}{\Delta t^2} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Delta x^2}. \tag{7.12}$$

The simplified 1D Maxwell's equations can be written as

$$\begin{cases} E_t &= H_x \\ H_t &= E_x \end{cases}, \tag{7.13}$$

which is equivalent to $E_{tt} = E_{xx}$ or $H_{tt} = H_{xx}$.

The FDTD method (second order centered difference for time and space derivatives) for (7.13) is defined on staggered grid for H:

$$\begin{cases} \frac{E_j^{n+1} - E_j^n}{\Delta t} & = \frac{H_{j+\frac{1}{2}}^{n+\frac{1}{2}} - H_{j-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} \\ \frac{H_{j+\frac{1}{2}}^{n+\frac{1}{2}} - H_{j+\frac{1}{2}}^{n-\frac{1}{2}}}{\Delta t} & = \frac{E_{j+1}^n - E_j^n}{\Delta x} \end{cases} \tag{7.14}$$

It is a simple excersie to show that (7.14) is equivalent to (7.12) solving $E_{tt} = E_{xx}$ if we ignore the initial conditions.

Next, we consider the scheme (7.12) on the interval $x \in [0, 2\pi]$ with periodic boundary conditions. Let $\lambda = \frac{\Delta t}{\Delta x}$, then (7.12) can be written as

$$U_j^{n+1} = 2U_j^n + \lambda^2(E - 2 + E^{-1})U_j^n - U_j^{n-1},$$

where E is the shift operator. Define

$$V_j^n = \begin{pmatrix} U_j^n \\ U_j^{n-1} \end{pmatrix},$$

then we get

$$V_j^{n+1} = \begin{pmatrix} 2 + \lambda^2(E - 2 + E^{-1}) & -1 \\ 1 & 0 \end{pmatrix} V_j^n. \tag{7.15}$$

Plugging in the ansatz $V_j^n = \hat{V}_k^n e^{\mathbb{i}\, kj\Delta x}$, we get

$$\hat{V}_k^{n+1} = \begin{pmatrix} 2 + \lambda^2(e^{\mathbb{i}\, k\Delta x} - 2 + e^{-\mathbb{i}\, k\Delta x}) & -1 \\ 1 & 0 \end{pmatrix} \hat{V}_k^n.$$

Thus

$$\mathcal{G} = \begin{pmatrix} 2 + \lambda^2(2\cos\xi - 2) & -1 \\ 1 & 0 \end{pmatrix}.$$

The eigenvalues of $\mathcal{G}$ are $\mu_1 = a + \sqrt{a^2 - 1}, \mu_2 = a - \sqrt{a^2 - 1}$ with $a = 1 + \lambda^2(\cos\xi - 1)$. Notice that $-1 \le a \le 1$ if and only if $1 - \frac{2}{\lambda^2} \le \cos\xi \le 1$.

- If $\lambda > 1$, consider those ξ_0 such that $\cos\xi_0 < 1 - \frac{2}{\lambda^2}$. Then $a(\xi_0) < -1$ and $|\mu_2(\xi_0)| = |a - \sqrt{a^2 - 1}| > 1$. The von Neumann stability is violated thus not stable.

- If $\lambda \le 1$, then $a^2 - 1 \le 0$ thus $\mu_1 = a + \mathbb{i}\sqrt{1 - a^2}, \mu_2 = a - \mathbb{i}\sqrt{1 - a^2}$. So $|\mu_i| = 1$ and the von Neumann stability is satisfied. On the other hand, $\mathcal{G}$ is not a normal matrix and $\|\mathcal{G}\| > 1$. Nonetheless, $\mathcal{G}$ is diagonalizable

if $\mu_1 \neq \mu_2$, which is true if $\cos\xi \neq 1$. So $\mathcal{G} = T\begin{pmatrix}\mu_1 & 0\\ 0 & \mu_2\end{pmatrix}T^{-1}$ implies $\mathcal{G}^n = T\begin{pmatrix}\mu_1^n & 0\\ 0 & \mu_2^n\end{pmatrix}T^{-1}$, thus

$$\|\mathcal{G}^n\| \le \|T\|\left\|\begin{pmatrix}\mu_1^n & 0\\ 0 & \mu_2^n\end{pmatrix}\right\|\|T^{-1}\| = \|T\|\|T^{-1}\|\max_i|\mu_i^n| = \|T\|\|T^{-1}\|. \tag{7.16}$$

We still need to discuss $\|T\|$ and $\|T^{-1}\|$ and the case $\cos\xi = 1$ (or $\xi = 0$), see the discussion below for stability.

First we estimate $\|T\|$ and $\|T^{-1}\|$ for the case $\lambda \le 1$ and $\xi \neq 0$ (since $\xi = k\Delta x$, we consider $k = 1, 2, \cdots, N-1$). By using the fact $\mu_1\mu_2 = 1$, the eigenvectors can be chosen as

$$T = \begin{pmatrix}\mu_1 & \mu_2\\ 1 & 1\end{pmatrix}, \quad T^{-1} = \frac{1}{\mu_1 - \mu_2}\begin{pmatrix}1 & -\mu_2\\ -1 & \mu_1\end{pmatrix}.$$

Since $\mu_1^* = \mu_2$ and $\mu_2^* = \mu_1$, we have

$$T^*T = \begin{pmatrix}2 & \mu_1^*\mu_2 + 1\\ \mu_1\mu_2^* + 1 & 2\end{pmatrix} = \begin{pmatrix}2 & (\mu_1+\mu_2)\mu_2\\ (\mu_1+\mu_2)\mu_1 & 2\end{pmatrix}$$

whose eigenvalues are bounded at least by 4 (let x be an eigenvalue of TT^*, then $(x-2)^2 = (\mu_1+\mu_2)^2\mu_1\mu_2 = 4a^2$), yielding $\|T\| \le 2$. With $\mu_1\mu_2 = 1$, $\mu_1 + \mu_2 = 2a$ and $\mu_1 - \mu_2 = 2\,\mathrm{i}\,\sqrt{1-a^2}$, we have

$$(T^{-1})^*T^{-1} = \frac{1}{(\mu_1-\mu_2)(\mu_1^*-\mu_2^*)}\begin{pmatrix}2 & -(\mu_1+\mu_2)\\ -(\mu_1+\mu_2) & 2\mu_1\mu_2\end{pmatrix} = \frac{1}{2(1-a^2)}\begin{pmatrix}1 & -a\\ -a & 1\end{pmatrix}.$$

Let x_i be eigenvalues of $(T^{-1})^*T^{-1}$, then

$$x_1 = \frac{1}{2}\frac{1}{1-a}, \quad x_2 = \frac{1}{2}\frac{1}{1+a}.$$

Since $a = 1 + \lambda^2(\cos(k\Delta x) - 1)$, for fixed $\lambda \le 1$, by Taylor expansion on $\cos\Delta x$, we have

$$|x_i| \le \frac{1}{2\lambda^2}\frac{1}{1-\cos\Delta x} = \mathcal{O}(\Delta x^{-2}),$$

thus

$$\|T^{-1}\| \le C\Delta x^{-1}.$$

With (7.16), we have $\|\mathcal{G}^n\| \le C\Delta x^{-1}$, which means the scheme (7.12) is not stable according to the definition of stability.

Notice that we have used inequalities in (7.16), which might not be sharp. We can also compute $\mathcal{G}^n$ directly by

$$\mathcal{G}^n(k,\Delta t) = T\begin{pmatrix}\mu_1^n & 0\\ 0 & \mu_2^n\end{pmatrix}T^{-1} = \frac{1}{\mu_1-\mu_2}\begin{pmatrix}\mu_1^{n+1}-\mu_2^{n+1} & -\mu_1^n+\mu_2^n\\ \mu_1^n-\mu_2^n & -\mu_1^{n-1}+\mu_2^{n-1}\end{pmatrix}.$$

Since $|\mu_i|=1$, we can rewrite them as $\mu_1 = e^{\mathbb{i}\,\theta}$, $\mu_2 = e^{-\mathbb{i}\,\theta}$. So

$$\begin{aligned}\mathcal{G}^n(k,\Delta t) &= \frac{1}{e^{\mathbb{i}\,\theta}-e^{-\mathbb{i}\,\theta}}\begin{pmatrix}e^{\mathbb{i}(n+1)\theta}-e^{-\mathbb{i}(n+1)\theta} & -e^{\mathbb{i}\,n\theta}+e^{-\mathbb{i}\,n\theta}\\ e^{\mathbb{i}\,n\theta}-e^{-\mathbb{i}\,n\theta} & -e^{\mathbb{i}(n-1)\theta}+e^{-\mathbb{i}(n-1)\theta}\end{pmatrix}\\ &= \begin{pmatrix}\frac{\sin(n+1)\theta}{\sin\theta} & -\frac{\sin n\theta}{\sin\theta}\\ \frac{\sin n\theta}{\sin\theta} & -\frac{\sin(n-1)\theta}{\sin\theta}\end{pmatrix}\end{aligned}$$

As $k\longrightarrow 0$, $\theta\longrightarrow 0$ thus $\frac{\sin(n+1)\theta}{\sin\theta}\approx n+1$. So we have shown the following result

Lemma 7.2. *For the scheme* (7.12)*, for fixed* $\lambda\le 1$*, each entry of* $\mathcal{G}^n(k,\Delta t)$ *for* $k=1$ *is* $\mathcal{O}(n)$*.*

Next we look at what may happen when $\xi=0$ and $\lambda<1$ (similarly for the case $\lambda=1$ with $\xi=\pi$). Recall the discrete frequencies are $k=0,1,\cdots,N-1$ in the discrete Fourier transform that we used to derive the amplification matrix $\mathcal{G}(\Delta t,k)$. We have

$$\mathcal{G}(\Delta t,0) = \begin{pmatrix}2 & -1\\ 1 & 0\end{pmatrix}$$

with a repeated eigenvalue $\mu=1$. The Jordan form and the eigen-decomposition are

$$\mathcal{G}(\Delta t,0) = T\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}T^{-1},$$

thus

$$\mathcal{G}(\Delta t,0)^n = T\begin{pmatrix}1 & n\\ 0 & 1\end{pmatrix}T^{-1}.$$

Obviously we have $\|\mathcal{G}(\Delta t,0)^n\|\to\infty$ and $\|\mathcal{G}(\Delta t,N/2)^n\|\to\infty$ (assume N is even) as $n\to\infty$. So the scheme is unstalbe by the original definition. Now let us try to understand what it means that the stability is lost only when $k=0$ for $\lambda<1$ (and also $k=0,N/2$ for $\lambda=1$). The fact $\|\mathcal{G}(\Delta t,0)^n\|\to\infty$ implies that $\lim_{n\to\infty}|\hat{U}_0^n|=\infty$. In the discrete Fourier transform, the zero-th frequency corresponds to

$$\hat{U}_k^n = \sum_{j=0}^{N-1} e^{-\mathbb{i}\,kj\Delta x}U_j^n,\quad k=0,$$

thus

$$\hat{U}_0^n = \sum_{j=0}^{N-1} U_j^n.$$

Therefore this means that any perturbation in the total sum of the initial condition will not vanish as mesh refines. For instance, consider solving the IVP

$$u_{tt} = u_{xx}, u(x,0) = 0, u_t(x,0) = 0,$$

with periodic b.c. on an interval. Then the exact solution is constant zero. If we use the leapfrog scheme with the following initial conditions:

$$U^0 \equiv 0, U^1 \equiv \Delta t.$$

Plugging initial conditions into the scheme (7.12), we obtain $U^m = m\Delta t$. For any n satisfying $n\Delta t = t$, $U^n \equiv t$ thus we do not have convergence at all. On the other hand, if we use a second order accurate initial conditions:

$$U^0 \equiv 0, U^1 \equiv \Delta t^2,$$

then $U^n \equiv n\Delta t^2 = t\Delta t \to 0$ as $n \to \infty$.

Similar discussion holds for $k = N$ when $\lambda = 1$. The frequency $k = 0$ corresponds to the vector $[1 \quad 1 \quad \cdots \quad 1]$ while $k = N$ corresponds to the vector $v(N) = [1 \quad -1 \quad 1 \quad -1 \cdots -1]$ (if N is even). So any perturbation of the form $\Delta t v(N)$ in the initial condition will destroy convergence.

Therefore, at least for the case $\lambda < 1$, as long as we have an accurate initial condition so that the perturbation in the total sum is smaller than $\mathcal{O}(\Delta t)$ (a second order initial condition can be achieved by Taylor expansion $u(x, \Delta t) \approx u(x,0) + \Delta t u_t(x,0)$ since both $u(x,0)$ and $u_t(x,0)$ are given), it is still possible to have convergence.

7.5.3 Convergence for the two way wave equation

We can modify the proof of the Lax equivalence theorem to prove the convergence for the scheme (7.12). First, replace U_j^n by $u(x_j, t^n)$ in (7.12), the residue is the local truncation error

$$\tau^n = \mathcal{O}(\Delta t^2) + O(\Delta x^2).$$

Second, replace U_j^n by $u(x_j, t^n)$ in (7.15), the residue is

$$\Delta t^2 \tau^n = \Delta t^2 [\mathcal{O}(\Delta t^2) + O(\Delta x^2)].$$

Let $V^{n+1} = C(\Delta t)V^n$ denote the leapfrog scheme. Suppose

$$V^{n+1} = \begin{pmatrix} U_0^{n+1} \\ U_0^n \\ U_1^{n+1} \\ U_1^n \\ \vdots \\ U_{N-1}^{n+1} \\ U_{N-1}^n \end{pmatrix},$$

then define $Q_{\Delta x}$ as the sampling operator of any function at the spatial grid points and two time steps:

$$Q_{\Delta x}u(x,t) = \begin{pmatrix} u(x_0,t) \\ u(x_0,t-\Delta t) \\ u(x_1,t) \\ u(x_1,t-\Delta t) \\ \vdots \\ u(x_{N-1},t) \\ u(x_{N-1},t-\Delta t) \end{pmatrix}.$$

Define

$$\delta^n = [C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)]u(x,t),$$

where $S(\Delta t)u(x,t) = u(x,t+\Delta t)$ is the exact solution operator. Then

$$\delta^n = \Delta t^2 \tau^n.$$

The actual error that we want to control to prove the convergence is

$$\begin{aligned} \varepsilon^{n+1} &= V^{n+1} - Q_{\Delta x}u(x,(n+1)\Delta t) \\ &= C(\Delta t)[V^n - Q_{\Delta x}u(x,n\Delta t)] + [C(\Delta t)Q_{\Delta x} - Q_{\Delta x}S(\Delta t)]u(x,n\Delta t) \\ &= C(\Delta t)\varepsilon^n + \delta^n \end{aligned}$$

By solving $\varepsilon^{n+1} = C(\Delta t)\varepsilon^n + \delta^n$ (we no longer assume $\varepsilon^0 = 0$), we get

$$\varepsilon^n = C(\Delta t)^n \varepsilon^0 + \sum_{k=0}^{n-1} C(\Delta t)^{n-k-1}\delta^k.$$

Let F denote the $2N \times 2N$ matrix representing the linear transformation of taking the discrete Fourier transform for U^n and U^{n+1} respectively in V^{n+1}, i.e.,

$$FV^{n+1} = \begin{pmatrix} \hat{U}_0^{n+1} \\ \hat{U}_0^n \\ \hat{U}_1^{n+1} \\ \hat{U}_1^n \\ \vdots \\ \hat{U}_{N-1}^{n+1} \\ \hat{U}_{N-1}^n \end{pmatrix}.$$

Let $\hat{V}^n = FV^n$ and the amplification matrix be $\mathcal{G}(k)$ ($k = 0, \cdots, N-1$ is the discrete frequency). Then the scheme $V^{n+1} = C(\Delta t)V^n$ is equivalent to $\hat{V}^{n+1} = FC(\Delta t)F^{-1}\hat{V}^n$ and $FC(\Delta t)F^{-1}$ is a block diagonal matrix:

$$FC(\Delta t)F^{-1} = G = \begin{pmatrix} \mathcal{G}(0) & & & \\ & \mathcal{G}(1) & & \\ & & \ddots & \\ & & & \mathcal{G}(N-1) \end{pmatrix}.$$

In otherwords, the discrete Fourier transform that we have been using can block diagonalize the matrix $C(\Delta t)$.

Thus the error satisfies

$$\varepsilon^n = F^{-1}G^nF\varepsilon^0 + \sum_{k=0}^{n-1} F^{-1}G^{n-k-1}F\delta^k,$$

$$F\varepsilon^n = G^nF\varepsilon^0 + \sum_{k=0}^{n-1} G^{n-k-1}F\delta^k,$$

$$\hat{\varepsilon}^n = G^n\hat{\varepsilon}^0 + \sum_{k=0}^{n-1} G^{n-k-1}\hat{\delta}^k.$$

For $\lambda = \frac{\Delta t}{\Delta x} \leq 1$, in the previous subsection we have shown $\|\mathcal{G}^n(k)\| = \mathcal{O}(n) = \mathcal{O}(\Delta t^{-1})$ for any n and Δt satisfying $n\Delta t = t$ for fixed time t. Thus $\|G^n\| = \mathcal{O}(\Delta t^{-1})$

For the local truncation error part, since F is unitary, $\hat{\delta}^k = \Delta t^2\hat{\tau}^k = \Delta t^2[\mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2)]$. Thus

$$\left\|\sum_{k=0}^{n-1} G^{n-k-1}\hat{\delta}^k\right\| \leq \sum_{k=0}^{n-1} \|G^{n-k-1}\|\|\hat{\delta}^k\| = \sum_{k=0}^{n-1} \mathcal{O}(\Delta t)[\mathcal{O}(\Delta t^2)+\mathcal{O}(\Delta x^2)] = \mathcal{O}(\Delta t^2)+\mathcal{O}(\Delta x^2).$$

In other words, $\|G^n\| = \mathcal{O}(\Delta t^{-1})$ does not decrease the order of convergence by Δt^{-1} for the local truncation error part!

We only need to look at the numerical initial conditions. If the initial condition is second order accurate, i.e., $\varepsilon^0 = \mathcal{O}(\Delta t^2)$ thus $\hat{\varepsilon}^0 = \mathcal{O}(\Delta t^2)$. Then $G^n\hat{\varepsilon}^0 = \mathcal{O}(\Delta t)$, which is only first order. For instance, $\hat{\varepsilon}^0(0)$ denote the first two components in the vector $\hat{\varepsilon}^0$, then

$$\mathcal{G}(0)^n\hat{\varepsilon}^0(0) = T\begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix}T^{-1}\begin{pmatrix} \mathcal{O}(\Delta t^2) \\ \mathcal{O}(\Delta t^2) \end{pmatrix} = T\begin{pmatrix} \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta t^2)n \\ \mathcal{O}(\Delta t^2) \end{pmatrix} = T\begin{pmatrix} \mathcal{O}(\Delta t) \\ \mathcal{O}(\Delta t^2) \end{pmatrix}.$$

So we still have the convergence $\lim_{\Delta t\to 0}\|\varepsilon^n\| = 0$.

To summarize, **the scheme** (7.12) **is not stable** by the stability definition even though we can still have convergence with more assumptions on initial conditions. On the other hand, the scheme (7.14) is stable if $\frac{\Delta t}{\Delta x} < 1$ thus (7.14) is convergent with consistent initial conditions.

Remark 7.5. *Recall that the 1D Maxwell equation* (7.13) *is equivalent to the equation* $E_{tt} = E_{xx}$*. However, the initial value problems for these two equations (even with periodic boundary conditions) are not necessarily equivalent. Consider the following initial value problems on the interval* $x \in [0, 2\pi]$ *with periodic boundary conditions:*

1. $E_{tt} = E_{xx}$ *with* $E(x, 0)$ *and* $E_t(x, 0)$ *given.*

2. *The system* (7.13) *with* $E(x, 0)$ *and* $H(x, 0)$ *given.*

For these two IVPs to be equivalent, $H(x, 0) = \int E_t(x, 0)\, dx$ *must hold. In other words, for generic periodic initial data* $E(x, 0)$*,* $E_t(x, 0)$ *and* $H(x, 0)$*, these two IVPs are not equivalent.*

Problem 7.1. *Show the scheme* (7.14) *with periodic boundary conditions on* $x \in [0, 2\pi]$ *is stable for* $\frac{\Delta t}{\Delta x} < 1$*, by plugging in the ansatz* $E_j^n = \hat{E}_k^n e^{\mathbb{i}\, kj\Delta x}$ *and* $H_j^{n+\frac{1}{2}} = \hat{H}_k^{n+\frac{1}{2}} e^{\mathbb{i}\, k(j+\frac{1}{2})\Delta x}$*. The ansatz* $H_j^{n+\frac{1}{2}} = \hat{H}_k^{n+\frac{1}{2}} e^{\mathbb{i}\, k(j+\frac{1}{2})\Delta x}$ *is equivalent to using the transform* $H_j^{n+\frac{1}{2}} = \sum_{k=0}^{N-1} \hat{H}_k^{n+\frac{1}{2}} e^{\mathbb{i}\, k(j+\frac{1}{2})\Delta x}$ *(why?).*

Problem 7.2. *Recall that the scheme* (7.14) *is formally equivalent to* (7.12)*. However, these two schemes are obviously different since one is stable and the other one is not. To understand the difference or advantage on a staggered grid, consider the initial data* $E(x, 0) = E_t(x, 0) = H(x, 0) \equiv 0$*, with which the two IVPs are equivalent. The scheme* (7.12) *is not convergent with the initial condition* $E^0 \equiv 0, E^1 \equiv \Delta t$*. Derive an initial condition* E^0 *and* $H^{\frac{1}{2}}$ *so that the solution to* (7.14) *is the same the solution to* (7.12) *with* $E^0 \equiv 0, E^1 \equiv \Delta t$*. What does this initial condition imply? Is there any contradiction to the fact that* (7.14) *is convergent with consistent initial conditions?*

7.6 Dissipative schemes

In practical applications, the spectral radius of the amplification matrix is often easy to evaluate. Looking for a sufficient condition, this time in terms of the spectral radius leads us to the concept of dissipation of a scheme, to which we now turn our attention.

Definition 7.9. *A finite difference scheme* $V^{n+1} = C(\Delta t)V^n$ *is called dissipative of order* $2r$ *if the amplification matrix satisfies:*

$$\rho[\mathcal{G}(\Delta t, k)] \leq 1 - \delta|\xi|^{2r},$$

where $\xi = k\Delta x$ *for all* $\Delta t, k$ *and* $\delta > 0$ *is independent of* k *and* Δt*.*

This condition means that the eigenvalues of the amplification matrix are bounded away from one in a way proportional to the parameter ξ. As mentioned earlier, it is in general true that even stable schemes have eigenvalue 1 for the mode $\xi = 0$. We shall let return to this fact in examples to come. Dissipation allows this case to happen, but all other eigenvalues are strictly inside the unit circle.

When a scheme is dissipative, it is very likely to be stable, even in the variable coefficient case, a fact that makes dissipation an important property of the schemes. We present some examples to illustrate the concept of dissipation and its relation to stability and "growth" of the numerical solution.

Example 7.9. *Consider the Lax-Wendroff scheme for $u_t = u_x$ with periodic boundary conditions:*

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

This scheme is second order accurate, both in space and time. By the ansatz $U_j^n = e^{\mathbb{i}\, kj\Delta t}\hat{U}_k^n$, we get the corresponding amplification factor

$$g(\xi) = 1 + \mathbb{i}\,\lambda \sin\xi + \lambda^2(\cos\xi - 1).$$

For convenience, let $\eta = \sin(\xi/2)$ then $\sin\xi = 2\sin(\xi/2)\cos(\xi/2) = 2\eta\sqrt{1-\eta^2}$, thus

$$g(\xi) = 1 - 2\lambda^2\eta^2 + 2\,\mathbb{i}\,\lambda\eta\sqrt{1-\eta^2},$$

and

$$|g(\xi)|^2 = 1 - 4\lambda^2(1-\lambda^2)\eta^4.$$

If $\lambda \leq 1$, we get $|g(\xi)| \leq 1$ for all ξ thus we have stability. If $\lambda < 1$, the scheme is dissipative of order 4. To see why this is true, we have

$$|g(\xi)|^2 = 1 - 4\lambda^2(1-\lambda^2)\left(\frac{\sin^4(\xi/2)}{(\xi/2)^4}\right)\left(\frac{\xi}{2}\right)^4 = 1 - 4\lambda^2(1-\lambda^2)\gamma\left(\frac{\xi}{2}\right)^4.$$

We also have

$$\frac{\sin\theta}{\theta} = \frac{\sin|\theta|}{|\theta|} \geq \frac{2}{\pi},$$

thus

$$\gamma = \frac{\sin^4(\xi/2)}{(\xi/2)^4} \geq \left(\frac{2}{\pi}\right)^4.$$

So we get

$$|g(\xi)|^2 \leq 1 - 4\lambda^2(1-\lambda^2)\left(\frac{2}{\pi}\right)^4\left(\frac{\xi}{2}\right)^4 = 1 - \frac{4}{\pi^4}\lambda^2(1-\lambda^2)\xi^4 \leq 1.$$

Notice, however, that if $\lambda = 1$, then the scheme is not dissipative.

Dissipation of a scheme may be desirable in some problems, as is the case of highly fluctuating initial data (or "noisy information"), and it ensures stability. But in other cases, if the effect of dissipation is too strong, we might lose our solution by an exaggerated smoothing mechanism, which is very likely to occur if we want to perform a large number of time iterations. Therefore, whether we should choose a dissipative scheme or not strongly depends on the particular problem we want to solve.

Example 7.10. *Consider again the problem $u_x = u_x$ with periodicity conditions, and the scheme:*

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

This scheme is both accurate and stable when the CFL condition $\lambda \leq 1$. holds. The amplification factor is given by:

$$g(\xi) = \cos\xi + \mathbb{i}\sin\xi$$

and therefore:

$$|g(\xi)| = \cos^2\xi + \lambda^2\sin^2\xi = 1 - (1-\lambda^2)\sin^2\xi.$$

By a similar argument to the one given in the previous example, it can be shown that when $\lambda < 1$, the scheme is dissipative of order 2. At this point we should notice that, as seen directly from expression of $|g(\xi)|$, the values $\xi = -\pi, 0, \pi$ yield $|g(\xi)| = 1$. Although the definition of dissipation does not hold exactly in the way stated, the inequality fails only for a finite number of values of ξ. We in general consider these schemes as dissipative ones. Again we have that the scheme reproduces the exact solution at the grid points when $\lambda = 1$, so in that case there is no dissipation.

Example 7.11. *Consider now the leap frog scheme for approximating the solution of $u_x = u_x$ with periodic boundary conditions. The amplification matrix is*

$$\mathcal{G}(\xi) = \begin{pmatrix} 2\,\mathbb{i}\,\lambda\sin\xi & 1 \\ 1 & 0 \end{pmatrix}.$$

If $\lambda < 1$, then the scheme is stable as discussed before, and the eigenvalues satisfy

$$\mu_1(\xi) = \mathbb{i}\,\lambda\sin\xi + \sqrt{1-\lambda^2\sin^2\xi},$$

$$\mu_2(\xi) = \mathbb{i}\,\lambda\sin\xi - \sqrt{1-\lambda^2\sin^2\xi},$$

$$|\mu_i(\xi)| = 1.$$

Thus $\rho(\mathcal{G}) = 1$ for all values of k and Δt, which implies the leapfrog scheme is not dissipative.

The following example illustrates how a non-dissipative scheme may give rise to a very a bad approximation of a system for which energy is being dissipated.

Example 7.12. *Let $u(x,t)$ be the solution of:*

$$u_x = u_x - \beta u,$$

where $\beta > 0$, and assume periodicity conditions. The usual energy estimates for this system can be evaluated multiplying by it and integrating by parts, yielding:

$$\frac{d}{dt}\|u(x,t)\|^2 = \frac{d}{dt}\int_0^{2\pi} |u(x,t)|^2\,dx = \int_0^{2\pi} \frac{d}{dx}u^2(x,t)\,dx - 2\beta\int_0^{2\pi} u^2(x,t)\,dx$$

$$= -2\beta\|u(x,t)\|^2,$$

where we have used $u(0,t) = u(2\pi,t)$ for all $t > 0$. Integrating with respect to time we obtain:

$$\|u(x,t)\|^2 = 2^{-2\beta t}\|u(x,0)\|^2$$

so the solution decreases in time, that is, the system is dissipating energy.

Consider now the leap frog scheme for this problem, given by:

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

with periodicity conditions:

$$U_{-1}^n = U_{N-1}^n, U_0^n = U_N^n.$$

It is easy to check that this scheme is second order accurate. If $\lambda = \frac{\Delta t}{\Delta x} < 1$, then $U_j^{n+1} = U_j^{n-1} + \frac{\Delta t}{\Delta x}(U_{j+1}^n - U_{j-1}^n)$ is stable. Thus by the Kreiss Perturbation Theorem 7.2, the scheme in this example is also stable.

The amplification matrix is

$$\mathcal{G}(\xi) = \begin{pmatrix} 2\,\mathrm{i}\,\lambda\sin\xi - 2\beta\Delta t & 1 \\ 1 & 0 \end{pmatrix}.$$

In particular, for the mode corresponding to $\xi = \pi$

$$\mathcal{G}(\pi) = \begin{pmatrix} -2\beta\Delta t & 1 \\ 1 & 0 \end{pmatrix}.$$

whose eigenvalues are

$$\mu_1(\pi) = -\beta\Delta t - \sqrt{1 + \beta^2\Delta t^2},$$

$$\mu_2(\pi) = -\beta\Delta t + \sqrt{1 + \beta^2\Delta t^2}.$$

If Δt is small but positive, we have:

$$\mu_1(\pi) \approx -\beta\Delta t - (1 + \frac{1}{2}\beta^2\Delta t^2) \approx -1 - \beta\Delta t$$

and thus, upon calling $T = n\Delta t$,

$$\mu_1(\pi)^n \approx (-1)^n(1 + \beta\Delta t)^n = (1 + \beta T/n)^n(-1)^n.$$

We know the scheme is accurate and stable, so by Lax equivalence theorem, it converges. Nonetheless, the concept of convergence involves taking limits of the approximations as $\Delta t \to 0$ with $T = n\Delta t$ fixed. In practice we deal with a fixed positive $\Delta t > 0$ and compute n time steps. As the number of time steps increases, the eigenvalue grows exponentially as:

$$\mu_1(\pi)^n \approx (-1)^n e^{\beta T}.$$

As discussed before, the energy of the true solution decreases exponentially with time, for any initial condition, whereas the scheme might give rise to increasing numerical solution in practice. To verify this statement, it is enough to consider a particular case for the initialization of the scheme and show that the corresponding numerical solution U^n grows in time. Consider the initial condition:

$$U_j^0 = u(x_j, 0) = (-1)^j.$$

In order to implement the scheme, we need to specify also the first time step U^1, which we give as

$$U_j^1 = \mu_1(\pi)(-1)^j.$$

Then the numerical solution is

$$U_j^n = (-1)^j q^n$$

where q satisfies:

$$q^{n+1} = q^{n-1} - 2\beta\Delta q^n.$$

This equation is equivalent to the quadratic equation:

$$q^2 + 2\beta\Delta t q - 1 = 0,$$

whose roots are preciously $\mu_1(\pi)$ and $\mu_2(\pi)$. Thus the general solution for q is of the form:

$$q = \alpha_1\mu_1(\pi) + \alpha_2\mu_2(\pi).$$

Since U_j^1 must coincide with the initialization given above, it follows that $q = \mu_1(\pi)$ thus $U_j^n = (-1)^j\mu_1^n(\pi)$, which grows exponentially with the number of iterations performed, keeping $\Delta t > 0$ fixed. In other words, if using a fixed time step Δt, for computing longer and longer time T, the energy of the numerical solution grows exponentially in T.

7.6.1 0-stability V.S. absolute stability

For a finite difference scheme $V^{n+} = C(\Delta t)V^n$, the stability that we defined in this chapter is to require $\|C(\Delta t)^n\| \leq Ke^{\alpha t}$ for any n and Δt satisfying $n\Delta t = t$, which is also called **Lax-Richtmyer stability**, which is very similar to the 0-stability as defined in Chapter 6. On the other hand, we did not define the absolute stability for the scheme $V^{n+} = C(\Delta t)V^n$. Nonetheless, sometimes we achieved the absolute stability by requiring the Lax-Richtmyer stability. For instance, the amplification factor for the upwind scheme solving $u_t = u_x$ is $g(k) = 1 - \lambda + \lambda e^{\mathrm{i}\, k\Delta x}$, and

$$|g^n| = |g|^n = \left[(1-\lambda)^2 + \lambda^2 + 2\lambda(1-\lambda)\cos\xi\right]^{\frac{n}{2}}.$$

For the Lax-Richitmyer stability, we need $|g^n|$ to be uniformly bounded as $n \to \infty$, which holds if and only if

$$(1-\lambda)^2 + \lambda^2 + 2\lambda(1-\lambda)\cos\xi \leq 1,$$

i.e.,

$$2(1-\cos\xi)\lambda(\lambda - 1) \leq 0.$$

Therefore, the Lax-Richitmyer stability holds if and only if $|g| \leq 1$, which is very similar to the absolute stability defined in Chapter 6. In other words, we actually have the "absolute stability" for the schemes which perform well numerically, e.g., upwind and leagfrog schemes for $u_t = u_x$.

However, the "absolute stability" is less general than the Lax-Richtmyer stability, which is one of the reasons that we did not introduce or define the "absolute stability". For instance, the leapfrog scheme has Lax-Richtmyer stability and the "absolute stability" for the equation $u_t = u_x$. For the perturbed equation $u_t = u_x - \beta u$ with any $\beta > 0$ in Example 7.12, the leapfrog scheme also has Lax-Richtmyer stability due to the Kreiss Perturbation Theorem (Theorem 7.2), but the "absolute stability" is lost.

7.7 Difference schemes for hyperbolic systems in one dimension

It is often the case that the dimension of the space variable may change dramatically the properties of the numerical schemes, here we shall focus on problems in one dimension. Throughout this section, x will denote a scalar, and $u(x,t) = (u_1(x,t), \cdots, u_p(x,t))^T$ will denote the solution of a system of hyperbolic partial differential equations. We will be interested in approximating the solution $u(x,t)$ of the general, nonlinear equation of the form:

$$u_t(x,t) = \frac{\partial}{\partial x} F(u(x,t)), \tag{7.17}$$

where $F(u)$ is a function $F(u_1, \cdots, u_p) = (F_1(u_1, \cdots, u_p), \cdots, F_p(u_1, \cdots, u_p))^T$, e.g., the Euler equations discussed in Chapter 5. Therefore we have:

$$\frac{\partial}{\partial x} F(u(x,t)) = \frac{\partial}{\partial u} F(u) \frac{\partial}{\partial x} u(x,t),$$

where $\frac{\partial}{\partial u} F(u)$ denotes the gradient matrix $A(u)$ with components $a_{ij}(u) = \frac{\partial F_i(u)}{\partial u_j}$ so that the nonlinear system can be written in the form:

$$u_t = A(u) u_x. \tag{7.18}$$

We now generalize the definitions given previously in Chapter 5 for hyperbolic partial differential equations in the nonlinear case.

Definition 7.10. *The nonlinear equation* (7.18) *is called weakly, strongly, symmetric or strictly hyperbolic if for every u_0 fixed, the corresponding linearized system:*

$$u_t = A(u_0) u_x$$

is weakly, strongly, symmetric or strictly hyperbolic, respectively.

As already mentioned before, the Lax equivalence theorem states basically that an accurate scheme is stable if and only if it converges, provided that the problem is strongly well posed. Weak well posedness may give rise to instabilities. Therefore, we shall consider only problems which are strongly, symmetric or strictly hyperbolic, yielding strong well posedness. We study separately the schemes which are accurate of order $(1,1)$, or first order schemes, and schemes which are accurate of order $(2,2)$, or second order schemes.

7.7.1 First order schemes

We shall consider two schemes: Friedrich's scheme and the upwind schemes. We will assume that the problem (7.18) is strongly well posed. The accuracy of the schemes can be checked directly in the nonlinear form (7.18), but in order to establish stability, as done for well posedness, we look at the linearized scheme substituting $A(u)$ by a constant matrix of the form $A(u_0)$, for which the problem is strongly well posed, as our previous assumption implies. Consider Friedrich's scheme:

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

where $F_{j+1}^n = F(U_{j+1}^n)$. This scheme is based on a first order approximation of the derivatives using Taylor expansion, and it can be easily shown that this scheme is first order accurate, and details are left to the reader. Linearizing the function $F(u)$ around some arbitrary value to u_0 we replace $A(u)$ by

a constant matrix A, so that the linearized problem is equivalent to the original problem with $F(u) = Au$. Substituting in the Friedrich's scheme, we get the linearized form of the scheme as:

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

The corresponding amplification matrix is given by:

$$\mathcal{G}(\xi) = I\cos\xi + \mathbb{i}\,\lambda A \sin\xi,$$

where $\xi = k\Delta x$, and I is the $p \times p$ identity matrix. If the original problem is strongly or strictly hyperbolic, then it follows that the matrix $A = A(u_0)$ is diagonalizable, i.e. there exists a matrix T such that

$$T^{-1}AT = \begin{pmatrix} a_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a_p \end{pmatrix}$$

where $a_1, \cdots, a_p$ are the real eigenvalues of A. Therefore:

$$T^{-1}\mathcal{G}(\xi)T = I\cos\xi + \mathbb{i}\,\lambda \begin{pmatrix} a_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a_p \end{pmatrix} \sin\xi$$

which is also diagonal with entries (eigenvalues) given explicitly by:

$$\mu_k(\xi) = cos\xi + \mathbb{i}\,\lambda a_k \sin\xi,$$

which implies that:

$$|\mu_k(\xi)|^2 = \cos^2\xi + \lambda^2 a_k^2 \sin^2\xi = 1 - (1 - \lambda^2 a_k^2)\sin^2\xi.$$

Therefore, if $\rho(A) = \max_k |a_k|$ satisfies the inequality

$$\frac{\Delta t}{\Delta x}\rho(A) \leq 1,$$

then von Neumann stability condition will hold and $|\mu_k(\xi)| \leq 1$ for all k and ξ. Furthermore, if $\frac{\Delta t}{\Delta x}\rho(A) < 1$, then $|\mu_k(\xi)|$ will be bounded away from 1 for all $0 \leq \xi < 2\pi$ except for $\xi = 0, \pi$. It is left as an exercise to prove that under strict inequality of von Neumann condition, the scheme is dissipative of order 2. Since $\mathcal{G}$ is diagonalized by a constant matrix T, the scheme for the linearized system is stable when $\frac{\Delta t}{\Delta x}\rho(A) \leq 1$. In practice, if the scheme for the linearized system is stable under the CFL condition $\frac{\Delta t}{\Delta x}\rho(A) \leq 1$, then the scheme for the nonlinear system is usually "stable" under the CFL condition $\frac{\Delta t}{\Delta x}\max_u \rho(A(u)) \leq 1$ for solving smooth solutions.

We now turn to the study of upwind schemes. These schemes are motivated by the scalar equation:

$$u_t = au_x$$

, when p = 1. If $a > 0$ the characteristics are straight lines moving to the left, and the scheme constructed in order to "follow" the physical characteristics is:

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{\Delta x} a(U_{j+1}^n - U_j^n), \quad a > 0, \tag{7.19}$$

and, as discussed before, the scheme is accurate and stable for $0 < a\lambda \leq 1$, for $\lambda = \Delta t/\Delta x$. On the other hand, if $a < 0$, then the characteristics point to the right and it is more reasonable to use the information carried by U_j^n and U_{j-1}^n, in order to evaluate U_j^{n+1} through the scheme:

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{\Delta x} a(U_j^n - U_{j-1}^n), \quad a < 0, \tag{7.20}$$

In this case, stability follows from the condition $-1 \leq \lambda a < 0$.

In order to extend the concept of upwind schemes to systems of hyperbolic equations some care must be taken. We shall gradually construct the general recursion formula.

Example 7.13. *Consider first the following example:*

$$\begin{pmatrix} u \\ v \end{pmatrix}_t = \begin{pmatrix} 0 & c \\ c & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x$$

where $c > 0$. This system is equivalent to:

$$(u+v)_t = c(u+v)_x$$
$$(u-v)_t = -c(u-v)_x$$

Therefore an upwind schemecan be constructed naturally for $U + V$ and $U - V$, which yields:

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

Adding and subtracting these two equations, we get the following equivalent scheme:

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x} c(V_{j+1}^n - V_{j-1}^n) + \frac{\Delta t}{2\Delta x} c(U_{j+1}^n - 2U_j^n + U_{j-1}^n)$$
$$V_j^{n+1} = V_j^n + \frac{\Delta t}{2\Delta x} c(U_{j+1}^n - U_{j-1}^n) + \frac{\Delta t}{2\Delta x} c(V_{j+1}^n - 2V_j^n + V_{j-1}^n). \tag{7.21}$$

Looking at these equations it is clear that even though we started with upwind schemes, (7.21) *are centered expressions: there is no longer any explicit direction for the characteristics. Also, it should be noticed that besides an approximation of a first order derivative, the above equations also contain an approximation to a second order derivative, which we did not have when we started the construction of the schemes. When generalizing the concept of an upwind scheme, we must allow for centered expressions that at first sight may not seem to carry information along characteristics, keeping in mind this simple example. The scheme* (7.21) *is a first order accurate (both in time and space) scheme for the first order convection equation, and a second order accurate (in space) scheme for the convection diffusion equation:*

$$u_t = cv_x + c\Delta x u_{xx}$$
$$v_t = cu_x + c\Delta x v_{xx},$$

which is called the modified equation for the scheme (7.21)*. Thus at least intuitively we expect the scheme* (7.21) *produces a smoother numerical solution than the exact solution to the original first order convection equation.*

We now write an equivalent expression for (7.19) *and* (7.20) *by adding and subtracting the appropriate terms*

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x}a(U_{j+1}^n - U_{j-1}^n) + \frac{\Delta t}{2\Delta x}a(U_{j+1}^n - 2U_j^n + U_{j-1}^n), \quad a > 0 \tag{7.22}$$

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x}a(U_{j+1}^n - U_{j-1}^n) - \frac{\Delta t}{2\Delta x}a(U_{j+1}^n - 2U_j^n + U_{j-1}^n), \quad a < 0 \tag{7.23}$$

from where it is clear now that the general form of upwind Scheme for the scalar case $p = 1$ *is given by:*

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x}a(U_{j+1}^n - U_{j-1}^n) + \frac{\Delta t}{2\Delta x}|a|(U_{j+1}^n - 2U_j^n + U_{j-1}^n) \tag{7.24}$$

While schemes (7.22) *and* (7.23) *are hard to generalize to the variable coefficient case in the form they are usually written, it is straightforward to implement* (7.24) *in the case* $a = a(x)$ *using the values of* $a(x)$ *and* $|a(x)|$*.*

As can be verified from (7.25)*, there is indeed a term that approximates a second derivative within the upwind schemes. Furthermore, this term has a positive coefficient,* $|a| > 0$*, which in turn introduces a dissipative mechanism for the scheme.*

In order to generalize (7.24) for systems of hyperbolic equations we first have to define the "absolute" value of a matrix that will play the role of $|a|$ in the scalar case. Consider again the linearized, strongly hyperbolic system, so that the matrix A is a constant, diagonalizable matrix:

$$u_t = Au_x.$$

Definition 7.11. *Let A be diagonalizable by T, so that:*

$$\Lambda = T^{-1}AT = \begin{pmatrix} a_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a_p \end{pmatrix}.$$

The absolute value of Λ is defined by:

$$|\Lambda| = \begin{pmatrix} |a_1| & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & |a_p| \end{pmatrix}$$

and the absolute value of the matrix A is defined to be $|A| = T|\Lambda|T^{-1}$, so that $|A|$ is also a $p \times p$ matrix which T itself diagonalizes.

Example 7.14. *For the matrix $A = \begin{pmatrix} 0 & c \\ c & 0 \end{pmatrix}$, we have:*

$$A = T \begin{pmatrix} c & 0 \\ 0 & -c \end{pmatrix} T^{-1}, \quad T = T^{-1} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Thus

$$|A| = T \begin{pmatrix} c & 0 \\ 0 & c \end{pmatrix} T^{-1} = cI.$$

The generalization of scheme (7.24) *to the system $u_t = Au_x$ is given by the scheme:*

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x} A(U_{j+1}^n - U_{j-1}^n) + \frac{\Delta t}{2\Delta x}|A|(U_{j+1}^n - 2U_j^n + U_{j-1}^n) \quad (7.25)$$

It is straightforward to verify that (7.21) *satisfy* (7.25).

Definition 7.12. *Let $f(x)$ be a real valued function of the variable x. We define the positive part f^+ of f as the function $f^+(x) = \max\{0, f(x)\}$ or equivalently:*

$$f^+(x) = \begin{cases} f(x), & \text{if} \quad f(x) > 0 \\ 0, & \text{if} \quad f(x) \le 0 \end{cases}$$

and analogously, the negative part $f^-(x)$ of f is defined by $f^-(x) = -\max\{0, -f(x)\}$. Using these definitions, it follows that:

$$f = f^+ + f^-, \quad |f| = f^+ - f^-.$$

Substituting in (7.24) the values of a and $|a|$ in terms of the positive and negative parts, we obtain an alternative expression for the upwind scheme in the scalar case, namely,

$$U_j^{n+1} = U_j^n + (a^+)\frac{\Delta t}{\Delta x}(U_{j+1}^n - U_j^n) + (a^-)\frac{\Delta t}{\Delta x}(U_j^n - U_{j-1}^n). \quad (7.26)$$

This representation of the upwind scheme has the advantage that it shows explicitly the directions of the characteristics that the scheme "picks' according to the sign of a, which becomes more useful when a is a variable coefficient $a(x)$. Following the natural extension, we can now define the positive and negative part of a diagonalizable matrix in terms of the absolute value.

Definition 7.13. *Let A be a diagonalizable matrix. We define the positive (negative) part of A by:*

$$A^+ = \frac{A + |A|}{2}, \quad A^- = \frac{A - |A|}{2}.$$

Scheme (7.25) can now be written in a more compact form using the positive and negative parts of the matrix A, yielding:

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

This expression gives the general form of the upwind scheme for approximating the solution of symmetric, strongly or strictly hyperbolic systems with constant coefficients. To generalize the scheme to the nonlinear case, where $A = A(u)$, we need some material on nonlinear equations in a more general scope. The foundamental method of this type is called Godunov's scheme, which we will not introduce in this chapter. We summarize now the concepts related to the upwind scheme in the linear case:

$$u_t = A u_x$$

where A is diagonalizable, so that the problem is strongly well posed. Accuracy of order (1,1) and stability of the upwind scheme follow straightforward assuming:

$$\rho(A) \frac{\Delta t}{\Delta x} \leq 1.$$

Indeed, one can decouple the system using the transformation $w = Tu$, which yields

$$w_t = \Lambda w_x.$$

The corresponding scheme defined by $W_j^n = TU_j^n$ has p components that satisfy schemes (7.24) or, equivalently, (7.26) with a, $|a|$, and a^+, a^- replaced in terms of the eigenvalues a_k of A. For each component the scheme for W_j^n is accurate of first order and stable, as an approximation of the solution of $w_t = \Lambda w_x$. Applying the bounded, linear transformation T to W_j^n, the result for the original problem is established.

7.7.2 Second order schemes

Roughly speaking, we can divide second order schemes into the dissipative and the non-dissipative ones. As before, we will assume strong well posedness of the problem $u_t = A(u)u_x$. Accuracy of the schemes can be evaluated directly for the schemes in general form, but in order to establish stability, we shall consider the linearized versions, as we did in the previous section. A representative of the class of dissipative schemes of second order accuracy is the Lax-Wendroff scheme, which we shall study first.

Definition 7.14. *A scheme for approximating the solution of $u_t = A(u)u_x$ is called a Lax-Wendroff scheme if under the assumption $A(u) = A$ (or $F(u) = Au$ is linear), the scheme reduces to:*

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x} A(U_{j+1}^n - U_{j-1}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x} A\right)^2 (U_{j+1}^n - 2U_j^n + U_{j-1}^n). \quad (7.27)$$

It may be shown that the scheme (7.27) is actually **the only second order scheme** for the linear problem that uses U_{j-1}^n, U_j^n, and U_{j+1}^n, to evaluate U_j^{n+1}.

Lax-Wendroff schemes arise from the idea of replacing time derivatives by space derivatives, using the equation $u_t = F(u)$, and approximating the latter by finite differences. Using a Taylor expansion for u, we have:

$$u(x, t + \Delta t) = u(x, t) + \Delta t u_t(x, t) + \frac{\Delta^2}{2} u_{tt}(x, t) + \mathcal{O}(\Delta t^3).$$

Since $u_t(x, t) = F(u(x, t))$, in the linear case where $F(u) = Au$ we get:

$$u_t(x, t) = Au_x(x, t), \quad u_{tt}(x, t) = A^2 u_{xx}(x, t).$$

Using now finite difference approximations for u_x and $u_x x$, it follows that the linear form of the scheme (7.27) is accurate of order (2,2). The amplification matrix of the linear form of the Lax-Wendroff scheme is:

$$\mathcal{G}(\xi) = I + \mathbb{i}\, \lambda A \sin\xi + \lambda^2 A^2 (\cos\xi - 1),$$

where, as usual, $\xi = k\Delta t$ and $\lambda = \Delta t/\Delta x$. Calling $\eta = sin(\xi/2)$ we can write:

$$\mathcal{G}(\xi) = I + 2\,\mathbb{i}\, \lambda A \eta \sqrt{1 - \eta^2} - 2\lambda^2 A^2 \eta^2.$$

Therefore any eigenvalue $\mu(\eta)$ of the amplification matrix will be of the form:

$$\mu(\eta) = 1 + 2\,\mathbb{i}\, \lambda \mu(A) \eta \sqrt{1 - \eta^2} - 2\lambda^2 \mu(A)^2 \eta^2,$$

which follows from the fact that A is diagonalizable. From the expression of the eigenvalues $\mu(\eta)$ of the amplification matrix we have:

$$|\mu(\eta)|^2 = 1 - \lambda^2 \mu(A)^2 \eta^4 (1 - \lambda^2 \mu(A)^2)$$

which holds for every eigenvalue of $\mathcal{G}(\xi)$. Recall that the spectral radius of $\mathcal{G}(\xi)$ is defined as the maximum value of $\mu(\eta)$. Therefore, upon letting μ_* be the eigenvalue of A which maximizes the above expression for $|\mu(\eta)|$ we get:

$$|\rho(\mathcal{G})|^2 = 1 - \lambda^2\mu_*^2\eta^4(1 - \lambda^2\mu_*^2)$$

Clearly, von Neumann condition will be satisfied if

$$\lambda\rho(A) \leq 1$$

which implies $\lambda\mu(A) \leq 1$ for all eigenvalues of A. Furthermore, if $\lambda\mu_* < 1$, then the scheme given by (7.27) is dissipative of order 4. Here the dissipation can be controlled through the parameter λ, or, equivalently, through the choice of Δt. In the nonlinear case we can construct different schemes which fall within the class of LaxWendroff schemes, depending on the way we approximate the derivatives.

For the nonlinear case, we have

$$u_{tt} = [F(u)]_{xt} = [F(u)_t]_x = [A(u)u_t]_x = [A(u)F(u)_x]_x.$$

Substituting $u_t = F(u)_x$ and the above expression in the Taylor expansion, we get:

$$u(x, t + \Delta t) = u(x, t) + \Delta t F(u)_x + \frac{\Delta t^2}{2}[A(u)F(u)_x]_x + \mathcal{O}(\Delta t^3).$$

The scheme originally proposed by Lax and Wendroff is based on approximating the space derivatives in the expansion above up to order $O(\Delta x^2)$ and is given by:

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{2\Delta x}(F_{j+1}^n - F_{j-1}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2 (A_{j+\frac{1}{2}}^n(F_{j+1}^n - F_j^n) - A_{j-\frac{1}{2}}^n(F_j^n - F_{j-1}^n)). \tag{7.28}$$

where:

$$F_j^n = F(U_j^n), \quad A_{j+\frac{1}{2}}^n = A(\frac{U_{j+1}^n + U_j^n}{2}).$$

Scheme (7.28) becomes rather inefficient in practical applications due to the many computations involved at each time step iteration in order to evaluate A, and F. A modification of this scheme which is very popular considers approximating derivatives at "half stages" of the iteration, using:

$$u(x, t + \Delta t) = u(x, t) + \Delta t u_t(x, t + \frac{1}{2}\Delta t) + \mathcal{O}(\Delta t^2),$$

and it is known as the *MacCormack* scheme. Each iteration has two steps corresponding to first order approximations of the solution at half steps.

The Scheme is given by:

$$U_j^* = U_j^n + \frac{\Delta t}{\Delta x}(F_{j+1}^n - F_j^n)$$
$$U_j^{n+1} = \frac{1}{2}\left(U_j^n + U_j^* + \frac{\Delta t}{\Delta x}(F_j^* - F_{j-1}^*)\right)$$

where

$$F_j^n = F(U_j^N), \quad F_j^* = F(U_j^*).$$

This scheme is a two-stage scheme which evaluates a "predictor" U_j^* and a "corrector" $U_j^{**} = U_j^* + \frac{\Delta t}{\Delta x}(F_j^* - F_{j-1}^*)$, and then forms U_j^{n+1} as the average $(U_j^* + U_j^{**})/2$.

It is clear that in order to evaluate U_j^{n+1} the scheme uses the same points in the grid at time in as the Lax-Wendroff scheme. Notice, however, that here we go from right to left at the middle stage $*$, and then from left to right. The "efficiency" of a scheme is often related to the cost in computer time of each iteration. In these terms, one can compare different schemes. For the Lax-Wendroff scheme, we need to evaluate F_{j-1}^n, F_j^n, F_{j+1}^n, $A_{j+\frac{1}{2}}^n$ and $A_{j-\frac{1}{2}}^n$ and perform matrix multiplications in each iteration, whereas MacCormack Scheme requires only the evaluation of F_j^n, F_{j+1}^n, F_j^* and F_{j-1}^*.

It only remains to prove the order of accuracy of MacCormack scheme. The fact that it belongs to the class of Lax-Wendroff schemes follows straightforward replacing $F(u)$ by Au with A a constant matrix.

The local truncation error of the MacCormack scheme is $\mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta t \Delta x)$, in which we assume $\Delta t = \mathcal{O}(\Delta x)$. Thus it is a second order accurate in space and time.

Among the class of second order non-dissipative schemes is the leap frog scheme. For the general non-linear equation, the scheme is given by:

$$U_j^{n+1} = U_j^{n-1} + \frac{\Delta t}{\Delta x}(F_{j+1}^n - F_{j-1}^n). \tag{7.29}$$

We analyzed this scheme in detail for the linear case, and found out that it is not dissipative and it is stable, provided that $\frac{\Delta t}{\Delta x}\rho(A) < 1$. The fact that (7.29) is accurate of second order follows a straightforward calculation. This scheme is generally more efficient than Lax-Wendroff schemes, although it needs roughly twice as much memory due to the dependence on two previous time stages to evaluate U^{n+1}, therefore in practice, we usually face the trade-off between efficiency and storage requirements. Since this is a non-dissipative scheme, it will not give good approximations for nonlinear equations. We now proceed to describe a way to introduce a dissipative term in (7.29) to deal with this problem. When adding a dissipative term in the form of a small perturbation, care must be taken so that the resulting linear scheme retains stability. Recall that in the linear case $F(u) = Au$,

the amplification matrix $\mathcal{G}(\xi)$ is a $2p \times 2p$ matrix (A itself is a $p \times p$ matrix) given by:

$$\mathcal{G}(\xi) = \begin{pmatrix} 2\,\mathbb{i}\,\lambda A \sin\xi & I \\ I & 0 \end{pmatrix}.$$

where now each of the entries is itself a $p \times p$ matrix. In order to express the eigenvalues $\mu(\xi)$ of $\mathcal{G}$ in terms of those of A, we use the fact that if A is diagonalizable by a matrix T, then $\hat{\mathcal{G}}$ possesses the same eigenvalues of $\mathcal{G}$, for:

$$\hat{\mathcal{G}}(\xi) = \begin{pmatrix} T^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} 2\,\mathbb{i}\,\lambda A \sin\xi & I \\ I & 0 \end{pmatrix} \begin{pmatrix} T & 0 \\ 0 & I \end{pmatrix} = \begin{pmatrix} 2\,\mathbb{i}\,\lambda T^{-1}AT \sin\xi & I \\ I & 0 \end{pmatrix}.$$

Recall that $T^{-1}AT$ is a diagonal matrix with diagonal entries $a_1, \cdots, a_p$. From this expression (by rearranging rows/columns, $\hat{\mathcal{G}}(\xi)$ is similar to a block diagonal matrix with 2×2 diagonal blocks $\begin{pmatrix} 2\,\mathbb{i}\,\lambda a_j \sin\xi & 1 \\ 1 & 0 \end{pmatrix}$), it follows that any eigenvalue $\mu(\xi)$ of the amplification matrix satisfies:

$$\mu^2(\xi) = 1 + 2\,\mathbb{i}\,\lambda a_j \sin\xi \mu(\xi),$$

for some $j = 1, \cdots, p$.

We will show that, if we add a dissipative term to the leap frog scheme at time level n, this will give rise to instabilities. By a dissipative term we mean an approximation to a second derivative, as would be a term of the form:

$$\varepsilon(U^n_{j+1} - 2U^n_j + U^n_{j-1}), \tag{7.30}$$

added to the scheme (7.29), where ε is a "small" perturbation. Notice that any modification at time level n will affect the first block in the amplification matrix. If the term added is (7.30), then the modified amplification matrix will be of the form:

$$\mathcal{G}(\xi) = \begin{pmatrix} 2\,\mathbb{i}\,\lambda A \sin\xi + \varepsilon \sin^2(\xi/2) I & I \\ I & 0 \end{pmatrix}$$

and therefore the eigenvalues will now satisfy:

$$\mu^2(\xi) = 1 + (2\,\mathbb{i}\,\lambda a_j \sin\xi + \varepsilon \sin^2(\xi/2))\mu(\xi).$$

In general, if E denotes the shift operator $EU^n_j = U^n_{j+1}$, adding a dissipative term at time level n amounts to modifying (7.29) yielding the scheme:

$$U^{n+1}_j = U^{n-1}_j + \frac{\Delta t}{\Delta x} A(U^n_{j+1} - U^n_{j-1}) + \varepsilon P(E) U^n_j, \tag{7.31}$$

where $P(E)$ is a function of the shift operator (in particular, the term in (7.30) corresponds to $P(E) = E - 2I + E^{-1}$). Since $P(E)$ approximates a

second order derivative, its Fourier transform $\hat{P}(\xi)$ will be a real function of ξ. It is this function $\hat{P}(\xi)$ which will appear now added in the first block of the amplification matrix and thus the modified eigenvalues will in general satisfy:

$$\mu^2(\xi) = 1 + (2\,\mathbb{i}\,\lambda a_j \sin\xi + \varepsilon\hat{P}(\xi))\mu(\xi).$$

for some eigenvalue at of A. The fact that (7.31) is an unstable scheme follows now from the following lemma, applied to the eigenvalues $\mu(\xi)$.

Lemma 7.3. *Let x_1 and x_2 be the solutions of the equation $x^2 - \alpha x - 1 = 0$. If both $|x_1| \leq 1$ and $|x_2| \leq 1$, then necessarily the coefficient α is purely imaginary.*

Proof. Let $x_1 = re^{\mathbb{i}\,\theta}$, then $x_1 x_2 = -1$ implies $x_2 = \frac{1}{r}e^{-\mathbb{i}\,\theta}$. Both $|x_1| \leq 1$ and $x_2 \leq 1$ imply $r = 1$. Thus $\alpha = x_1 + x_2 = 2\,\mathbb{i}\sin\theta$. □

Remark 7.6. *Using exactly the same analysis, we may conclude in general that the leap frog scheme gives rise to instabilities when it is used to approximate parabolic equations. For the heat equation, this can also be explained by the stability region of the leapfrog method, which is only on the imaginary axis, while the centered finite difference used in approximating the second order derivatives will give real eigenvalues, as discussed in Example 6.1.*

In order to introduce the correct amount of dissipation, we must add the dissipation term at time level $n-1$. We shall use the following operator $E^{\frac{1}{2}}$ which is defined as $E^{\frac{1}{2}}U_j^n = U_{j+\frac{1}{2}}^n$. Using this notation, the leap frog scheme (7.29) can be rewritten in the form:

$$U_j^{n+1} = U_j^{n-1} + \frac{\Delta t}{\Delta x}(E^{\frac{1}{2}} - E^{-\frac{1}{2}})(E^{\frac{1}{2}} + E^{-\frac{1}{2}})F_j^n$$

We shall show now that the modification of the scheme that is dissipative is given in general form by the expression:

$$U_j^{n+1} = U_j^{n-1} + \frac{\Delta t}{\Delta x}(E^{\frac{1}{2}} - E^{-\frac{1}{2}})(E^{\frac{1}{2}} + E^{-\frac{1}{2}})F_j^n - \frac{\varepsilon}{16}(E^{\frac{1}{2}} - E^{-\frac{1}{2}})^4 U_j^{n-1}. \tag{7.32}$$

Let $\eta = \sin(\xi/2)$, the amplification matrix of the linearized scheme (7.32) is:

$$\mathcal{G}(\xi) = \begin{pmatrix} 2\,\mathbb{i}\,\lambda A\sin\xi & (1-\varepsilon\eta^4)I \\ I & 0 \end{pmatrix}$$

and the eigenvalues hold the relations:

$$\mu^2(\xi) = 1 - \eta^4 + 2\,\mathbb{i}\,\lambda\mu(A)\sin\xi\sin\xi\mu(\xi),$$

for some eigenvalue $\mu(A)$ of A. Therefore:

$$\mu(\xi) = \mathbb{i}\,\lambda\mu(A)\sin\xi \pm \sqrt{1 - |\mu(A)|^2\sin^2\xi - \varepsilon\eta^4}$$

and we have $|\mu(\xi)|^2 = 1 - \varepsilon\eta^4$, provided that

$$1 - |\lambda\mu(A)|^2 \sin^2 \xi - \varepsilon\eta^4 > 0, \tag{7.33}$$

for all eigenvalues of A and all ξ. Under this condition, the modified scheme (7.32) is stable and dissipative. Remark, though, that in order for (7.33) to hold, whenever we add dissipation ($\varepsilon > 0$), we must also decrease the value of $\lambda = \Delta t/\Delta x$. This means that for a fixed space grid, a larger number of time steps must be evaluated to get the approximation of the solution at some given time t.

8

Iterative methods for solving linear systems

In this chapter we briefly discuss a few iterative methods for solving the large sparse linear systems arising from discretizing time-dependent PDEs. We start with two such examples:

- *Implicit time discretization for the diffusion terms.* Consider solving the heat equation $u_t = u_{xx}$ with periodic b.c. on $x \in [0, \pi]$. Let us use the centered difference for the spatial derivative. If we use the explicit forward Euler for the time derivative,

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

  then the amplification factor is $g(\xi) = 1 + 2\frac{\Delta t}{\Delta x^2}(\cos\xi - 1)$ where $\xi = \Delta x$. The Lax-Richtmyer stability $|g(\xi)^n| \leq Ke^{\alpha t}$ holds if and only if $|g(\xi)| \leq 1$, which implies $\frac{\Delta t}{\Delta x^2} \leq \frac{1}{2}$. In practice, the time step $\Delta t = \frac{1}{2}\Delta x^2$ is unbearably small. A larger time step like $\Delta t = \mathcal{O}(\Delta x)$ can be achieved if using the backward Euler time discretization,

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

  For implementing this implicit scheme, a linear system $AU^{n+1} = U^n$ must be solved in each time step. The matrix A approximates the operator $I - \Delta t\frac{\partial^2}{\partial x^2}$.

- *Incompressible flows.* If we take the curl of *2D incompressible Navier-Stokes*

$$\mathbf{u}_t + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \nu\Delta\mathbf{u}, \quad \nabla\cdot\mathbf{u} = 0,$$

  we get the vorticity stream-function formulation of the the 2D incompressible Navier-Stokes equations:

$$\omega_t + u\omega_x + v\omega_x = \frac{1}{Re}\Delta\omega, \tag{8.1}$$

$$\Delta\psi = \omega, \qquad \langle u, v\rangle = \langle -\psi_y, \psi_x\rangle, \tag{8.2}$$

Here ψ is the stream function and $\omega = \nabla \times \mathbf{u} = v_x - u_y$ ($\nabla \times \mathbf{u}$ is a scalar because $\mathbf{u}$ is a 2D vector) is the vorticity. Suppose we use forward Euler to solve (8.1):

$$\omega^{n+1} = \omega^n - u^n\omega_x^n - v^n\omega_x^n + \frac{1}{Re}\Delta\omega^n,$$

then in each time step we need to ψ^n by solving $\Delta\psi^n = \omega^n$ to obtain the velocity by computing $u^n = -\psi_y^n, v^n = \psi_x^n$.

If we use finite difference methods on a rectangular domain, then the linear systems involved the two examples above can surely be solved by the eigenvector method discussed in Chapter 2. However, the eigenvector method can be used for solving a system $Ax = b$ only when the matrix A has a tensore structure like $A = I \otimes B + C \otimes I$, which no longer holds in general. For instance, if we solve a elliptic equation in the form of

$$\nabla \cdot (a(x, y)\nabla u) = f,$$

then any non-constant coefficient $a(x, y)$ will destroy the tensor structure. The following are two quick examples of this kind:

- Consider solving the Poisson equation $u_{xx} + u_{yy} = f$ on a disk. Then we can use the finite difference method in the polar coordinates, under which the disk becomes a rectangle. The Poisson equation $u_{xx}+u_{yy} = f$ in the polar coordinates becomes,

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} = f(r, \theta).$$

- Consider the variable density incompressible Naiver-Stokes equations,

$$\begin{aligned} \rho_t + (u\rho)_x + (v\rho)_y &= 0, \\ \rho(\mathbf{u}_t + (\mathbf{u}\cdot\nabla)\mathbf{u}) + \nabla p - \nu(\rho)\Delta\mathbf{u} &= \mathbf{f}. \end{aligned}$$

To obtain the pressure for evolving $\mathbf{u}$, we can take the divergence of the second equation divided by ρ, then we get an elliptic equation for the pressure

$$\nabla \cdot (\frac{1}{\rho(x, y)}\nabla p) = \nabla \cdot (\mathbf{f} + \frac{1}{\rho}\nu(\rho)\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}).$$

8.1 Linear iterative methods

We discuss the linear iterative methods with matrix splitting for solving the linear system $Au = f$, obtained from discretizing with the centered

difference for the 1D Poisson equation $-u_{xx} = f$ on the interval $[0, 1]$ with homogeneous Dirichlet boundary conditions:

$$-u_{j-1} + 2u_j - u_{j+1} = \Delta x^2 f_j, \tag{8.3}$$

or the 2D Poisson equation $-u_{xx} = f$ on the square $[0, 1] \times [0, 1]$ with homogeneous Dirichlet boundary conditions (assuming $\Delta x = \Delta y$):

$$-u_{i-1,j} - u_{i,j-1} + 4u_{i,j} - u_{i+1,j} - u_{i,j+1} = \Delta x^2 f_{i,j}, \tag{8.4}$$

See Chapter 2 for details. For the 1D case, $A = K$ and the system is

$$\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{n-1} \\ u_n \end{pmatrix} = \Delta x^2 \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix}. \tag{8.5}$$

For the 2D case, assume $\Delta x = \Delta y$, then $A = K \otimes I + I \otimes K$.

Suppose we split the matrix A as $A = B - C$ for any nonsingular matrix B, then the exact solution u to the linear system satisfies:

$$Bu = Cu + f,$$

thus we use the following iterative methods,

$$Bu^{k+1} = Cu^k + f,$$

in which u^k will converge to the exact solution if convergence is guaranteed.

By subtracting the two equations above, the error $e^k = u^k - u$ satisfies,

$$e^{k+1} = B^{-1}Ce^k = Me^k,$$

where $M = B^{-1}C$ is called the iteration matrix. Thus $e^k = M^k e^0$, and the convergence is guaranteed if $\rho(M) < 1$ due to the following fact:

Theorem 8.1. *A square matrix M satisfies $\lim_{k\to\infty} M^k = 0$ if and only if its spectral radius $\rho(M) < 1$.*

If M is a normal matrix, i.e, it can be unitarily diagonalized

$$M = U \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} U^*,$$

then

$$M^k = U \begin{pmatrix} \lambda_1^k & & \\ & \ddots & \\ & & \lambda_n^k \end{pmatrix} U^*,$$

thus $\|M^k\| = \max_k |\lambda_j|^k = \rho(M)^k$, therefore $\|e^k\| \leq \|M^k\|\|e^0\| = \rho(M)^k\|e^0\|$ the convergence rate is $\rho(M)$.

If M is not a normal matrix, then the asymptotic convergence rate is $\rho(M)$ due to the fact:

Theorem 8.2. Gelfand's formula. *Any square matrix M satisfies*

$$\lim_{k\to\infty} \|M^k\|^{\frac{1}{k}} = \rho(M).$$

8.1.1 Jacobi and weighted Jacobi iterations

Let D be a diagonal matrix denoting the diagonal part of the matrix A, L be a lower triangular matrix denoting the lower triangular part of the matrix $-A$, U be a upper triangular matrix denoting the upper triangular part of the matrix $-A$. In other words, assume $A = D - L - U$. The Jacobi iteration for solving $Au = f$ is defined as

$$Du^{k+1} = (L + U)u^k + f.$$

For the 1D Poisson scheme (8.3), the Jacobi iteration is equivalent to

$$-u_{j-1}^k + 2u_j^{k+1} - u_{j+1}^k = \Delta x^2 f_j.$$

It can be implemented an iteration of

$$u_j^{k+1} = \frac{1}{2}u_{j-1}^k + \frac{1}{2}u_{j+1}^k + \frac{1}{2}\Delta x^2 f_j, \tag{8.6}$$

and the 2D case is

$$u_{i,j}^{k+1} = \frac{1}{4}u_{i,j-1}^k + \frac{1}{4}u_{i,j+1}^k + \frac{1}{4}u_{i-1,j}^k + \frac{1}{4}u_{i+1,j}^k + \frac{1}{4}\Delta x^2 f_{i,j}.$$

Now let us compute the spectral radius of the iteration matrix $M = D^{-1}(D - A)$ for the second finite difference scheme on a $N \times N$ mesh solving the 2D Poisson equation with zero boundary conditions, in which $A = K \otimes I + I \otimes K$ and $D = 4I \otimes I$. We have $M = D^{-1}(D - A) = I - \frac{1}{4}A$ thus the eigenvalues (see Chapter 2 for the eigenvalues of A) are

$$\begin{aligned}\lambda_{i,j}(M) = 1 - \frac{1}{4}\lambda_{i,j}(A) &= 1 - \frac{1}{4}\left[2 - \cos(i\frac{\pi}{N+1}) + 2 - \cos(j\frac{\pi}{N+1})\right] \\ &= \frac{1}{2}\cos(i\frac{\pi}{N+1}) + \frac{1}{2}\cos(j\frac{\pi}{N+1}).\end{aligned}$$

Thus $\rho(M) = \cos\theta$ with $\theta = \frac{1}{N+1}$. Similarly for the 1D case $\lambda_j(M) = \cos(j\frac{\pi}{N+1}) = \cos(j\theta)$. In particular the convergence rate for large N is

$$\rho(M) = \cos\theta \approx 1 - \frac{1}{2}\theta^2 = 1 - \frac{1}{2}\frac{\pi^2}{(N+1)^2} \approx 1 - cN^{-2},$$

which means that the convergence is slower for larger system.

The Jacobi iteration can be regarded as solving a time-dependent heat equation till the steady state. If using the forward Euler for the equation $u_t = u_{xx} + f$, we get

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{\Delta x^2}(u_{j-1}^n - 2u_j^n + u_{j+1}^n) + \Delta t f_j,$$

where becomes (8.6) if $\Delta t = \frac{1}{2}\Delta x^2$ (the Lax-Richtmyer stability requires the amplification factor $|g(\xi)| \leq 1$ in forward Euler solving $u_t = u_{xx}$, which implies $\Delta t \leq \frac{1}{2}\Delta x^2$).

The weighted Jacobi iteration is to use the splitting $A = D/w - (D/w - A)$ where w is a constant parameter/weight, which results in the iteration

$$D/w u^{k+1} = (D/w - A)u^k + f,$$

and its iteration matrix is $M = (D/w)^{-1}(D/w - A) = I - wD^{-1}A$. Notice that $w = 1$ is the original Jacobi iteration. For the 1D Poisson equation case, we have $\lambda_j(M) = 1 - \frac{1}{2}w\lambda_j(A) = 1 - w + w\cos(j\theta)$ with $\theta = \frac{\pi}{N+1}$. For the comparison of the two eigenvalues, see Figure 8.1, where $|\lambda_j(M)|$ in the weighted Jacobi is smaller for large j (meaning faster convergence for high frequencies) and $|\lambda_j(M)|$ in the weighted Jacobi is larger for small j (meaning slower convergence for low frequencies).

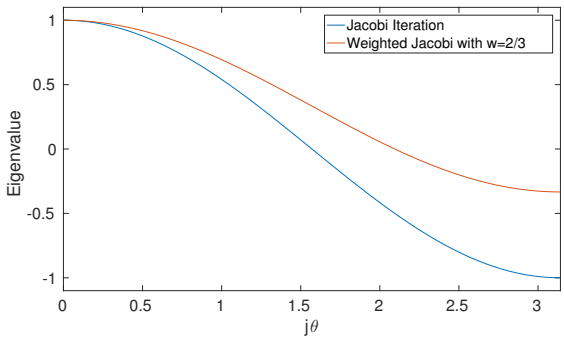


Figure 8.1: The eigenvalues of iteration matrices in the Jacobi iteration and the weighted Jacobi iteration with $w = \frac{2}{3}$.

8.1.2 Gauss-Seidel iteration

Suppose $A = D - L - U$ as before, then the Gauss-Seidel iteration is

$$(D - L)u^{k+1} = Uu^k + f,$$

where the system $(D-L)x = b$ can be easily solved by forward substitution since $D-L$ is lower triangular. For the 1D Poisson scheme (8.3), the Gauss-Seidel iteration is conceptually

$$-u_{j-1}^{k+1} + 2u_j^{k+1} - u_{j+1}^k = \Delta x^2 f_j,$$

and in 2D it is

$$-u_{i-1,j}^{k+1} - u_{i,j-1}^{k+1} + 4u_{i,j}^{k+1} - u_{i+1,j}^k - u_{i,j+1}^k = \Delta x^2 f_{i,j}.$$

To find the eigenvalues for $M = (D - L)^{-1}U$, assume λ and v form a eigenvalue-eigenvector pair for M, then $Mv = \lambda v$ thus

$$Uv = \lambda(D - L)v,$$

$$\lambda Dv = \lambda Lv + Uv,$$

which in 2D is equivalent to

$$4\lambda v_{i,j} = \lambda v_{i-1,j} + \lambda v_{i,j-1} + v_{i+1,} + v_{i,j+1}.$$

Consider a change of variable by introducing a vector w so that $v_{i,j} = \lambda^{\frac{i+j}{2}} w_{i,j}$, then w satisfies

$$4\lambda^{\frac{i+j+2}{2}} w_{i,j} = \lambda^{\frac{i+j+1}{2}} w_{i-1,j} + \lambda^{\frac{i+j+1}{2}} w_{i,j-1} + \lambda^{\frac{i+j+1}{2}} w_{i+1,j} + \lambda^{\frac{i+j+1}{2}} w_{i,j+1}.$$

Therefore the vector w satisfies

$$4\lambda^{\frac{1}{2}} w_{i,j} = w_{i-1,j} + w_{i,j-1} + w_{i+1,} + w_{i,j+1},$$

whose matrix form is precisely $\lambda^{\frac{1}{2}} Dw = (L+U)w$, i.e., $D^{-1}(L+U)w = \lambda^{\frac{1}{2}} w$. Since $D^{-1}(L+U)$ is the iteration matrix in the Jacobi iteration, this means that $\lambda(M_{GS}) = \lambda(M_{Jacobi})^2$. Thus $\rho(M_{GS}) = \cos^2\theta$, which means that the Gauss-Seidel iteration is twice as fast as the Jacobi iteration.

The Gauss-Seidel iteration can also be regarded as a scheme solving a time-dependent equation. Consider the equation

$$u_t = u_{xx} - \varepsilon u_{xt} + f,$$

and the scheme

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} - \varepsilon \frac{(u_j^{n+1} - u_{j-1}^{n+1}) - (u_j^n - u_{j-1}^n)}{\Delta x \Delta t} + f_j,$$

which is accurate around $t = (n + \frac{1}{2})\Delta t$ and $x = x_j$. If setting $\Delta t = \Delta x^2$ and $\varepsilon = \Delta x$, the scheme becomes the Gauss-Seidel iteration.

8.1.3 SOR

Another popular iteration method is successive overrelaxation (SOR), which is a combination of Jacobi and Gauss-Seidel. The matrix splitting is $A = \frac{1}{w}(D - wL) - \frac{1}{w}[(1-w)D + wU]$, thus

$$\frac{1}{w}(D - wL)u^{k+1} = \frac{1}{w}[(1-w)D + wU]u^k + f,$$

which is equivalent to

$$Du^{k+1} = Du^k + w(Lu^{k+1} + Uu^k - Du^k) + wf.$$

For the 2D Poisson scheme, this is

$$4u_{i,j}^{k+1} = 4u_{i,j}^k + w(u_{i,j-1}^{k+1} + u_{i-1,j}^{k+1} + u_{i,j+1}^k + u_{i+1,j}^k - 4u_{i,j}^k) + w\Delta x^2 f_{i,j}.$$

There are ways to choose w to improve the spectral radius from $\rho(M) = 1 - cN^{-2}$ to $\rho(M) = 1 - cN^{-1}$.

8.2 Steepest descent

Consider any linear system $Ax = b$ where A is a real square matrix and x and b are real vectors of size n. It is equivalent to solving a minimization problem:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|^2.$$

The method of gradient descent is the simplest first order method for solving a minimization problem $\min_{x \in \mathbb{R}^n} f(x)$:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k),$$

where $\alpha_k > 0$ is the optimal step size to be chosen to guarantee convergence. The steepest descent method is to use the optimal step size α_k along the search direction $-\nabla f(x_k)$. Recall that the function ascent/descent the most along the positive/negative gradient: the directional derivative along a unit vector $\mathbf{u}$ is

$$D_{\mathbf{u}} f(x) = \nabla f(x) \cdot \mathbf{u} = \|\nabla f(x)\|\|\mathbf{u}\| \cos\theta,$$

where θ is the angle between two vectors $\nabla f(x)$ and $\mathbf{u}$, thus $D_{\mathbf{u}} f(x)$ attends its maximum (minimum) at $\theta = 0$ ($\theta = \pi$). First order here refers to the fact that only the gradient $\nabla f(x)$ is used in the algorithm. A typical second order method is the Newton's method: for example, suppose $n = 1$, if the minimizer of $f(x)$ exists (a convex function $f(x)$ suffices for the existence, and convex function means that $f''(x) \geq 0$), it must be a critical point of $f(x)$, thus we can solve $f'(x) = 0$ instead, and the second order derivative $f''(x)$ is needed in Newton's method solving $f'(x) = 0$.

To simplify the discussion, from now on, we assume A is real symmetric and positive semi-definite (if the coefficient matrix in $Ax = b$ is not symmetric or positive definite, then we consider solving $A^TAx = A^Tb$ instead). For $Ax = b$, define the cost function

$$f(x) = \frac{1}{2}x^TAx - x^Tb,$$

and its gradient is

$$\nabla f(x) = Ax - b.$$

The function $f(x)$ is convex because $\nabla^2 f(x) = A \geq 0$, which guarantees that the minimizer of $f(x)$ must be a solution to $\nabla f(x) = 0$.

In the steepest descent method $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$, the step size α_k can be chosen so that $f(x_{k+1})$ is the smallest:

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x_k - \alpha \nabla f(x_k)).$$

Let $\phi(\alpha) = f(x_k - \alpha \nabla f(x_k))$. Let r_k denote $-\nabla f(x_k) = b - Ax_k$, which is also called residue. Then

$$\begin{aligned}
\phi(\alpha) &= f(x_k + \alpha r_k) \\
&= \frac{1}{2}x_k^TAx_k + \alpha r_k^TAx_k + \frac{1}{2}\alpha^2 r_k^TAr_k - x_k^Tb - \alpha r_k^Tb \\
&= \frac{1}{2}x_k^TAx_k - x_k^Tb + \alpha r_k^T(Ax_k - b) + \frac{1}{2}\alpha^2 r_k^TAr_k \\
&= f(x_k) - \alpha r_k^Tr_k + \frac{1}{2}\alpha^2 r_k^TAr_k,
\end{aligned}$$

which is quadratic in α thus attains its minimum at the critical point. By setting $\phi'(\alpha) = 0$, we get

$$\alpha_k = \arg\min_{\alpha \geq 0} \phi(\alpha) = \frac{r_k^Tr_k}{r_k^TAr_k}.$$

The steepest descent method can be implemented as iterations of the following steps:

1. $r_k = b - Ax_k$.
2. $\alpha_k = \frac{r_k^Tr_k}{r_k^TAr_k}$.
3. $x_{k+1} = x_k + \alpha_k r_k$.

We can use the eigenvectors v_j and eigenvalues λ_j of A to undertand the convergence of the steepest descent method. Define the error $e_k = x_k - x$, then $r_k = b - Ax_k = Ax - Ax_k = -Ae_k$ and we have:

$$e_{k+1} = e_k + \alpha_k r_k.$$

Since A is real symmetric, we can choose orthonormal eigenvectors v_j ($j = 1, \cdots, n$) which can span the whole space $\mathbb{R}^n$. The error can be expressed as linear combinations of v_j:

$$e_k = \sum_{j=1}^{n} a_j v_j.$$

We also have

$$\begin{aligned}
r_k = -Ae_k = -\sum_{j=1}^{n} \xi_j A v_j = -\sum_{j=1}^{n} a_j \lambda_j v_j, \\
e_k^T A e_k = (\sum_{j=1}^{n} a_j v_j^T)(\sum_{j=1}^{n} a_j \lambda_j v_j) = \sum_{j=1}^{n} a_j^2 \lambda_j, \\
r_k^T r_k = \sum_{j=1}^{n} a_j^2 \lambda_j^2, \\
r_k^T A r_k = \sum_{j=1}^{n} a_j^2 \lambda_j^3
\end{aligned}$$

Assume $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$, then (or we can simply use the Courant-Fischer-Weyl min-max principle to obtain the following),

$$\lambda_1 r_k^T r_k \leq r_k^T A r_k \leq \lambda_n r_k^T r_k,$$

and also

$$\lambda_1 e_k^T A e_k \leq r_k^T r_k \leq \lambda_n e_k^T A e_k.$$

Define the energy norm $\|e\|_A = (e^T A e)^{\frac{1}{2}}$ which is easier to work with than the Euclidean norm. Then

$$\begin{aligned}
\|e_{k+1}\|_A^2 &= e_{k+1}^T A e_{k+1} \\
&= (e_k + \alpha_k r_k)^T A (e_k + \alpha_k r_k) \\
&= e_k^T A e_k + 2\alpha_k r_k^T A e_k + \alpha_k^2 r_k^T A r_k \\
&= e_k^T A e_k - 2\frac{r_k^T r_k}{r_k^T A r_k} r_k^T r_k + \left(\frac{r_k^T r_k}{r_k^T A r_k}\right)^2 r_k^T A r_k \\
&= e_k^T A e_k - \frac{(r_k^T r_k)^2}{r_k^T A r_k} \\
&= \|e_k\|_A^2 \left(1 - \frac{(r_k^T r_k)^2}{r_k^T A r_k e_k^T A e_k}\right) \\
&= \|e_k\|_A^2 w,
\end{aligned}$$

where

$$w = 1 - \frac{r_k^T r_k}{r_k^T A r_k} \frac{r_k^T r_k}{e_k^T A e_k} \leq 1 - \frac{1}{\lambda_n}\lambda_1 = 1 - \frac{\lambda_1}{\lambda_n}.$$

The condition number for a positive definite real symmetric matrix A is $\kappa = \frac{\lambda_n}{\lambda_1}$, thus $w \leq \frac{\kappa-1}{\kappa}$. Therefore,

$$\|e_k\|_A \leq \sqrt{\frac{\kappa-1}{\kappa}}\|e_{k-1}\|_A \leq \left(\frac{\kappa-1}{\kappa}\right)^{\frac{k}{2}} \|e_0\|_A.$$

Recall that $Ax = b$, therefore $x^T Ax = x^T b$ thus

$$x^T Ax = -2(\frac{1}{2}x^T Ax - x^T b) = -2f(x).$$

For a symmetric A, we have

$$\begin{aligned}\|e_k\|_A^2 &= (x_k - x)^T A(x_k - x) = x_n^T Ax_k - x^T Ax_k - x_k^T Ax + x^T Ax \\ &= x_k^T Ax_k - b^T x_k - x_k^T b + x^T Ax = 2f(x_k) - 2f(x).\end{aligned}$$

We can summarize the convergence rate of steepest descent for linear system as the following:

Theorem 8.3. *Let A be a positive definite matrix with eigenvalues $0 < \lambda 1_1 \leq \cdots \leq \lambda_n$. For steepest descent minimizing a quadratic function $f(x) = \frac{1}{2}x^T Ax - x^T b$, the convergence rate is linear (exponentially fast):*

$$\|e_{k+1}\|_A \leq \sqrt{\frac{\lambda_n - \lambda_1}{\lambda_n}}\|e_k\|_A,$$

and

$$f(x_{k+1}) - f(x_*) \leq \left(1 - \frac{\lambda_1}{\lambda_n}\right)[f(x_k) - f(x_*)].$$

Remark 8.1. *The steepest descent has an exponential convergence rate $\sqrt{1-\frac{1}{\kappa}}$, which is slow in practice for large κ from large matrices, e.g., solving the Poisson equation in 2D/3D.*

8.3 The Conjugate Gradient method

We still assume A is real symmetric and $A > 0$. In practice, the steepest descent method is simple to apply however very slow. The conjugate gradient method is a faster popular choice. The conjugate gradient method can be regarded as an acceleration of steepest descent:

$$x_{k+1} = x_k + \alpha_k(r_k + \gamma_k(x_k - x_{k-1})),$$

where α_k and γ_k are parameters. This formula shows that the new change in position, $x_{k+1} - x_k$, is a linear combination of the steepest descent direction and the previous change in position $x_k - x_{k-1}$. It can be rewritten as

$$x_{k+1} = x_k + \alpha_k p_k,$$

where the search direction p_k is

$$p_k = r_k + \gamma_k(x_k - x_{k-1}) = r_k + \gamma_k\alpha_{k-1}p_{k-1} = r_k + \beta_{k-1}p_{k-1}.$$

These formulas can be summarized as

$$x_{k+1} = x_k + \alpha_k p_k \tag{8.7}$$
$$r_{k+1} = r_k - \alpha_k A p_k \tag{8.8}$$
$$p_{k+1} = r_{k+1} + \beta_k p_k. \tag{8.9}$$

We still need to determine the initial search direction p_0 and the parameters α and β. Suppose p_k is known, then we can ask for a α_k so that $f(x_k+\alpha_k p_k)$ is the smallest. Minimizing the function $\phi(\alpha) = f(x_k+\alpha p_k) = f(x_k) - \alpha r_k^T p_k + \frac{1}{2}\alpha^2 p_k^T A p_k$ gives us the best α_k:

$$\alpha_k = \frac{p_k^T r_k}{p_k^T A p_k}. \tag{8.10}$$

Using this optimal α_k, we have

$$f(x_{k+1}) = f(x_k) - \frac{(p_k^T r_k)^2}{p_k^T A p_k}.$$

From the formula above, we can see that $p_0 = r_0$ will guarantee $f(x_1) < f(x_0)$. Now we assume $p_0 = r_0$ which will imply other useful properties. By (8.8) and (8.10), we get

$$p_k^T r_{k+1} = p_k^T r_k - \alpha_k p_k^T A p_k = 0.$$

Together with (8.9), we get

$$p_{k+1}^T r_{k+1} = r_{k+1}^T r_{k+1} + \beta_k p_k^T r_{k+1} = r_{k+1}^T r_{k+1}, \quad k \geq 0.$$

Notice that $p_0 = r_0$ ensures $p_k^T r_k = r_k^T r_k$ for $k \geq 0$. Thus (8.10) becomes

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k},$$

and we have

$$f(x_{k+1}) = f(x_k) - \frac{(r_k^T r_k)^2}{p_k^T A p_k}.$$

Next we choose some β to minimize $p_k^T A p_k$ thus to minimize $f(x_{k+1})$. Since (8.9) implies

$$p_k^T A p_k = r_k^T A r_k + 2\beta_{k-1} r_k^T p_{k-1} + \beta_{k-1}^2 p_{k-1}^T A p_{k-1},$$

the best choice of β_{k-1} is,

$$\beta_{k-1} = -\frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}, \quad k \geq 1,$$

or equivalently

$$\beta_k = -\frac{r_{k+1}^T A p_k}{p_k^T A p_k}, \quad k \geq 0,$$

which with (8.9) implies

$$p_{k+1}^T A p_k = r_{k+1}^T p_k + \beta_k p_k^T A p_k = 0. \tag{8.11}$$

The property $p_{k+1}^T A p_k = 0$ means that the search direction p_{k+1} is A-orthogonal to the previous one p_k. In other words, p_{k+1} is conjugate to p_k. By (8.11) and (8.9), we get

$$p_k^T A p_k = r_k^T A p_k + \beta_{k-1} p_{k-1} A p_k = r_k^T A p_k,$$

which with (8.8) and (8.10) implies,

$$r_{k+1}^T r_k = r_k^T r_k - \alpha_k p_k^T A r_k = 0.$$

Therefore, by (8.8), we have

$$r_{k+1}^T r_{k+1} = r_{k+1}^T r_k - \alpha_k r_{k+1}^T A p_k = -\alpha_k r_{k+1}^T A p_k,$$

thus the formula for β becomes

$$\beta_k = -\frac{r_{k+1}^T A p_k}{p_k^T A p_k} = \frac{1}{\alpha_k} \frac{r_{k+1}^T r_{k+1}}{p_k^T A p_k} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}.$$

Now we summarize the formula for the conjugate gradient method:

$$p_0 = r_0 = b - A x_0, \tag{8.12a}$$

$$\alpha_k = \frac{\|r_k\|^2}{p_k^T A p_k}, \tag{8.12b}$$

$$x_{k+1} = x_k + \alpha_k p_k, \tag{8.12c}$$

$$r_{k+1} = r_k - \alpha_k A p_k, \tag{8.12d}$$

$$\beta_k = \frac{\|r_{k+1}\|^2}{\|r_k\|^2} \tag{8.12e}$$

$$p_{k+1} = r_{k+1} + \beta_k p_k. \tag{8.12f}$$

The following important property of conjugate gradient method can be shown by induction:

Theorem 8.4. *For the conjugate gradient method defined above, the search direction p_k and the residue r_k satisfies:*

$$r_k^T r_j = p_k^T A p_j = 0, \quad \forall k \neq j.$$

An implication of this property is that the conjugate gradient method in theory converges in at most n steps for a $n \times n$ matrix, because $r_j \in \mathbb{R}^n$ and $\mathbb{R}^n$ can have at most n linear dependent vectors. On the other hand, the iteration (8.12) is unstable subject to round-off errors, thus it will never give the exact solution to the linear system, though usually (8.12) is still a good choice for finding an approximate solution to certain accuracy.

Remark 8.2. *Notice that we have assumed A is real symmetric positive definite. For solving $Ax = b$ with a real symmetric positive semi-definite matrix A, e.g., the Poisson equation with purely Neumann b.c., then* (8.12) *might diverge if $Ax = b$ does not have a solution. Notice that $f(x) = \frac{1}{2}x^T Ax - b^T x$ always has minimizers for $A \geq 0$, even if $Ax = b$ is an inconsistent linear system.*

In practice, usually it is not affordable to have n iteration steps thus it is important to analyze the convergence rate, for which we have

Theorem 8.5. *If $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are eigenvalues of the positive definite matrix A, then the error in the conjugate gradient method satisfies*

$$\|e_k\|_A \leq 2\left(\frac{\sqrt{\lambda_n} - \sqrt{\lambda_1}}{\sqrt{\lambda_n} + \sqrt{\lambda_1}}\right)^k \|e_0\|_A.$$

The proofs of these two theorems can be found in [12]. A weaker estimate is given by the condition number κ:

$$\|e_k\|_A \leq 2\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^k \|e_0\|_A.$$

The dominating operations in each iteration of either steepest descent or conjugate gradient are matrix-vector multiplications. For many problems, A is sparse and one matrix-vector multiplication requires $\mathcal{O}(m)$ operations where m is the number of nonzero entries in A.

Suppose we wish to reduce the norm of the error by a factor of ε, which is to achieve $\|e_k\|_A \leq \varepsilon \|e_0\|_A$. Then by the convergence rate, the maximum number of iterations required in steepest descent is

$$k \leq \left\lceil \frac{1}{2}\kappa \ln(\frac{1}{\varepsilon}) \right\rceil,$$

and the maximum number of iterations in CG is

$$k \leq \left\lceil \frac{1}{2}\sqrt{\kappa} \ln(\frac{2}{\varepsilon}) \right\rceil.$$

So the time complexity of steepest descent is $\mathcal{O}(m\kappa)$ and the time complexity of CG is $\mathcal{O}(m\sqrt{\kappa})$. Both have the space complexity $\mathcal{O}(m)$.

For the second order finite difference scheme solving the Poisson equation with Dirichlet boundary conditions on $[0,1]$ with n grid points in 1D (on $[0,1]\times[0,1]$ with $n = N \times N$ grid points in 2D; on $[0,1]\times[0,1]\times[0,1]$ with $n = N \times N \times N$ grid points in 3D), the condition number of the coefficient A is $\kappa = \frac{1-\cos(N\theta)}{1-\cos\theta}$ where $\theta = \pi\frac{1}{N+1}$. We have

$$\kappa = \frac{1-\cos(N\theta)}{1-\cos\theta} = \frac{1+\cos(\theta)}{1-\cos\theta} \approx \frac{1+1+\frac{1}{2}\theta^2}{1-1+\frac{1}{2}\theta^2} = \mathcal{O}(N^2) = \mathcal{O}(n^{\frac{2}{d}}),$$

where d is the dimension. Therefore, steepest descent has complexity $\mathcal{O}(n^2)$ and CG has complexity $\mathcal{O}(n^{\frac{3}{2}})$ for 2D problems, and steepest descent has complexity $\mathcal{O}(n^{\frac{5}{3}})$ and CG has complexity $\mathcal{O}(n^{\frac{4}{3}})$ for 3D problems.

The multigrid method for the elliptic problems has complexity $\mathcal{O}(n)$ in any dimension.

8.4 Multigrid methods

In this section we only consider the linear system $Au = b$ obtained in the finite difference scheme for the Poisson equation. The (weighted) Jacobi iteration (and Gauss-Seidel) produce smooth errors. The high frequencies error vector e can be nearly removed in a few iterations, e.g., see Figure 8.1. But low frequencies are reduced very slowly, and convergence requires $\mathcal{O}(n^2)$ iterations, which is unacceptable. The multigrid idea is to change to a coarser grid, on which "smooth becomes rough" and low frequencies act like higher frequencies.

On coarser grids a big piece of the error is removable. We iterate a few times then change from fine to coarse, and coarse to fine. The multigrid method can solve many sparse systems to high accuracy in a fixed number of iterations, not growing with n.

8.4.1 Interpolation and restriction

Now consider solving $-u_{xx} = f$ on $[0,1]$ with homogeneous Dirichlet boundary conditions, i.e., the system (8.5). The key steps in a multigrid method are the two matrices R and I (in this section I denotes the interpolation matrix rather than the identity unless otherwise specified):

- A **restriction matrix** $R = R_h^{2h}$ transfers vectors from the fine grid with grid size $h = \Delta x$ to the coarse grid with size $2h$.
- An **interpolation matrix** $I = I_{2h}^h$ returns to the fine grid from the coarse grid.

- The original matrix on the fine grid is denoted as A_h, which is approximated by $A_{2h} = RA_hI$ on the coarse grid.

To see an example of R and I, suppose $h = \frac{1}{8}$ and $2h = \frac{1}{4}$, then the coarse grid has three grid points and the fine grid has seven grid points, for which the matrix R has size 3×7 and I has size 7×3. We use the following simple linear interpolation interpolation:

$$I\mathbf{v} = \mathbf{u}: \quad \frac{1}{2}\begin{pmatrix} 1 & & \\ 2 & & \\ 1 & 1 & \\ & 2 & \\ & 1 & 1 \\ & & 2 \\ & & 1 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} v_1/2 \\ v_1 \\ v_1/2 + v_2/2 \\ v_2 \\ v_2/2 + v_3/2 \\ v_3 \\ v_3/2 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{pmatrix},$$

where $\mathbf{v}$ is defined on the coarse grid and $\mathbf{u}$. In other words, we *linear interpolation* for the in-between values u_1, u_3, u_5, u_7.

For the restriction matrix, we can simply assign $v_1 = u_2$, $v_2 = u_4$, and $v_3 = u_6$. Another way is to use the full weight operator by setting $R = \frac{1}{2}I^T$:

$$R\mathbf{u} = \mathbf{v}: \quad \frac{1}{4}\begin{pmatrix} 1 & 2 & 1 & & & & \\ & & 1 & 2 & 1 & & \\ & & & & 1 & 2 & 1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{pmatrix} = \begin{pmatrix} (u_1 + 2u_2 + u_3)/4 \\ (u_3 + 2u_4 + u_5)/4 \\ (u_5 + 2u_6 + u_7)/4 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}.$$

The advantages of the full weight operator include

- the matrix RA_hI is still symmetric positive definite.
- $RA_hI = A_{2h}$. See Example 8.1 below.

For the 2D equation $-u_{xx} - u_{yy} = f$ on $[0,1] \times [0,1]$ with homogeneous Dirichlet boundary conditions. Assume $h = \Delta x = \Delta y$. We can use the same linear interpolation then the interpolation matrix is

$$I2D = I \otimes I.$$

For instance, if U denotes a 2D array with $U(j,i)$ denoting the point value at (x_i, y_j) in the mesh. Then

$$I2Dvec(U) = (I \otimes I)vec(U) = vec(IUI^T),$$

which represents the linear interpolation in a dimension by dimension fashion. And the restriction matrix is

$$R2D = R \otimes R = \frac{1}{4}I^T \otimes I^T = \frac{1}{4}(I \otimes I)^T = \frac{1}{4}I2D^T.$$

The coefficient matrix on the fine mesh is $A_h = \frac{1}{h^2}K2D = \frac{K_h}{h^2}\otimes Id + Id \otimes \frac{K_h}{h^2}$, thus

$$\begin{aligned} R2D * A_h * I2D &= (R \otimes R)(\frac{K_h}{h^2} \otimes Id + Id \otimes \frac{K_h}{h^2})(I \otimes I) \\ &= (R\frac{K_h}{h^2}I) \otimes (I * Id * R) + (I * Id * R) \otimes (R\frac{K}{h^2}I) \\ &= \frac{K_{2h}}{(2h)^2} \otimes (I * Id * R) + (I * Id * R) \otimes \frac{K_{2h}}{(2h)^2}, \end{aligned}$$

where we use the fact that $R\frac{K_h}{h^2}I = \frac{K_{2h}}{(2h)^2}$, see Example 8.1. However, $(I * Id * R)$ is not a smaller identity matrix. Therefore $R2D * A_h * I2D = A_{2h}$ is no longer true.

8.4.2 A two-grid V-cycle

We first consider the multigrid method using only two grids. We use the subscript h to denote a notation defined on the grid of size h. The system we want to solve is denoted as

$$A_h u = f_h,$$

and u denotes the exact solution to this system. The iterations on each grid can use Jacobi (or weighted Jacobi with $w = \frac{2}{3}$) or Gauss-Seidel. For the larger problem on the fine grid, iteration converges slowly to the low frequency smooth part of the solution. Let u_h be the Jacobi's solution after a few iterations. The multigrid method transfers the current residue $r_h = f_h - A_h u_h$ to the coarse grid. Define the error as $e = u - u_h$ then it satisfies

$$A_h e = A_h(u - u_h) = b_h - A_h u_h = r_h.$$

We iterate a few times on the coarse $2h$ grid, to approximate the coarse-grid error by E_{2h}, then interpolate back to E_h on the fine grid, and make the correction to $u_h + E_h$.

This fine-coarse-fine loop is a two-grid V-cycle, also called a v-cycle (small v cycle). Here are the steps in one v-cycle:

1. Iterate on $A_h u = b_h$ to reach u_h (e.g., 3 Jacobi or Gauss-Seidel steps). The iteration step is also called relaxation.
2. Restrict the residual $r_h = f_h - A_h u_h$ to the coarse grid by $r_{2h} = R_h^{2h} r_h$.
3. Solve $A_{2h}E_{2h} = r_{2h}$, either by $E_{2h} = A_{2h}^{-1} r_{2h}$ or by 3 Jacobi iterations with initial guess $E = 0$. Here A_{2h} denotes the discretization matrix on the coarser mesh, which happens to be RA_hI in the 1D case.
4. Interpolate E_{2h} back to $E_h = I_{2h}^h E_{2h}$. Add E_h to u_h.
5. Iterate 3 more times by Jacobi's method on $A_h u = f_h$ starting from the improved $u_h + E_h$.

8.4.3 The errors e_h and E_h

Even if we solve the coarse grid equation exactly in step 3 above, the multigrid error correction E_h is not equal to the true fine-grid error $e_h = u - u_h$. But these two errors are related. We can track down steps from E to e.

First, the residue satisfies $r_h = A_h e_h$ thus

$$E_h = IE_{2h} = IA_{2h}^{-1} r_{2h} = I(RA_hI)^{-1} r_{2h} = I(RA_hI)^{-1} Rr_h = I(RA_hI)^{-1} RA_h e_h.$$

So $E_h = Se_h$ and

$$S = I(RA_hI)^{-1} RA_h,$$

where S satisfies

$$S^2 = I(RA_hI)^{-1} RA_h I(RA_hI)^{-1} RA_h = I(RA_hI)^{-1} RA_h = S.$$

So the multigrid correction $E_h = Se_h$ is not the whole error, but a projection of e_h. The new error after a v-cyle is $e_h - E_h = (Id - S)e_h$ where Id denotes the identity matrix. The matrix $Id - S$ a two-grid operator. This matrix $Id - S$ plays the same role as the iteration matrix $M = B^{-1}C$ in the matrix splitting iteration method.

Example 8.1. *Suppose $h = \Delta x = \frac{1}{6}$, then the matrices are*

$$A_h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix},$$

$$R = \frac{1}{4}\begin{pmatrix} 1 & 2 & 1 & & \\ & & 1 & 2 & 1 \end{pmatrix}, \quad I = \frac{1}{2}\begin{pmatrix} 1 & \\ 2 & \\ 1 & 1 \\ & 2 \\ & 1 \end{pmatrix},$$

$$RA_h = \frac{1}{(2h)^2}\begin{pmatrix} 0 & 2 & 0 & -1 & 0 \\ 0 & -1 & 0 & 2 & 0 \end{pmatrix},$$

$$\textbf{Coarse grid matrix}: \quad A_{2h} = RA_hI = \frac{1}{(2h)^2}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Notice that A_{2h} happens to be the discrete Laplacian matrix on the coarse grid, which is still true for smaller h. The matrix S is

$$S = IA_{2h}^{-1} RA_h = \begin{pmatrix} 0 & 1/2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1/2 & 0 \end{pmatrix}.$$

Due to its three columns of zeros, the nullspace of S contains all fine-grid vectors of the form $(e_1, 0, e_3, 0, e_5)^T$, which are vectors that do no appear on the coarse grid. If the error e_h has this form, then $E_h = Se_h$ would be zero, meaning that there is no improvement from the multigrid. Notice that any vector in such a form represents high frequencies. We do not expect a large component of those high frequency vectors in e_h because of the smoothing in (weighted) Jacobi iterations.

8.4.4 High and low frequencies in $\mathcal{O}(n)$ operations

If $Sv = \lambda v$, then $\lambda v = Sv = S^2 v = \lambda^2 v$ thus $\lambda^2 = \lambda$. So the eigenvalues of S must be 0 or 1. In the previous example, the eigenvalues of S are $1, 1, 0, 0, 0$. The eigenvectors reveal what multigrid does:

- $\lambda = 0$. The eigenvectors have the form $(e_1, 0, e_3, 0, e_5)^T$. In this case multigrid makes no changes at all.

- $\lambda = 1$. The two eigenvectors are $(1, 2, 2, 2, 1)^T$ and $(1, 2, 0, -2, -1)^T$. Those have large low-frequency components. If the error e_h are spanned by these two vectors, then $(Id - S)e_h = \mathbf{0}$, which is perfect. Such errors are not exactly sines but a large part of the low-frequency error is removed.

In other words, the Jacobi iteration handles the high frequencies and multigrid handles the low frequencies. In practice, we do not exactly solve $A_{2h}E_{2h} = r_{2h}$. But it can be shown that a multigrid cycle with good smoothing can reduce the error by a constant factor ρ that is independent of h. A typical value is $\rho = 0.1$ while it might be $\rho = 0.99$ in Jacobi's method. We can achieve a given relative accuracy in a fixed number of cycles. Since each step of each cycle requires only $\mathcal{O}(n)$ operations on sparse problems of size n, multigrid is an $\mathcal{O}(n)$ algorithm, independent of the dimension of the problem.

For solving the second order finite difference equation for the Poisson equation, instead of achieving a given relative accuracy, we may want a solution with accuracy $\mathcal{O}(h^2) = \mathcal{O}(N^{-2})$ which matches the discretization error. In this case we need more than a fixed number of v-cycles. To reach $\rho^k = (N^{-2})$ we need $k = \mathcal{O}(\log N)$ cycles.

The two-grid v-cycle extends to a natural way to more grids. It can go down to coarser grids $2h, 4h, 8h$ and back up to $4h, 2h, h$. This nested sequence of v-cycles is a V-cycle. The W-cycle stays coarse longer, which is generally superior to a V-cycle. The full multigrid cycle is asymptotically better than V or W. The full multigrid stars on the coarsest grid. The operation counts of the full multigrid is $\mathcal{O}(n)$ even for the higher required accuracy $e = \mathcal{O}(h^2)$.

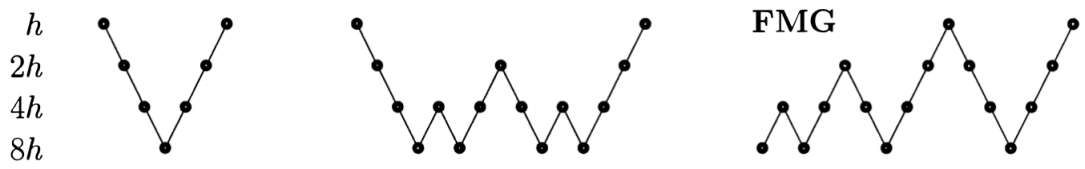


Figure 8.2: V-cycle, W-cycle and the full multigrid (FMG).

We can use the weighted Jacobi with $w = \frac{2}{3}$ for the iteration/relaxation. One V-cycle on the grid of size h can be denoted as the following operator (inputs are an initial guess u_h and the right hand side data f_h, the output is an improved u_h by the V-cycle).

$$\textbf{V-Cycle}: \qquad u_h \leftarrow \mathrm{V}^h(u_h, f_h)$$

- Iterate/Relax $A_h u = f_h$ three times with initial guess u_h.
- Compute restriction $f_{2h} = R_h^{2h} r_h$.
  - Iterate/Relax $A_{2h}E = f_{2h}$ three times with initial guess $E_{2h} = 0$.
  - Compute restriction $f_{4h} = R_{2h}^{4h} r_{2h}$.
    - Iterate/Relax $A_{4h}E = f_{4h}$ three times with initial guess $E_{4h} = 0$.
    - Compute restriction $f_{8h} = R_{4h}^{8h} r_{4h}$.
      - Solve $A_{8h}E = f_{8h}$ (exactly).
    - Correct $E_{4h} \leftarrow E_{4h} + I_{8h}^{4h} E_{8h}$.
    - Iterate/Relax $A_{4h}E = f_{4h}$ three times with initial guess E_{4h}.
  - Correct $E_{2h} \leftarrow E_{2h} + I_{4h}^{2h} E_{4h}$.
  - Iterate/Relax $A_{2h}E = f_{2h}$ three times with initial guess E_{2h}.
- Correct $u_h \leftarrow u_h + I_{2h}^{h} E_{2h}$.
- Iterate/Relax $A_h u = f_h$ three times with initial guess u_h.

We have used coarse grids to obtain improved initial guesses for fine-grid problems. In looking at the V-cycle, we might ask how to obtain an informed initial guess for the first fine-grid relaxation. Nested iteration would suggest solving a problem on a coarse of size $2h$. But how can we obtain a good initial guess for the size $2h$ problem? Nested iteration sends us to $4h$. Clearly, we are on another recursive path that leads to the coarsest grid. The algorithm that joins nested iteration with the V-cycle is called the full multigrid V-cycle (FMG). We can initialize the coarse-grid right sides

by transferring f_h from the fine grid. Another option is to use the original right-side function $f(x)$ sampled on coarse grids.

f

$$\textbf{FMG V-Cycle}: \qquad u_h \leftarrow \text{FMG}^h(f_h)$$

Initiate $f_{2h} = R_h^{2h} f_h, f_{4h} = R_{2h}^{4h} f_{2h}, f_{8h} = R_{8h}^{4h} f_{4h}$ (or simply sample $f(x)$ on these coarser grids).

· Solve exactly or iterate/relax $A_{8h}u_{8h} = f_{8h}$.

∗ Set $u_{4h} = I_{8h}^{4h} u_{8h}$.

∗ $u_{4h} \leftarrow V^{4h}(u_{4h}, f_{4h})$.

– Set $u_{2h} = I_{4h}^{2h} u_{4h}$.

– $u_{2h} \leftarrow V^{2h}(u_{2h}, f_{2h})$.

- Set $u_h = I_{2h}^h u_{2h}$.
- $u_h \leftarrow V^h(u_h, f_h)$.

Read [1] for more details.

Remark 8.3. *The FMG V-cycle should be used iteratively as an iterative solver. In order to do so, we can apply the FMG V-cycle to the error equation $A_h e_h = r_h$ instead of $A_h u = f_h$. In other words, we can implement it as an iteration consisting of three steps:*

1. $r_h = f_h - A_h u_h$
2. $E_h \leftarrow FMG^h(r_h)$
3. *Correction* $u_h \leftarrow u_h + E_h$

8.5 Preconditioned Conjugate Gradient

In practice, for solving $Ax = b$ we can consider solving instead an equivalent system

$$PAP^T y = Pb$$

where P is a *preconditioner* matrix. Recall that the performance of Conjugate Gradient method solving $Ax = b$ depends on the condition number $\kappa(A)$. If we can find a nonsingular matrix P such that $\kappa(PAP^T) \ll \kappa(A)$, then we can more efficiently solve the system $PAP^T y = Pb$ with CG then find x by $x = P^T y$.

For instance, if $A = LL^T$ is the Cholesky factorization (L is lower triangular), then $P = L^{-1}$ is the most ideal preconditioner, because $PAP^T = I$ thus $\kappa(P^1 AP^T) = 1$. The full Cholesky factorization costs $\mathcal{O}(n^3)$ which is as expensive as solving $Ax = b$ by Gaussian elimination. On the other hand, we can use a cheaper approximate Cholesky factorization to construct

a preconditioner, i.e., if $A \approx \tilde{L}\tilde{L}^T$ ($\tilde{L}$ is lower triangular) then use $P = \tilde{L}^{-1}$. For a sparse coefficient matrix $A = LL^T$, the matrix L may have non-zero fill-in for the zero entries of A. The incomplete Cholesky factorization will have zero fill-in, returning an approximation $A \approx \tilde{L}\tilde{L}^T$.

By applying the conjugate gradient method to $PAP^T y = Pb$, we get the following algorithm:

$$\begin{aligned}
\bar{p}_0 &= \bar{r}_0 = Pb - PAP^T y_0, \\
\bar{\alpha}_k &= \frac{\|\bar{r}_k\|^2}{\bar{p}_k^T PAP^T \bar{p}_k}, \\
y_{k+1} &= y_k + \bar{\alpha}_k \bar{p}_k, \\
\bar{r}_{k+1} &= \bar{r}_k - \bar{\alpha}_k PAP^T \bar{p}_k, \\
\bar{\beta}_k &= \frac{\|\bar{r}_{k+1}\|^2}{\|\bar{r}_k\|^2} \\
\bar{p}_{k+1} &= \bar{r}_{k+1} + \bar{\beta}_k \bar{p}_k.
\end{aligned}$$

With the change of variables $x = P^T y$ and $p = P^T \bar{p}$, multiplying certain rows by either P or P^T, we get

$$\begin{aligned}
p_0 &= P^T \bar{r}_0 = P^T Pb - P^T PAx_0, \\
\bar{\alpha}_k &= \frac{\|\bar{r}_k\|^2}{p_k^T A p_k}, \\
x_{k+1} &= x_k + P^T \bar{\alpha}_k P^{-T} p_k = x_k + \bar{\alpha}_k p_k, \\
\bar{r}_{k+1} &= \bar{r}_k - \bar{\alpha}_k PAp_k = \bar{r}_k - P\bar{\alpha}_k Ap_k, \\
\bar{\beta}_k &= \frac{\|\bar{r}_{k+1}\|^2}{\|\bar{r}_k\|^2} \\
p_{k+1} &= P^T \bar{r}_{k+1} + P^T \bar{\beta}_k P^{-T} p_k = P^T \bar{r}_{k+1} + \bar{\beta}_k p_k.
\end{aligned}$$

Now introduce new variables $r_0 = b - Ax_0$ and $r = P^{-1}\bar{r}$, then we get

$$\begin{aligned}
r_0 &= b - Ax_0, \quad p_0 = P^T P r_0 \\
\bar{\alpha}_k &= \frac{r_k^T P^T P r_k}{p_k^T A p_k}, \\
x_{k+1} &= x_k + \bar{\alpha}_k p_k, \\
r_{k+1} &= r_k - \bar{\alpha}_k Ap_k, \\
\bar{\beta}_k &= \frac{r_{k+1}^T P^T P r_{k+1}}{r_k^T P^T P r_k}, \\
p_{k+1} &= P^T P r_{k+1} + \bar{\beta}_k p_k,
\end{aligned}$$

Finally, let $M = P^{-1}[P^{-1}]^T$ and $z = M^{-1}r = P^T Pr$. Set $\bar{\alpha} = \alpha$ and $\bar{\beta} = \beta$, then we get the following implementation:

$$r_0 = b - Ax_0, \quad z_0 = M^{-1}r_0, \quad p_0 = z_0 \tag{8.13a}$$

$$\alpha_k = \frac{r_k^T z_k}{p_k^T A p_k}, \tag{8.13b}$$

$$x_{k+1} = x_k + \alpha_k p_k, \tag{8.13c}$$

$$r_{k+1} = r_k - \alpha_k A p_k, \tag{8.13d}$$

$$z_{k+1} = M^{-1} r_{k+1} \tag{8.13e}$$

$$\beta_k = \frac{z_{k+1}^T r_{k+1}}{z_k r_k}, \tag{8.13f}$$

$$p_{k+1} = z_{k+1} + \beta_k p_k. \tag{8.13g}$$

In practice we can simply use any matrix $M^{-1} \approx A^{-1}$ in the implementation (8.13) without deriving the matrix P. But here both M and A must be symmetric positive definite. Otherwise the iteration above may not converge. Also, M must be a matrix that can be efficiently inverted ($Mz = r$ must be efficiently solved) for the algorithm to make any sense.

Example 8.2. *Consider solving a 1D variable coefficient problem as discussed in Section 2.11:*

$$-(a(x)u'(x))' = f(x), \quad x \in [0,1],$$

with homogeneous Dirichlet boundary conditions. A conservative discretization must be used:

$$-\frac{1}{\Delta x^2}[-a_{j-\frac{1}{2}}u_{j-1} + (a_{j-\frac{1}{2}} + a_{j+\frac{1}{2}})u_j - a_{j+\frac{1}{2}}u_{j+1}] = f_j,$$

where $a_{j-\frac{1}{2}} = a(x_j - \frac{1}{2}\Delta x)$. *The matrix vector form of this scheme is* $Au = f$ *where* A *is a real symmetric tridiagonal matrix:*

$$A = -\frac{1}{\Delta x^2}\begin{pmatrix} a_{\frac{1}{2}} + a_{\frac{3}{2}} & -a_{\frac{3}{2}} & & \\ -a_{\frac{3}{2}} & a_{\frac{3}{2}} + a_{\frac{5}{2}} & -a_{\frac{5}{2}} & \\ & \ddots & \ddots & \ddots \end{pmatrix}.$$

Notice that A *reduces to tridiagonal* $(-1, 2, -1)$ K *matrix if* $a(x) \equiv 1$. *Thus we can use the preconditioner* $M = K$ *in Preconditioned CG* (8.13) *even for solving a 2D variable coefficient problem* $-\nabla \cdot (a(x,y)\nabla u) = f$, *since* $K2D$ *matrix can be efficiently inverted by eigenvector method as we have discussed in Chapter 2.*

Example 8.3. *In practice, multigrid methods such as the V-cycle can also be used as a preconditioner in the Preconditioned Conjugate Gradient method*

(8.13). *In this case, the matrix* M^{-1} *in* (8.13e) *should be implemented as the V-cycle operator. Recall that V-cycle for solving* $A_h u = f_h$ *is given as* $u_h \leftarrow \textbf{V-cycle}(u_h, f_h)$. *We can implement the step* (8.13e) *as*

$$z_{k+1} = \textbf{V-cycle}(0, r_{k+1}),$$

i.e., applying V-cycle to r_{k+1} *with initial guess* 0.

Problem 8.1. *Consider solving the 1D (or 2D) Poisson equation* $-u'' = f$ *with PCG* (8.13) *with* M^{-1} *implemented as V-cycle. Recall that the matrix* M *(or equivalently* M^{-1}*) must be real symmetric and positive definite. With the weighted Jacobi relaxation/smoother and the restriction matrix defined as transpose of the interpolation operator, show that the matrix representation of a two-grid V-cycle operator using initial guess* 0 *is real symmetric and positive definite. Namely, show the matrix for the linear operator* $\textbf{V-cycle}(0, r)$ *is equivalent to applying a positive definite matrix to* r.

9

A brief introduction to nonlinear conservation laws

Preliminaries

- **Model problem**: Scalar conservation law

$$u_t + f_x(u) = 0. \tag{9.1}$$

  Given initial: $u(x, 0)$. Note, the subscript in (9.1) denotes derivative, for instance $u_t = \partial_t u$ and $f_x = \partial_x f$.

- **Weak solution**: Multiply test function $\phi \in C_0^1(\mathbb{R} \times \mathbb{R}^+)$ on (9.1) and integrate over space and time

$$\int_0^{+\infty} \int_{-\infty}^{+\infty} (u_t + f_x(u))\, \phi \, \mathrm{d}x\mathrm{d}t = 0. \tag{9.2}$$

  Here, C_0^1 is the space of function that are continuous differentiable with compact support. Integrate by part, yield the weak solution

$$\int_0^{+\infty} \int_{-\infty}^{+\infty} (\phi_t u + \phi_x f(u))\, \mathrm{d}x\mathrm{d}t = -\int_{-\infty}^{+\infty} \phi(x, 0) u(x, 0)\, \mathrm{d}x. \tag{9.3}$$

- **Numerical scheme**

$$U_j^{n+1} = U_j^n - \frac{k}{h} \left(F(U^n; j) - F(U^n; j-1) \right) \tag{9.4}$$

  Here, $F(U^n; j)$ is a flux function which is allow to depend on any finite number of elements of the vector U^n, "*centered*" about the j^{th} point.

$$F(U^n; j) = F(U_{j-p}^n, U_{j-p+1}^n, \cdots, U_{j+q}^n). \tag{9.5}$$

- **Consistency**: The numerical flux function F reduces to the true flux f for the case of constant flow, namely for all $\bar{u} \in \mathbb{R}$,

$$F(\bar{u}; j) = f(\bar{u}) \tag{9.6}$$

  Recall the concept of *Lipschitz continuous*: $|F(U^n; j) - f(\bar{u})| \leq K \max_{-p \leq i \leq q} |U_{j+i} - \bar{u}|$.

- **Discrete conservation**: The numerical flux on cell interface is single-valued. Therefore, we have

$$\sum_j U_j^{n+1} = \sum_j U_j^n \quad \text{for all } n. \tag{9.7}$$

- **Example**: Lax–Friedrichs method, see book page 125, equation (12.15) and (12.16).

Lax–Wendroff Theorem

Theorem 9.1 (Lax–Wendroff)**.** *Consider a sequence of grids indexed by* $\ell = 1, 2, \cdots$*, with mesh parameters* $k_\ell, h_\ell \to 0$ *as* $\ell \to \infty$*, let* $U_\ell(x,t)$ *denote the numerical approximation computed with a* ***consistent*** *and* ***conservative*** *method on the* ℓ^{th} *grid. Suppose* U_ℓ *converges to a function* u *as* $\ell \to \infty$*, in the sense that:*

1. *The* 1*-norm convergences:* $\|U_\ell - u\|_{1,\Omega} \to 0$ *as* $\ell \to \infty$*, where* $\Omega = [a,b] \times [0,T]$*.*

2. *Total variation bounded: there exists an* $R > 0$*, such that* $\mathrm{TV}(U_\ell(\cdot,t)) < R$*, for all* $0 \le t \le T$*,* $\ell = 1, 2, \cdots$*.*

Then, u *is a weak solution of the conservation law.*

Remark 9.1. *The Lax–Wendroff theorem does not guarantee that we do converge.*

Remark 9.2. *Even we have convergence, the Lax–Wendroff theorem does not guarantee that the weak solution obtained satisfy the entropy condition.*

Remark 9.3. *In case of a subsequence from a scheme convergences to physically correct solution (satisfies entropy condition), then the limit of this subsequence is a weak solution.*

Outline proof of Lax–Wendroff Theorem

- **Starting point**: The numerical scheme (9.4) for discussion.

- **Goal**: Show the limit function u satisfies (9.3).

- "**Roadmap**" and **motivation**:

  – Step 1. Analog the argument of deriving weak solution, multiply "test function" $\phi(x_j, t_n)$ on both side of the numerical scheme (9.4).

$$\text{integrate}: \int_0^{+\infty} \int_{-\infty}^{+\infty} \quad \to \quad \text{sum}: \sum_{n=0}^{+\infty} \sum_{j=-\infty}^{+\infty}$$

  – Step 2. Analog the argument of deriving weak solution, "integra-

tion by part" becomes "summation by part". Here use formulae

$$\sum_{n=0}^{m} a_n(b_{n+1}-b_n) = a_m b_{m+1} - a_0 b_0 + \sum_{n=0}^{m-1}(a_n - a_{n+1})b_{n+1}, \tag{9.8}$$

$$\sum_{j=-m}^{m} a_j(b_j - b_{j-1}) = a_m b_m - a_{-m} b_{-m-1} - \sum_{j=-m}^{m-1}(a_{j+1} - a_j)b_j. \tag{9.9}$$

Idea: original sum involves the product of a_j with differences of b's. Rewrite: final sum involves the product of b_j with differences of a's

- Step 3. Figuring out suitable conditions for our goal.
  - We will see how the conditions 1 and 2 are applied in the proof.
  - Review the concept of *the* 1*-norm convergences* and *total variation*, see book page 131.

- **Something to keep in mind**: the support of test function is compact, namely $\phi(x_j, t_n) = 0$ for $|j|$ or n sufficiently large.

More details of the proof for Lax–Wendroff Theorem

- Apply Step 1, we get:

$$\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\phi(x_j,t_n)(U_j^{n+1}-U_j^n)=-\frac{k}{h}\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\phi(x_j,t_n)\left(F(U^n;j)-F(U^n;j-1)\right). \tag{9.10}$$

- Apply Step 2 (note the support of ϕ is compact), we get:
  - Apply (9.8) to the left-hand side in (9.10) for "index n",

$$\begin{aligned}\text{LHS}&=\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\phi(x_j,t_n)(U_j^{n+1}-U_j^n)\\&=-\sum_{j=-\infty}^{+\infty}\phi(x_j,t_0)U_j^0-\sum_{j=-\infty}^{+\infty}\sum_{n=1}^{+\infty}\left(\phi(x_j,t_n)-\phi(x_j,t_{n-1})\right)U_j^n.\end{aligned} \tag{9.11}$$

  - Apply (9.9) to the right-hand side in (9.10) for "index j",

$$\begin{aligned}\text{RHS}&=-\frac{k}{h}\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\phi(x_j,t_n)\left(F(U^n;j)-F(U^n;j-1)\right)\\&=\frac{k}{h}\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\left(\phi(x_{j+1},t_n)-\phi(x_j,t_n)\right)F(U^n;j).\end{aligned} \tag{9.12}$$

  Therefore, substitute (9.11) and (9.12) into (9.10), we obtain

$$\begin{aligned}&-\sum_{j=-\infty}^{+\infty}\phi(x_j,t_0)U_j^0-\sum_{j=-\infty}^{+\infty}\sum_{n=1}^{+\infty}\left(\phi(x_j,t_n)-\phi(x_j,t_{n-1})\right)U_j^n\\&\qquad-\frac{k}{h}\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\left(\phi(x_{j+1},t_n)-\phi(x_j,t_n)\right)F(U^n;j)=0.\end{aligned} \tag{9.13}$$

  Multiply h on both side above and move the last two terms to the right side, we get

$$\underbrace{hk\sum_{n=1}^{+\infty}\sum_{j=-\infty}^{+\infty}\frac{\phi(x_j,t_n)-\phi(x_j,t_{n-1})}{k}U_j^n}_{=\mathrm{T}_1}$$
$$\underbrace{+hk\sum_{n=0}^{+\infty}\sum_{j=-\infty}^{+\infty}\frac{\phi(x_{j+1},t_n)-\phi(x_j,t_n)}{h}F(U^n;j)}_{=\mathrm{T}_2}=\underbrace{-\sum_{j=-\infty}^{+\infty}\phi(x_j,t_0)U_j^0}_{=\mathrm{T}_3}. \tag{9.14}$$

- Apply Step 3, take limit $\ell \to \infty$ ($k_\ell, h_\ell \to 0$)
  - The term T_1 and T_3 are handled by using condition 1.
  - The term T_2 is handled by using condition 2.

How to take limit?

- Review our **goal**: we want to obtain the following convergence, as $\ell \to \infty$ ($k_\ell, h_\ell \to 0$).

$$\mathrm{T}_1 \to \int_0^{+\infty}\int_{-\infty}^{+\infty} \phi_t u \,\mathrm{d}x\mathrm{d}t, \tag{9.15}$$

$$\mathrm{T}_2 \to \int_0^{+\infty}\int_{-\infty}^{+\infty} \phi_x f(u) \,\mathrm{d}x\mathrm{d}t, \tag{9.16}$$

$$\mathrm{T}_3 \to \int_{-\infty}^{+\infty} \phi(x,0)u(x,0)\mathrm{d}x. \tag{9.17}$$

- Notice, ϕ has compact support. For each ℓ, only finitely many terms in the sum of terms T_1, T_2, T_3 are non-zero. thus the sums are well-defined.

- Employ the notation $U_\ell(x_j, t_n)$ for piecewise constant function defined by U_j^n for a grid ℓ on $[x_{j-1/2}, x_{j+1/2}) \times [t_n, t_{n+1})$. The (9.18) is a Riemann sum of step functions, which can be written as

$$\begin{aligned}
&\underbrace{\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x,t)-\phi_\ell(x,t-k)}{k} U_\ell(x,t)\,\mathrm{d}x\mathrm{d}t}_{=\mathrm{T}_1} \\
&+\underbrace{\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x+h,t)-\phi_\ell(x,t)}{h} F(U_\ell(x-ph,t),\cdots,U_\ell(x+qh,t))\,\mathrm{d}x\mathrm{d}t}_{=\mathrm{T}_2} \\
&\qquad = -\underbrace{\int_{-\infty}^{+\infty} \phi_\ell(x,0)U_\ell(x,0)\,\mathrm{d}x}_{=\mathrm{T}_3}.
\end{aligned} \tag{9.18}$$

- For the term T_1, in order to obtain (9.15), we employ condition "*the 1-norm convergences*".

  - Recall our goal is to show (9.15). Insert term "$u - u$" after U_ℓ, we get

$$\begin{aligned}
\mathrm{T}_1 = &\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x,t)-\phi_\ell(x,t-k)}{k} u(x,t)\,\mathrm{d}x\mathrm{d}t \\
&+\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x,t)-\phi_\ell(x,t-k)}{k} \Big(U_\ell(x,t)-u(x,t)\Big)\,\mathrm{d}x\mathrm{d}t.
\end{aligned} \tag{9.19}$$

  - By mean value theorem, $\exists \xi_{\ell,t} \in [t-k, t]$, such that

$$\phi_\ell(x,t) - \phi_\ell(x,t-k) = k\phi_t(x,\xi_{\ell,t}) \tag{9.20}$$

    Recall $\phi \in C_0^1(\mathbb{R}\times\mathbb{R}^+)$ with compact support and the definition of ϕ_ℓ.

– Use the condition "*the 1-norm convergences*", consider the support of ϕ_t is compact and $\phi_t \le \|\phi_t\|_{L^\infty}$. Take limit of the following expression

$$\begin{aligned} \mathrm{T}_1 = &\int_0^{+\infty}\int_{-\infty}^{+\infty} \phi_t(x,\xi_{\ell,t})\, u(x,t)\,\mathrm{d}x\mathrm{d}t \\ &+ \underbrace{\int_0^{+\infty}\int_{-\infty}^{+\infty} \phi_t(x,\xi_{\ell,t})\,\Big(U_\ell(x,t)-u(x,t)\Big)\,\mathrm{d}x\mathrm{d}t}_{\le\ \|\phi_t\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\,\|U_\ell-u\|_{1,\Omega}}. \end{aligned} \tag{9.21}$$

- For the term T_2, insert term "$f(U_\ell(x,t)) - f(U_\ell(x,t))$" after F, we have

$$\begin{aligned} \mathrm{T}_2 = &\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x+h,t)-\phi_\ell(x,t)}{h}\, f(U_\ell(x,t))\,\mathrm{d}x\mathrm{d}t \\ +&\int_0^{+\infty}\int_{-\infty}^{+\infty} \frac{\phi_\ell(x+h,t)-\phi_\ell(x,t)}{h}\,\Big(F(U_\ell(x-ph,t),\cdots,U_\ell(x+qh,t))-f(U_\ell(x,t))\Big)\,\mathrm{d}x\mathrm{d}t \\ &= \mathrm{S}_1+\mathrm{S}_2 \end{aligned} \tag{9.22}$$

Let us show $\mathrm{S}_2 \to 0$ as $\ell \to \infty$.

– By mean value theorem, $\exists \xi_{\ell,x} \in [x, x+h]$, such that

$$\phi_\ell(x+h,t)-\phi_\ell(x,t) = h\phi_x(\xi_{\ell,x},t). \tag{9.23}$$

Recall $\phi \in C_0^1(\mathbb{R}\times\mathbb{R}^+)$ with compact support and the definition of ϕ_ℓ.

– Rewrite the term S_2 into the summation with respect to n and j. Note ϕ has compact support, we can assume $\phi = 0$ for all $t > T$.

$$\begin{aligned} |S_2| &\le \|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\int_0^T\int_{-\infty}^{+\infty} |F(U_\ell(x-ph,t),\cdots,U_\ell(x+qh,t))-f(U_\ell(x,t))|\,\mathrm{d}x\mathrm{d}t \\ &= \|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\, hk\sum_{n=0}^{T/k}\sum_{j=-\infty}^{+\infty} |F(U_\ell(x_{j-p},t_n),\cdots,U_\ell(x_{j+q},t_n))-f(U_\ell(x_j,t))| \end{aligned} \tag{9.24}$$

– Flux F is Lipschitz continuous, see LeVeque's book page 126 equation (12.19).

$$\begin{aligned} &|F(U_\ell(x_{j-p},t),\cdots,U_\ell(x_{j+q},t))-f(U_\ell(x_j,t))| \\ \le &K\max_{-p\le i\le q}|U_\ell(x_{j+i},t)-U_\ell(x_j,t)|. \end{aligned} \tag{9.25}$$

Substitute (9.25) into (9.24), notice the width of stencil is finite $p+q+1$, the term S_2 can be bounded by (trick: telescoping

summation)

$$|S_2| \le \|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\ hk \sum_{n=0}^{T/k} \sum_{j=-\infty}^{+\infty} K \max_{-p\le i\le q} |U_\ell(x_{j+i},t_n) - U_\ell(x_j,t_n)|$$
$$\le K\|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\ h \sum_{n=0}^{T/k} k\Big((p+q+1) \sum_{j=-\infty}^{+\infty} |U_\ell(x_j,t_n) - U_\ell(x_{j-1},t_n)|\Big). \tag{9.26}$$

– Recall U_ℓ is bounded total variation, see LeVeque's book page 131 equation (12.39 and 12.40). The term S_2 can be bounded by

$$|S_2| \le K(p+q+1)\|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\ h\Big(k \sum_{n=0}^{T/k} \mathrm{TV}(U_\ell(\cdot,t_n))\Big)$$
$$\le KRT(p+q+1)\|\phi_x\|_{L^\infty(\mathbb{R}^+\times\mathbb{R})}\ h \tag{9.27}$$

Therefore, $S_2 \to 0$ as $h \to 0$ ($\ell \to \infty$).

– *Example*: Lax–Friedrichs flux (stencil width 2).

- The rest terms S_1 and T_3 can be processed similarly.

10

Boundary conditions for hyperbolic systems

10.1 Statement of the problem

So far we have studied hyperbolic partial differential equations assuming periodicity conditions. In this chapter, we analyze the treatment of general boundary conditions. We shall state the problem under consideration via the model example:

$$u_t = au_x, \quad u(x,0) = f(x), \quad x \in (-1,1). \tag{10.1}$$

In general, initial values of $f(x)$ defined on $x \in (-1,1)$, are not enough to solve for $u(x,t)$. In addition to $f(x)$, we need to specify boundary condition. The first problem is to determine what type of boundary conditions should we specify, the second is to study well posedness of the problem with boundary conditions. In this example, an immediate answer may be given to the first question using *characteristics*. The equation for the characteristics of (10.1) is:

$$dt = -\frac{1}{a}dx,$$

and therefore $u(x,t)$ is constant when $x + at$ is a constant. This follows immediately, since calling $\phi(t) = u(k - at, t)$ for k a constant, we have:

$$\frac{d\phi(t)}{dt} = u_x(k - at, t)(-a) + u_t(k - at, t) = 0.$$

Therefore, the solution is a constant along the characteristics, which for this problem, are straight lines. The information given by $f(x)$ for $-1 < x < 1$ is therefore insufficient to solve the problem, since we do not have values at points $x > 1$. We therefore need to specify for a > 0, right boundary conditions in the form:

$$u(1,t) = g_+(t), \quad \text{if} \quad a > 0.$$

Analogously, if $a < 0$, we need to specify left boundary conditions:

$$u(-1,t) = g_-(t), \quad \text{if} \quad a < 0.$$

Suppose that $a > 0$, so that we specify right boundary conditions. Can we also specify left boundary conditions safely? The answer is no. The values of $u(-1,t)$ must coincide with $u(x,t)$ along the corresponding characteristic in order for the solution to be continuous. This is in general true and therefore we must determine first which boundary conditions are needed and avoid overspecification. We now turn to the second problem for which we shall use energy estimates. We define the energy of the system in (10.1) by:

$$E(t) = \int_{-1}^{1} u^2(x,t)\,dx.$$

Under periodic boundary conditions, we saw that there is conservation of energy, but for this problem this is no longer true, since there is an external influence through the boundaries. Differentiating the energy we get:

$$\begin{aligned}\frac{d}{dt}E(t) = \frac{d}{dt}\int_{-1}^{1} u^2(x,t)\,dx &= 2a\int_{-1}^{1} u(x,t)u_x(x,t)\,dx \\ &= a\int_{-1}^{1}\frac{d}{dx}u^2(x,t)\,dx \\ &= a[u^2(1,t) - u^2(-1,t)].\end{aligned}$$

We can now see how the boundary condition affects the energy of the system. Define:

$$g(t) = \begin{cases} g_+(t), & \text{if} \quad a > 0 \\ g_-(t), & \text{if} \quad a < 0 \end{cases}$$

then, whether a is positive or negative, we have

$$\frac{d}{dt}E(t) \le |a| g^2(t).$$

If $g(t) = 0$, then there is dissipation of energy. Dissipative mechanisms for these equations come from the boundaries. Physically, there is an interaction between the system and the exterior: energy goes out from the system at the left boundary (for $a > 0$), and it is "pumped" into the system through the right boundary, depending on the right boundary condition given. If there is no energy pumped into the system ($g(t) = 0$), then the energy flows out from the system and eventually reaches zero. In the periodic case, we require $u(1,t) = u(-1,t)$ and conservation of energy is a consequence of the fact that energy flows into the system at the same rate that it goes out from it through the boundaries. Upon integration of the last inequality, we get:

$$E(t) = \int_{-1}^{1} u^2(x,t)\,dx \le \int_{-1}^{1} f(x,t)\,dx + |a|\int_0^t g^2(s)\,ds,$$

that is, we control the norm of the solution at any given time $t > 0$ by the bounded function $\int_0^t g^2(s)\,ds$, which yields well posedness of the solution. We conclude that for these problems:

- There is a definite direction of inflow-outflow related to the characteristics,
- We must specify the incoming flow through the corresponding boundary, but not the outgoing flow,
- The energy of the system may decay.

It is in general true that for linear hyperbolic equations we control the system's energy through the boundary conditions and therefore when modelling real life problems, special care must be taken on the specification of boundary conditions. Let us now turn to a more interesting example, where inflow of energy may come from both boundaries.

Example 10.1. *Consider the wave equation:*

$$u_{tt} = u_{xx}, \quad |x| < 1.$$

As we saw in Chapter 5, we can decouple this system in order to get two scalar equations of the form (10.1)*. Define:*

$$v = u_t, \quad = u_x,$$

then we have:

$$\begin{pmatrix} v \\ w \end{pmatrix}_t = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix}_x$$

which, in turn, is equivalent to:

$$u_t^I = -u_x^I$$
$$u_t^{II} = u_x^{II}$$

where $u^I = (v-w)/2$ and $u^{II} = (v+w)/2$. From the discussion on (10.1)*, it is clear now that for this problem, we must specify both boundary conditions. There are two classes of characteristics: one with slope* 1 *related to u^I, and the other with slope -1, related to u^{II}. We must, however, take into account that u^I and u^{II} might still be coupled, precisely through the boundary conditions. In order to see this coupling mechanism, suppose that the boundary conditions are given for the original variable $u(x,t)$ by:*

$$u(1,t) = g_+(t), \quad u(-1,t) = g_-(t).$$

This gives the boundary conditions for u_t, although not for u_x. In terms of the variables u^I and u^{II}, we have:

$$\begin{aligned} u^I(-1,t) &= \frac{u_t(-1,t) - u_x(-1,t)}{2} \\ &= -\frac{u_t(-1,t) + u_x(-1,t)}{2} + u_t(-1,t) \\ &= -u^{II}(-1,t) + \frac{d}{dt}g_-(t). \\ u^{II}(1,t) &= -u^I(1,t) + g'_+(t). \end{aligned}$$

If the initial conditions are

$$u(x,0) = F(x), \quad u_t(x,0) = f_2(x),$$

then we have:

$$u_x(x,0) = f_1(x), \quad u_t(x,0) = f_2(x),$$

and therefore we have initial conditions for u^I and u^{II}:

$$u^I(x,0) = \frac{f_2(x) - f_1(x)}{2},$$

$$u^{II}(x,0) = \frac{f_2(x) + f_1(x)}{2}.$$

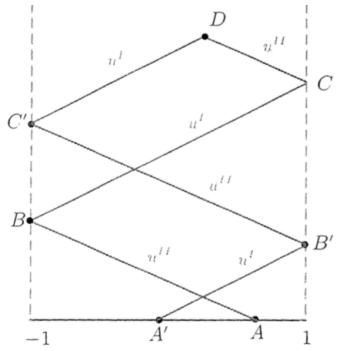


Figure 10.1: An illustration of characteristics and boundary conditions.

In Figure (10.1)*, the solution evaluated at some point D is constructed in the following way: u^{II} has the same value at B as the initial value A. At the point B, $u^I = u^{II} + g'_-(t)$, and u^I maintains this value up to the point C in the right boundary. At that point C, $u^{II} = u^I + g'_+(t)$, and this is the value of u^{II} at the point D. Analogously, following now the characteristics, we obtain the value of u^I at the point D. As seen in this example, the boundary conditions $g_-(t)$ and $g_+(t)$ give the amount of "reflection" at the boundaries, relating u^I and u^{II} at those points. Therefore, specification of the initial and boundary conditions provide enough data to solve the problem.*

We cannot, however, specify any kind of boundary conditions at will, since we may lack enough information to solve the problem, as the following example shows.

Example 10.2. *Consider the equation* $u_{tt} = u_{xx}$, $x \in [-1, 1]$ *under the boundary conditions:*

$$u_t(-1, t) + u_x(-1, t) = 0,$$

$$u_t(1, t) + u_x(1, t) = 0.$$

As before, define u^I *and* u^{II} *and we get a decoupled system for* u^I *and* u^{II}. *But now we have the boundary conditions:*

$$u^{II}(-1, t) = 0, \quad u^{II}(1, t) = 0,$$

and there are no boundary conditions for u^I. *Following the solid line from point A to point B, we see that there might be a contradiction between the initial value of* u^{II} *at A, and the boundary value at B, set to zero. This is an over specification and we should not use boundary condition for* u^{II} *at the left boundary. On the other hand, we are lacking information about* u^I.

Let us summarize the results illustrated in the examples, extracting the ideas to be generalized later on. Upon diagonalizing the system, we expressed the problem through the equivalent equation

$$\begin{pmatrix} u^I \\ u^{II} \end{pmatrix}_t = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u^I \\ u^{II} \end{pmatrix}_x$$

The fact that there is one negative eigenvalue and one positive eigenvalue allows us to decouple the original equation, where now there is one characteristic slope for each of the new variables u^I and u^{II}. In general, it should also be true that the number of right (left) boundary conditions coincides with the number of positive (negative) eigenvalues. Given these conditions as $g_-(t)$ and $g_+(t)$, we get:

$$u^I(-1, t) = -u^{II}(-1, t) + g'_-(t),$$

$$u^{II}(1, t) = -u^I(1, t) + g'_+(t).$$

10.2 Boundary conditions for 1D hyperbolic systems

Consider the one dinnensional hyperbolic system:

$$w_t = Aw_x$$

where A is a $p \times p$ matrix, and $-1 < x < 1$. We will assume that the system is strongly hyperbolic (see Chapter 5) thus A can be diagonalized. Furthermore, we will assume that A has nonzero eigenvalues. We order the eigenvalues of A, denoted by $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_p$, so that if A has r negative eigenvalues, then these will correspond to $\lambda_1, \cdots, \lambda_r$, and $0 < \lambda_{r+1} \leq \cdots \leq \lambda_p$. Since A is diagonalizable, there exists a matrix T such that:

$$T^{-1}AT = \begin{pmatrix} -\Lambda^I & 0 \\ 0 & \Lambda^{II} \end{pmatrix}$$

where Λ^I and Λ^{II} are diagonal matrices with positive entries:

$$\Lambda^I = \begin{pmatrix} -\lambda_1 & & \\ & \ddots & \\ & & -\lambda_r \end{pmatrix}, \quad \Lambda^{II} = \begin{pmatrix} \lambda_{r+1} & & \\ & \ddots & \\ & & \lambda_p \end{pmatrix}.$$

Under the transformation induced by T, we obtain the equivalent system:

$$\begin{pmatrix} u^I \\ u^{II} \end{pmatrix}_t = \begin{pmatrix} -\Lambda^I & 0 \\ 0 & \Lambda^{II} \end{pmatrix} \begin{pmatrix} u^I \\ u^{II} \end{pmatrix}_x,$$

where u^I contains the first r components of Tw and u^{II} the last $(p-r)$ components.

Let L be an $r \times (p-r)$ matrix, and R be a $(p-r) \times r$ matrix and consider the boundary conditionsin the form:

$$u^I(-1,t) = Lu^{II}(-1,t), \tag{10.2}$$

$$u^{II}(1,t) = Ru^I(1,t). \tag{10.3}$$

The notation is clear: here, L and R stand for "left" and "right" boundary conditions, respectively. The aim of this section is to prove that the problem with boundary conditions above is well posed. We shall do this in two steps, first considering the case $\|L\|\|R\| \leq 1$, and then the case $\|L\|\|R\| > 1$.

Theorem 10.1. *For the boundary conditions* (10.2) *and* (10.3)*, if* $\|L\|\|R\| \leq 1$*, then the solutions* u^I *and* u^{II} *do not grow in time.*

Proof. Rescale the matrices by

$$S^I = \|R\|(\Lambda^I)^{-1}, \quad S^{II} = \|L\|(\Lambda^{II})^{-1}.$$

Note that this scaling is used to obtain a speed of propogation of one. We want to show the energy

$$E(t) = \int_{-1}^{1} \|u^I\|^2 + \|u^{II}\|^2 \, dx$$

decays w.r.t. time. Instead, we will show an equivalent norm decays. Define

$$\tilde{E}(t) = \int_{-1}^{1} \left(\langle u^I, S^I u^I \rangle + \langle u^{II}, S^{II} u^{II} \rangle \right) dx.$$

Notice that S^I and S^{II} are diagonal matrices. So we get

$$\langle u^I, S^I u^I \rangle = \sum_{i=1}^{r} u_i^I S_{ii}^I u_i^I = \|R\| \sum_{i=1}^{r} \frac{|u_i^I|^2}{|\lambda_i|},$$

$$\langle u^{II}, S^{II} u^{II} \rangle = \sum_{i=r}^{p-r} u_i^{II} S_{ii}^{II} u_i^{II} = \|L\| \sum_{i=r}^{p-r} \frac{|u_i^{II}|^2}{|\lambda_i|},$$

thus

$$\frac{\min\{\|L\|, \|R\|\}}{\max_i \|\lambda_i\|} E(t) \leq \tilde{E}(t) \leq \frac{\max\{\|L\|, \|R\|\}}{\min_i \|\lambda_i\|} E(t).$$

Evaluating the derivatives, we get

$$\begin{aligned}
\frac{d}{dt}\tilde{E}(t) &= \int_{-1}^{1} \frac{d}{dt} \left(\langle u^I, S^I u^I \rangle + \langle u^{II}, S^{II} u^{II} \rangle \right) dx \\
&= \int_{-1}^{1} \left(\langle u_t^I, S^I u^I \rangle + \langle u^I, S^I u_t^I \rangle + \langle u_t^{II}, S^{II} u^{II} \rangle + \langle u^{II}, S^{II} u_t^{II} \rangle \right) dx \\
&= \int_{-1}^{1} 2 \left(\langle u^I, S^I u_t^I \rangle + \langle u^{II}, S^{II} u_t^{II} \rangle \right) dx \\
&= \int_{-1}^{1} 2 \left(\langle u^I, -S^I \Lambda^I u_x^I \rangle + \langle u^{II}, S^{II} \Lambda^{II} u_x^{II} \rangle \right) dx \\
&= \int_{-1}^{1} \frac{d}{dx} \left(\langle u^I, -S^I \Lambda^I u^I \rangle + \langle u^{II}, S^{II} \Lambda^{II} u^{II} \rangle \right) dx \\
&= \int_{-1}^{1} \frac{d}{dx} \left(-\|R\| \langle u^I, u^I \rangle + \|L\| \langle u^{II}, u^{II} \rangle \right) dx \\
&= \left(-\|R\| \langle u^I, u^I \rangle + \|L\| \langle u^{II}, u^{II} \rangle \right) \Big|_{x=-1}^{x=1}.
\end{aligned}$$

Plugging in the boundary conditions (10.2) and (10.3), we get

$$\begin{aligned}
\frac{d}{dt}\tilde{E}(t) &= \left(-\|R\| \langle u^I, u^I \rangle + \|L\| \langle u^{II}, u^{II} \rangle \right) \Big|_{x=1} \\
&\quad - \left(-\|R\| \langle u^I, u^I \rangle + \|L\| \langle u^{II}, u^{II} \rangle \right) \Big|_{x=-1} \\
&= \left(-\|R\| \langle u^I, u^I \rangle + \|L\| \langle Ru^I, Ru^I \rangle \right) \Big|_{x=1} \\
&\quad - \left(-\|R\| \langle Lu^{II}, Lu^{II} \rangle + \|L\| \langle u^{II}, u^{II} \rangle \right) \Big|_{x=-1} \\
&= (-\|R\| + \|L\|\|R\|^2) \|u^I(1,t)\|^2 - (-\|R\|\|L\|^2 + \|L\|) \|u^{II}(-1,t)\|^2 \\
&= \|R\| (\|L\|\|R\| - 1) \|u^I(1,t)\|^2 + \|L\| (\|L\|\|R\| - 1) \|u^{II}(-1,t)\|^2.
\end{aligned}$$

Thus $\|L\|\|R\| \leq 1$ implies $\frac{d}{dt}\tilde{E}(t) \leq 0$. □

The above result relates the boundary conditions given through L and R, with the amount of the growth on the solution that comes as a reflection at the boundaries. This statement is illustrated geometrically in the following example.

Example 10.3. *Consider the case where $p = 2$, and A has one negative and one positive eigenvalue. Then the system is decoupled as:*

$$u_t^I = -a_1 u_x^I,$$

$$u_t^{II} = a_2 u_x^{II},$$

where $a_1 = -\lambda_1 > 0$, $a_2 = \lambda_2 > 0$. The boundary conditions (10.2) *and* (10.3) *are now:*

$$u^I(-1,t) = Lu^{II}(-1,t),$$

$$u^{II}(1,t) = Ru^I(1,t),$$

where now L and R are real numbers. In Figure (10.1)*, if the value of u^{II} at A is α, then at B u^I has value $L\alpha$, and u^{II} has value $LR\alpha$ at C, thus u^{II} has value $LR\alpha$ at D. Therefore, if $|LR| \leq 1$, the magnitude of u^{II} does not increase in time, whereas if $|LR| > 1$, the function u^{II} will grow as time goes on. An analogous argument follows for u^I, which yields the relation between the boundary conditions R and L, and the amount of the reflection.*

It turns out that this is a general result. When R and L are matrices, if $\|L\|\|R\| > 1$, then $\frac{d}{dt}\tilde{E}(t)$ is no longer bounded by 0, and solutions may grow in time. Nevertheless, this growth is bounded by a function of the form $e^{\alpha t}\|u(x,0)\|$, as the following result shows.

Theorem 10.2. *For the boundary conditions* (10.2) *and* (10.3)*, if $\|L\|\|R\| > 1$, then there exist an energy function $E(t)$ such that*

$$E(t) \leq Ke^{\alpha t}.$$

Proof. Rescale different scaling matrices by

$$S^I = \frac{1}{\|L\|}(\Lambda^I)^{-1}, \quad S^{II} = \frac{1}{\|R\|}(\Lambda^{II})^{-1}.$$

Define the energy as

$$E(t) = \int_{-1}^{1} \Big((1+\varepsilon x)\langle u^I, S^I u^I\rangle + (1-\varepsilon x)\langle u^{II}, S^{II} u^{II}\rangle \Big)\, dx,$$

where $\varepsilon \in [0,1]$ is to be determined later. It is easy to show this energy is equivalent to the "standard" energy. We have

$$\begin{aligned}
\frac{d}{dt}E(t) &= \int_{-1}^{1} \frac{d}{dt}\left((1+\varepsilon x)\langle u^I, S^I u^I\rangle + (1-\varepsilon x)\langle u^{II}, S^{II}u^{II}\rangle\right) dx \\
&= \int_{-1}^{1} 2\left((1+\varepsilon x)\langle u^I, S^I u_t^I\rangle + (1-\varepsilon x)\langle u^{II}, S^{II}u_t^{II}\rangle\right) dx \\
&= \int_{-1}^{1} 2\left((1+\varepsilon x)\langle u^I, -S^I\Lambda^I u_x^I\rangle + (1-\varepsilon x)\langle u^{II}, S^{II}\Lambda^{II}u_x^{II}\rangle\right) dx \\
&= \int_{-1}^{1} \left((1+\varepsilon x)\frac{d}{dx}\langle u^I, -S^I\Lambda^I u^I\rangle + (1-\varepsilon x)\frac{d}{dx}\langle u^{II}, S^{II}\Lambda^{II}u^{II}\rangle\right) dx \\
&= \int_{-1}^{1} \left(-\frac{1}{\|L\|}(1+\varepsilon x)\frac{d}{dx}\langle u^I, u^I\rangle + \frac{1}{\|R\|}(1-\varepsilon x)\frac{d}{dx}\langle u^{II}, u^{II}\rangle\right) dx.
\end{aligned}$$

Integration by parts gives us

$$\begin{aligned}
\frac{d}{dt}E(t) =& -\frac{1+\varepsilon}{\|L\|}\|u^I(1,t)\|^2 + \frac{1-\varepsilon}{\|L\|}\|u^I(-1,t)\|^2 + \frac{\varepsilon}{\|L\|}\int_{-1}^{1}\|u^I(x,t)\|^2\,dx \\
&+ \frac{1-\varepsilon}{\|R\|}\|u^{II}(1,t)\|^2 - \frac{1+\varepsilon}{\|R\|}\|u^{II}(-1,t)\|^2 + \frac{\varepsilon}{\|R\|}\int_{-1}^{1}\|u^{II}(x,t)\|^2\,dx
\end{aligned}$$

The boundary conditions (10.2) implies that

$$\begin{aligned}
&\frac{1-\varepsilon}{\|L\|}\|u^I(-1,t)\|^2 - \frac{1+\varepsilon}{\|R\|}\|u^{II}(-1,t)\|^2 \\
=&\frac{1-\varepsilon}{\|L\|}\|Lu^{II}(-1,t)\|^2 - \frac{1+\varepsilon}{\|R\|}\|u^{II}(-1,t)\|^2 \\
\le&(1-\varepsilon)\|L\|\|u^{II}(-1,t)\|^2 - \frac{1+\varepsilon}{\|R\|}\|u^{II}(-1,t)\|^2 \\
=&\|u^{II}(-1,t)\|^2\left((1-\varepsilon)\|L\| - \frac{1+\varepsilon}{\|R\|}\right) \\
=&\|u^{II}(-1,t)\|^2\left(\|L\| - \frac{1}{\|R\|} - \varepsilon(\|L\| + \frac{1}{\|R\|})\right),
\end{aligned}$$

which is zero if we choose $\varepsilon = \frac{\|L\|\|R\|-1}{\|L\|\|R\|+1}$.

Similarly, the boundary conditions (10.3) implies that

$$\begin{aligned}
&-\frac{1+\varepsilon}{\|L\|}\|u^I(1,t)\|^2+\frac{1-\varepsilon}{\|R\|}\|u^{II}(1,t)\|^2\\
=&-\frac{1+\varepsilon}{\|L\|}\|u^I(1,t)\|^2+\frac{1-\varepsilon}{\|R\|}\|Ru^I(1,t)\|^2\\
\le&-\frac{1+\varepsilon}{\|L\|}\|u^I(1,t)\|^2+(1-\varepsilon)\|R\|\|u^I(1,t)\|^2\\
=&\|u^I(1,t)\|^2\left(-\frac{1+\varepsilon}{\|L\|}+(1-\varepsilon)\|R\|\right)\\
=&\|u^I(1,t)\|^2\left(-\frac{1}{\|L\|}+\|R\|-\varepsilon(\frac{1}{\|L\|}+\|R\|),\right)
\end{aligned}$$

which is also zero if we choose $\varepsilon=\frac{\|L\|\|R\|-1}{\|L\|\|R\|+1}$.

For the two integrals left, notice that for $|x|\le 1$, we have $1+\varepsilon x\ge 1-\varepsilon$ thus $1+\varepsilon(1+x)\ge 1$. Then

$$\begin{aligned}
&\frac{\varepsilon}{\|L\|}\int_{-1}^{1}\|u^I(x,t)\|^2\,dx\\
\le&\frac{\varepsilon}{\|L\|}\int_{-1}^{1}\frac{1+\varepsilon x}{1-\varepsilon}\|u^I(x,t)\|^2\,dx\\
=&\frac{1}{\|L\|}\frac{\varepsilon}{1-\varepsilon}\int_{-1}^{1}(1+\varepsilon x)\|u^I(x,t)\|^2\,dx\\
=&\frac{1}{\|L\|}\frac{\varepsilon}{1-\varepsilon}\int_{-1}^{1}(1+\varepsilon x)\langle u^I,S^I\Lambda^I u^I\rangle\|L\|\,dx\\
=&\frac{\varepsilon}{1-\varepsilon}\int_{-1}^{1}(1+\varepsilon x)\langle u^I,S^I\Lambda^I u^I\rangle\,dx\\
\le&\frac{\varepsilon}{1-\varepsilon}\rho(\Lambda^I)\int_{-1}^{1}(1+\varepsilon x)\langle u^I,S^I u^I\rangle\,dx\\
\le&\frac{\varepsilon}{1-\varepsilon}\rho(A)\int_{-1}^{1}(1+\varepsilon x)\langle u^I,S^I u^I\rangle\,dx.
\end{aligned}$$

Similarly, we have

$$\frac{\varepsilon}{\|R\|}\int_{-1}^{1}\|u^{II}(x,t)\|^2\,dx\le\frac{\varepsilon}{1-\varepsilon}\rho(A)\int_{-1}^{1}(1-\varepsilon x)\langle u^{II},S^{II}u^{II}\rangle\,dx.$$

Therefore, by setting $\varepsilon=\frac{\|L\|\|R\|-1}{\|L\|\|R\|+1}$, we get

$$\frac{d}{dt}E(t)\le\frac{\varepsilon}{1-\varepsilon}\rho(A)E(t)=\frac{1}{2}(\|L\|\|R\|-1)\rho(A)E(t).$$

Combining this with Gronwall's inequality,[1] we get that

$$E(t) \leq E(0)e^{\alpha t},$$

where $\alpha = \frac{1}{2}(\|L\|\|R\| - 1)\rho(A) > 0$. □

We have thus established that $w_t = Aw_x$ with boundary conditions (10.2) and (10.3) yields well posedness. It is in general true that the problem is well posed if and only if the boundary conditions are of the form:

$$u^I(-1,t) = Lu^{II}(-1,t) + g_-(t),$$

$$u^{II}(1,t) = Ru^I(1,t) + g_+(t).$$

We have studied the case $g_-(t) = g_+(t) \equiv 0$, which illustrates the appropriate treatment of boundary conditions through energy estimates in a somewhat simpler notation. The arguments can be generalized for nonzero $g_-(t)$ and $g_+(t)$, but we omit the details here.

10.3 Kreiss theory, the multidimensional case

So far we have studied the problem of well posedness of hyperbolic system in one dimension. For one dimensional systems, the matrix A can be diagonalized under strong hyperbolicity, and the treatment of boundary conditions strongly relies on the "splitting" of the diagonalized matrix into Λ^I and Λ^{II}.

Consider now the multidimensional system:

$$u_t = \sum_{j=1}^{d} A_j \frac{\partial u}{\partial x_j}, \tag{10.4}$$

where $x = (x_1, \cdots, x_d)^T$ and $u = (u_1, \cdots, u_p)$. We will assume that the system is symmetric hyperbolic so that we can diagonalize one of the d matrices A_j and symmetrize the others under the same transformation. Since, in general, we cannot diagonalize simultaneously all the d matrices appearing in (10.4), Kreiss theory is based on looking at one boundary at a time. That is, we consider the domain to be of the form

$$-\infty < x_j < +\infty, \quad j = 2, \cdots, d$$

[1] Here is a proof of Gronwall's inequality. Suppose $\phi'(t) \leq \alpha\phi(t)$, then

$$\frac{d}{dt}(\phi(t)e^{-\alpha t}) = -\alpha\phi(t)e^{-\alpha t} + \phi'(t)e^{-\alpha t} \leq 0.$$

Thus $\phi(t)e^{-\alpha t} \leq \phi(0)$ which gives us

$$\phi(t) \leq \phi(0)e^{\alpha t}.$$

$$0 \leq x_1 < +\infty$$

and we analyze the appropriate left boundary condition for x_1. Let T be the transformation that diagonalizes A_1 and symmetrizes the other matrices $A_2, \cdots, A_d$ and denote:

$$A = T^{-1} A_1 T,$$

$$B_j = T^{-1} A_j T, \quad j = 2, \cdots, d.$$

We will use the same notation $u(x,t)$ to denote the function under the transformation T, so that the original problem is equivalent to:

$$u_t = A \frac{\partial u}{\partial x_1} + \sum_{j=2}^{d} B_j \frac{\partial u}{\partial x_j}, \tag{10.5}$$

where A is a diagonal matrix and $B_j, 2 < j < d$ are symmetric matrices. We assume that the eigenvalues of A are ordered such that:

$$\lambda_1 \leq \cdots \leq \lambda_r < 0 < \lambda_{r+1} \leq \cdots \leq \lambda_p,$$

and we shall split A accordingly, by:

$$A = \begin{pmatrix} -\Lambda^I & 0 \\ 0 & \Lambda^{II} \end{pmatrix}.$$

We also split u by

$$u = \begin{pmatrix} u^I \\ u^{II} \end{pmatrix}$$

where u^I denotes the first r components of u and u^{II} denote the last $(p-r)$ components. The number r stands for the number of negative eigenvalues of the matrix A. At the boundary $x = 0$, we must specify r left boundary conditions for the component $u^I(0, x_2, \cdots, x_d, t)$. We shall consider the boundary conditions to be of the form:

$$u^I(0, x_2, \cdots, x_d, t) = L u^{II}(0, x_2, \cdots, x_d, t), \tag{10.6}$$

where L is an $r \times (p-r)$ matrix.

Unlike the one dimensional case, having boundary conditions of the form (10.6) does not ensure well posedness of the problem. In this section, we study necessary conditions under which the problem is not well posed, and later state necessary conditions for well posedness. Energy estimates involve the definition of an appropriate energy for the system, which in the multi-dimensional case may lead to cumbersome analysis. The main contribution of Kreiss theory on boundary conditions lies on the construction of an appropriate framework which turns the problem into an algebraic problem, so

that the aforementioned conditions may be easily verified. Our first task will be to look for necessary conditions for the problem

$$u_t = A\frac{\partial u}{\partial x_1} + \sum_{j=2}^{d} B_j \frac{\partial u}{\partial x_j},$$

$$u^I(0, x_2, \cdots, x_d, t) = Lu^{II}(0, x_2, \cdots, x_d, t),$$

not to be well posed. Roughly speaking, we will look for "bad" boundary conditions for L, but we will not require these conditions to be also sufficient.

As mentioned before, we shall construct an appropriate framework that will simplify the characterization of such bad situations, however, the notation may seen complicated. We shall therefore present the main arguments in a step-by-step way. The next two results set up the fundamental ideas that will help us simplify the problem.

Lemma 10.1. *Let s be any complex number and $\omega_2, \cdots, \omega_d$ to be any real numbers. Let $\hat{u}(x_1)$ be a solution of the ordinary differential equation*

$$s\hat{u}(x_1) = A\frac{d\hat{u}}{dx_1}(x_1) + \mathbb{i}\sum_{j=2}^{d} B_j\omega_j\hat{u}(x_1), \tag{10.7}$$

then $u(x,t)$ defined by

$$u(x,t) = e^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}\hat{u}(x_1)$$

satisfies (10.5).

Proof. The proof follows by substitution. The $u(x,t)$ defined above satisfies

$$\begin{aligned}
\frac{\partial u}{\partial t} &= su(x,t) = e^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}[s\hat{u}(x_1)] \\
&= e^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}A\frac{d\hat{u}}{dx_1} + e^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}\,\mathbb{i}\sum_{j=2}^{d} B_j\omega_j\hat{u}(x_1) \\
&= Ae^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}\frac{d\hat{u}}{dx_1} + \sum_{j=2}^{d} B_j(\mathbb{i}\,\omega_j)e^{st}e^{\mathbb{i}\sum_{j=2}^{d}\omega_j x_j}\hat{u}(x_1) \\
&= A\frac{\partial u}{\partial x_1}(x,t) + \sum_{j=2}^{d} B_j(\mathbb{i}\,\omega_j)u(x,t).
\end{aligned}$$

The definition of $u(x,t)$ implies $\frac{\partial u}{\partial x_j} = \mathbb{i}\,\omega_j u$ for $j = 2, \cdots, d$, thus

$$\frac{\partial u}{\partial t} = A\frac{\partial u}{\partial x_1} + \sum_{j=2}^{d} B_j\frac{\partial u}{\partial x_j}. \tag{10.8}$$

□

The above proof is quite straightforward. However, it does not tell us anything about the relation between the original partial differential equation (10.5) and the ordinary differential equation satisfied by $\hat{u}(x_1)$. Actually, this ordinary differential equation is obtained through a transformation of $u(x,t)$. Consider the Fourier transform of $u(x,t)$ applied to the coordinates $x_2, \cdots, x_d$, and Laplace transform on the time variable. Call $\tilde{u}(x_1, \omega, s)$ the resulting transform, where $\omega = (\omega_2, \cdots, \omega_d)$ denotes the Fourier variables. Then we get:

$$s\tilde{u}(x_1,\omega,s) = A\frac{\partial}{\partial x_1}\tilde{u}(x_1,\omega,s) + \mathbb{i}\sum_{j=2}^{d} B_j\omega_j\tilde{u}(x_1,\omega,s).$$

The difference between this equation and (10.7) is a subtle one: here ω and s are variables, but in (10.7) they play the role of parameters. Although we will use (10.7), it is important to keep in Inind that these parameters contain relevant information about the solution $u(x,t)$. For instance, considering $\omega_2 = 0$ would yield a solution $u(x,t)$ that does not depend on the corresponding variable x_2, and we would be studying a $(d-1)$-dimensional problem instead of the original d-dimensional problem. We shall therefore allow ω to be arbitrary, not fixed at any particular value. The parameters in (10.7) is related to the initial condition $u(x,0)$. For example, if we consider the solution $\hat{u}(x_1)$ for $s = 0$, we get a function $u(x,t)$ in (10.8) that is independent of time. Solutions that are time independent correspond to particular initial conditions. In general, given a value for ω, the value of s for which we define $\hat{u}(x_1)$ will be related to a particular initial condition.

Before stating the next result, we recall from the definition of well posedness that the problem (10.5) is not well posed if for every pair of constants K and α, there is a bounded initial condition $f(x)$ such that $\|u(t)\| > Ke^{\alpha t}\|f\|$ where $u(x,t)$ satisfies (10.5) and $u(x,0) = f(x)$.

Notice now that for all solutions of the form (10.8) we have for each $x = (x_1, x_2, \cdots, x_d)$, $|u(x,0)| = |\hat{u}(x_1)|$ and therefore $\|u(0)\| = \|\hat{u}\|$.

Lemma 10.2. *The problem* (10.5) *is not well posed if for some s with $Re(s) > 0$,* (10.7) *has a bounded solution $\hat{u}(x_1)$.*

Proof. Suppose $\hat{u}(x_1)$ satisfies (10.7) for some s with $Re(s) > 0$, then

$$u(x,t) = e^{st}\exp(\mathbb{i}\sum_{j=2}^{d}\omega_j x_j)\hat{u}(x_1)$$

is a solution to (10.5). Moreover, if $u(x,t)$ is a solution to (10.5) then so is $w_\beta(x,t) = u(\beta x, \beta t)$ for any $\beta > 0$. Thus for any fixed $\beta > 0$, $f(x) = \exp(\mathbb{i}\sum_{j=2}^{d}\omega_j\beta x_j)\hat{u}(\beta x_1)$ is a bounded initial condition, for which a solution to (10.5) is

$$u(x,t) = e^{s\beta t}\exp(\mathbb{i}\sum_{j=2}^{d}\omega_j\beta x_j)\hat{u}(\beta x_1) = e^{s\beta t}f(x).$$

Let T be fixed. For any K and α, we can find a β such that $e^{Re(s)\beta t} > Ke^{\alpha T}$. For this β, $f(x) = \exp(\mathbb{i}\sum_{j=2}^{d}\omega_j\beta x_j)\hat{u}(\beta x_1)$ is a bounded initial condition, and $u(x,t) = e^{s\beta t}f(x)$ is a solution. Moreover,

$$\|u(x,t)\| = e^{Re(s)\beta t}\|f(x)\| > Ke^{\alpha t}\|f(x)\|, \quad \forall t \in [0,T].$$

□

Now we have stated a necessary condition for the problem to be ill-posed. But how does this relate to the boundary operator L? In what follows, we focus on the characterization of such solutions $\hat{u}(x_1)$ of the ordinary differential equation (10.7) that are bounded and correspond to a value of s with $Re(s) > 0$, requiring that they also satisfy the boundary conditions.

Collecting the results obtained so far, we conclude that the problem (10.5) with boundary condition (10.6) will not be well posed if for some s with $Re(s) > 0$ the problem:

$$s\hat{u}(x) = A\frac{d\hat{u}}{dx}(x) + \mathbb{i}\sum_{j=2}^{d} B_j\omega_j\hat{u}(x), \quad x \geq 0, \tag{10.9}$$

$$\hat{u}^I(0) = Lu^{II}(0), \tag{10.10}$$

has a bounded solution $\hat{u}(x)$. We have changed now the notation x_1 by x, since the problem is already a one-dimensional one and the variables x_j, $j > 2$ do not appear. Equations (10.9) represent, for given ω and s, an initial valued ordinary differential equation for $\hat{u}(x)$. Since A is a diagonal matrix with no zero eigenvalue, A^{-1} exists and we may rewrite (10.9) as:

$$\frac{d}{dx}\hat{u}(x) = M\hat{u}, \tag{10.11}$$

where

$$M = A^{-1}(sI - \mathbb{i}\sum_{j=2}^{d}\omega_j B_j),$$

is a $p\times p$ matrix depending on s and ω, but not on x. Let $\kappa_\nu(s)$ ($\nu = 1, \cdots, p$) denote the eigenvalues of M (they also depend on ω but we do not make this dependence explicit in our notation).

Notice that if $\phi_\nu(s)$ is the eigenvector such that

$$M\phi_\nu(s) = \kappa_\nu(s)\phi_\nu(s),$$

then the function $e^{\kappa_\nu(s)x}\phi_\nu(s)$ satisfies (10.11):

$$\begin{aligned}\frac{d}{dx}e^{\kappa_\nu(s)x}\phi_\nu(s) &= \kappa_\nu(s)e^{\kappa_\nu(s)x}\phi_\nu(s)\\ &= e^{\kappa_\nu(s)x}M\phi_\nu(s)\\ &= Me^{\kappa_\nu(s)x}\phi_\nu(s).\end{aligned}$$

In order to characterize the general for in of the bounded solutions of (10.11) we make the use of the following result without proof (How to prove it? Hint: the number of eigenvalues with positive real parts is a continuous function of ω. See Lemma 10.5.4 in [4] for the detailed proof):

Lemma 10.3. *For every s such that $Re(s) > 0$, M has r eigenvalues with negative real part and no purely imaginary eigenvalues.*

We will order the eigenvalues of M such that:

$$Re[\kappa_\nu(s)] < 0, \quad \nu = 1, \cdots, r,$$

$$Re[\kappa_\nu(s)] > 0, \quad \nu = r+1, \cdots, p.$$

Since the eigenvalues of a matrix are continuous functions of the entries of the matrix, we consider each $\kappa_\nu(s)$ as a continuous function of s. For $Re(s) > 0$, the general solution of (10.11) is of the form:

$$\hat{u}(x) = \sum_{\nu=1}^{p} \rho_\nu e^{\kappa_\nu(s)x} \phi_\nu(s)$$

where ρ_ν are coefficients to be determined by the initial condition. An alternative way to write the solution is to use the matrix exponential $\hat{u}(x) = e^{Mt}\hat{u}(0)$.

From the above expression, since $Re[\kappa_\nu(s)] > 0$ for $\nu > r$, it follows that in order for $\hat{u}(x)$ to be a bounded solution of x, $\rho_\nu = 0$ for $\nu = r+1, \cdots, p$.

Therefore if $Re(s) > 0$, the bounded solutions of (10.11) have the general form

$$\hat{u}(x) = \sum_{\nu=1}^{r} \rho_\nu e^{\kappa_\nu(s)x} \phi_\nu(s). \tag{10.12}$$

In addition, we require $\hat{u}(x)$ to satisfy (10.10). Note that if we denote, for each ν, $\phi^I(s)$ the first r copponents of $\phi(s)$ and $\phi^{II}(s)$ the last $(p-r)$ components, (10.10) reduces to:

$$\sum_{\nu=1}^{r} \rho_\nu (\phi_\nu^I(s) - L\phi_\nu^{II}(s)) = 0.$$

which is a $r \times r$ linear system for ρ_ν. Thus it can be written as

$$Q(s)\rho = 0,$$

where

$$\rho = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_r \end{pmatrix}$$

and the matrix $Q(s)$ has the form

$$Q_{\nu j}(s) = e^{\kappa_\nu(s)}[\phi_\nu^I - L\phi_\nu^{II}]_j.$$

Therefore, in order for (10.12) to be a non-trivial bounded solution of (10.9), it is necessary that $det[Q(s)] = 0$, so that $Q(s)\rho = 0$ admits a nontrivial solution. We summarize the result in the following.

Theorem 10.3. *If $det[Q(s)] = 0$ for some s with $Re(s) > 0$, then the problem* (10.5) *with boundary conditions* (10.6) *is not well posed.*

In practical situations, it is often impossible to evaluate an explicit expression for $det[Q(s)]$ as a function of s, so the verification of the conditions in the Theorem above is not straightforward. The main difficulty is that one generally obtains an equation for the eigenvalues $\kappa_\nu(s)$ which also depends on ω and other coefficients, and it may be difficult to characterize which are the eigenvalues for which $Re[\kappa_\nu(s)] < 0$. We shall give a detailed example on how Kreiss theory is applied for a specific problem, but before, we state without proof the conditions for the problem to be well posed.

Theorem 10.4. Kreiss. *If $det[Q(s)] \neq 0$ for all complex numbers s with $Re(s) \geq 0$ then the problem* (10.5) *and* (10.6) *is strongly well posed.*

And finally, we have:

Theorem 10.5. Hersch. *If $det[Q(s)] \neq 0$ for all complex numbers s with $Re(s) > 0$ and $det[Q(s_0)] = 0$ for some purely imaginary s_0, then the problem* (10.5) *and* (10.6) *is weakly well posed.*

As already mentioned, if we consider $\omega_j = 0$ for all $j \geq 2$, then the problem reduces to a one-dimensional problem. For this case we studied in previous section and showed that the boundary condition $u^I(0,t) = Lu^{II}(0,t)$ preserves well posedness. Therefore, it is always true that $det[Q(s)] \neq 0$ for all s with $Re(s) \geq 0$ if $\omega_j = 0$ for all $j \geq 2$.

As an example, we consider the two dimensional system (d = 2):

$$\frac{\partial}{\partial t}\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\frac{\partial}{\partial x}\begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\frac{\partial}{\partial y}\begin{pmatrix} u \\ v \end{pmatrix}, \qquad \begin{matrix} 0 \leq x \leq \infty \\ -\infty \leq y \leq \infty \end{matrix}$$

so that $p = 2$. Here A is already diagonal and B is symmetric. The number of left boundary conditions is $r = 1$, and we consider it to be of the form:

$$u(0, y, t) = lv(0, y, t),$$

where l is a real number. The ODE (10.9) is

$$s\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\frac{d}{dx}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} + \mathbb{i}\,\omega\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix}.$$

And (10.10) becomes

$$\hat{u}(0) = l\hat{v}(0).$$

In this example $A = A^{-1}$, so we obtain

$$\begin{aligned}\begin{pmatrix}\hat{u}\\ \hat{v}\end{pmatrix}_x &= \begin{pmatrix}-1 & 0\\ 0 & 1\end{pmatrix}\left(sI - \begin{pmatrix}0 & \mathbb{i}\,\omega\\ \mathbb{i}\,\omega & 0\end{pmatrix}\right)\begin{pmatrix}\hat{u}\\ \hat{v}\end{pmatrix}\\ &= \begin{pmatrix}-s & \mathbb{i}\,\omega\\ -\mathbb{i}\,\omega & s\end{pmatrix}\begin{pmatrix}\hat{u}\\ \hat{v}\end{pmatrix},\end{aligned}$$

which gives an explicit form of M. The eigenvalues of M must satisfy

$$\kappa_\nu^2(s) = s^2 + \omega^2, \tag{10.13}$$

where s is complex and ω is real.

It is clear now that even for this simple example, it is not convenient to try to express $\kappa_1(s) = Re[\kappa_1(s)] + \mathbb{i}[\kappa_1(s)]$ as a function of s such that if $Re(s) > 0$ then $Re[\kappa_1(s)] < 0$. Instead, we use the equation (10.13) which is satisfied by $\kappa_1(s)$.

The eigenvector $\phi_\nu(s)$ satisfies

$$\begin{pmatrix}-s & \mathbb{i}\,\omega\\ -\mathbb{i}\,\omega & s\end{pmatrix}\begin{pmatrix}\phi_\nu^I\\ \phi_\nu^{II}\end{pmatrix} = \kappa_\nu(s)\begin{pmatrix}\phi_\nu^I\\ \phi_\nu^{II}\end{pmatrix},$$

which gives two linearly dependent equations equivalent to

$$-(s + \kappa_\nu(s))\phi_\nu^I(s) + \mathbb{i}\,\omega\phi_\nu^{II}(s) = 0.$$

Recall that eigenvectors are defined up to a normalization constant, so that in general we have that:

$$\begin{aligned}\phi_\nu^I(s) &= \mathbb{i}\,\omega,\\ \phi_\nu^{II}(s) &= s + \kappa_\nu(s).\end{aligned}$$

Of course for $\phi_\nu(s)$ to be an eigenvector, it must be a nonzero vector. This rules out the case $\omega = 0$, for which $\kappa_1(s) = -s$, $\kappa_2(s) = s$ and $\phi_1(s)$ is the zero vector. As already mentioned before, if $\omega = 0$ we are reducing the problem to a one-dimensional one, which is well posed according to the results of last section.

According to our previous discussion, we have that for any s with $Re(s) > 0$, the bounded solutions of the ODE are of the form:

$$\begin{pmatrix}\hat{u}\\ \hat{v}\end{pmatrix} = \rho_1\begin{pmatrix}\mathbb{i}\,\omega\\ s + \kappa_1(s)\end{pmatrix}e^{\kappa_1(s)x}$$

where $Re[\kappa_1(s)] < 0$ if $Re(s) > 0$. So far we have not made use of the boundary condition. Now we want to see is for some s with $Re(s) > 0$, there

is a nontrivial solution of the form above which also satisfies the boundary condition. If so, the problem is not well posed.

Using the boundary condition at $x = 0$, we get

$$\rho_1 \mathbb{i}\,\omega = l\rho_1(s + \kappa_1(s)),$$

where $Q(s) = \mathbb{i}\,\omega - l(s+\kappa_1(s))$ is a 1×1 matrix. The condition $det[Q(s)] = 0$ becomes

$$\mathbb{i}\,\omega = l(s + \kappa_1(s)). \tag{10.14}$$

In other words, the statement that $det[Q(s)] = 0$ *for some* s *with* $Re(s) > 0$ is equivalent to *equations* (10.13) *and* (10.14) *have a solution* $(\kappa_1(s), s)$ *satisfying* $Re(s) > 0$ *and* $Re[\kappa_1(s)] < 0$.

Notice that without the constraint $Re[\kappa_1(s)] < 0$, the solution to both (10.13) and (10.14) may be $\kappa_2(s)$, which is not the one giving bounded solutions to the ODE (10.9).

The problem is clearly an algebraic one, we can express (10.14) as $\kappa_\nu(s) = \frac{1}{l}\,\mathbb{i}\,\omega - s$ and plug it into (10.13). Later we determine whether the solution corresponds to $\nu = 1$ or $\nu = 2$. We get

$$-\frac{\omega^2}{l^2} - \frac{2\,\mathbb{i}\,\omega s}{l} + s^2 = s^2 + \omega^2,$$

$$\frac{2\,\mathbb{i}\,s}{l} = -\omega\frac{1 + l^2}{l^2}.$$

So the value of s satisfying both (10.13) and (10.14) must be

$$s_0 = \mathbb{i}\,\omega\frac{l^2 + 1}{2l}.$$

And

$$\kappa_\nu(s_0) = \mathbb{i}\,\omega\frac{1 - l^2}{2l}. \tag{10.15}$$

Since s_0 is purely imaginary, the first conclusion is that there is no solution to (10.13) and (10.14) with $Re(s) > 0$. Thus Theorem 10.3 does not apply. The only solution to (10.13) and (10.14) holds $Re(s_0) = 0$ and $Re[\kappa_\nu(s_0)] = 0$. In order to determine whether Theorem 10.4 or Theorem 10.5 is applicable, we must know if $\nu = 1$ or $\nu = 2$.

As defined above, $\kappa_\nu(s)$ is the continuous function of s which is an eigenvalue of M and such that $Re[\kappa_1(s)] < 0 < Re[\kappa_2(s)]$ for $Re(s) > 0$.

In order to determine whether $\kappa_\nu(s_0)$ in (10.15) is $\kappa_1(s_0)$ or $\kappa_2(s_0)$, we make a perturbation analysis. Indeed, by the definition of $\kappa_\nu(s)$, we have for any positive real number $\alpha > 0$:

$$Re[\kappa_1(s_0 + \alpha)] < 0 < Re[\kappa_2(s_0 + \alpha)]. \tag{10.16}$$

Since $\kappa_\nu(s)$ are solutions to (10.13), then both $\kappa_\nu(s)$ are continuous functions of s, so that for small $\alpha > 0$,

$$\kappa_\nu(s_0 + \alpha) = \kappa_\nu(s_0) + \beta_\nu,$$

where $|\beta_\nu| = |\kappa_\nu(s_0 + \alpha) - \kappa_\nu(s_0)|$ is of the order α. We think of perturbing s_0 along a line so that $s_\alpha \equiv s_0 + \alpha$ only differs from s_0 in the real part, it may be that $\beta_\nu = \kappa_\nu(s_0 + \alpha) - \kappa_\nu(s_0)$ has a nonzero imaginary part. In any cases, $|\beta_\nu| = \mathcal{O}(\alpha)$, and so $|\beta_\nu|$ will be negligible when α is very small.

The sign of $Re(\beta_\nu)$ will give us the answer of which one is $\kappa_\nu(s_0)$, since by (10.16):

$$Re(\beta_1) < 0 < Re(\beta_2).$$

The eigenvalues $\kappa_\nu(s_0 + \alpha)$ must also satisfy

$$\kappa_\nu^2(s_0 + \alpha) = (s_0 + \alpha)^2 + \omega^2,$$

thus

$$\beta_\nu^2 + 2\beta_\nu\kappa_\nu(s_0) + \kappa_\nu(s_0)^2 = s_0^2 + 2\alpha s_0 + \alpha^2 + \omega^2,$$

and since $\kappa_\nu^2(s_0) = s_0^2 + \omega^2$, neglecting the terms of order $\mathcal{O}(\alpha^2)$, this yields:

$$2\beta_\nu\kappa_\nu(s_0) = 2\alpha s_0.$$

Using (10.15) and $s_0 = \mathbb{i}\,\omega\frac{l^2+1}{2l}$, we get

$$\beta_\nu = \alpha\frac{l^2+1}{l^2-1}.$$

In the last expression we have assumed $l \neq 1$. From this expression we obtain:

- If $|l| < 1$, then $Re(\beta_\nu) > 0$ implies $\nu = 2$, and therefore the solution to (10.13) and (10.14) is $\kappa_2(s_0)$. We conclude that $det[Q(s)] \neq 0$ for all s with $Re(s) \geq 0$ and therefore the problem is well posed.
- If $|l| > 1$, then $Re(\beta_\nu) < 0$, so that $\kappa_1(s_0)$ is a solution to (10.14) yielding that $det[Q(s_0)] = 0$ for $s_0 = \mathbb{i}\,\omega\frac{l^2+1}{2l}$. The problem is therefore weakly but not strongly well posed.

As for the case $l = 1$, we have $s_0 = \mathbb{i}\,\omega$, $\kappa_\nu(s_0) = 0$, so the perturbation analysis will not give the information about ν being 1 or 2. However, the case $l = 1$ turns out to be simple enough so that we can apply the energy estimates directly and do not have to use Kreiss theory. We show how to handle this case for periodic boundary conditions instead of general inifinite domains:

Claim: *The problem*

$$\frac{\partial}{\partial t}\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\frac{\partial}{\partial x}\begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\frac{\partial}{\partial y}\begin{pmatrix} u \\ v \end{pmatrix},$$

defined in the space domain $x \in [0,1], y \in [0,2\pi]$ *satisfying the periodicity condition in* y*:*

$$u(x,0,t) = u(x,2\pi,t), \quad v(x,0,t) = v(x,2\pi,t)$$

with left boundary condition and right boundary condition given by

$$u(0,y,t) = lv(0,y,t), \quad l = 1,$$

$$v(1,y,t) = 0,$$

is strongly well posed.

Remark 10.1. *Periodic boundary conditions on* y *are assumed in order to simplify the computations. We have taken the finite domain* $0 \leq x \leq 1$*, which requires a right boundary condition for* v *(in the same way at we need a left boundary condition for* u *at* $x = 0$*). We use* $v(1,y,t) = 0$ *which can be generalized to the natural condition* $\lim_{x\to+\infty} v(x,y,t) = 0$ *when we consider* $x \geq 0$*. This condition ensures boundedness.*

Proof. The equations imply

$$u_t = -u_x + v_y, \quad v_y = v_x + u_y.$$

Define the energy by

$$E(t) = \int_0^{2\pi} dy \int_0^1 dx \left(u^2(x,y,t) + v^2(x,y,t)\right).$$

Then we have

$$\frac{1}{2}E'(t) = \int_0^{2\pi} dy \int_0^1 dx \, (uu_t + vv_t)$$

thus

$$\begin{aligned} E'(t) &= \int_0^{2\pi} dy \int_0^1 dx \left[-(u^2)_x + (v^2)_x\right] + 2\int_0^{2\pi} dy \int_0^1 dx \, [uv_y + vu_y] \\ &= \int_0^{2\pi} [-u^2(1,y,t) + u^2(0,y,t) + v^2(1,y,t) - v^2(0,y,t)]dy + 2\int_0^{2\pi} dy \int_0^1 dx[(uv)_y]. \end{aligned}$$

Plugging in the boundary conditions, we get

$$E'(t) = -\int_0^{2\pi} u^2(1,y,t)dy + \int_0^1 [u(x,2\pi,t)v(x,2\pi,t) - u(x,0,t)v(x,0,t)]dx. \tag{10.17}$$

The first term is non-positive and the second integral vanishes due to y-periodicity, so $E'(t) \leq 0$, proving the claim. □

When we considered the infinite domains, the integrations were performed on $x \in \mathbb{R}$ and $y \in \mathbb{R}$. In this case we impose the condition:

$$\lim_{x\to+\infty} u(x,y,t) = \lim_{x\to+\infty} v(x,y,t) = 0,$$

so that in (10.17) the first term would also vanish.

If imposing

$$\lim_{y\to+\infty} u(x,y,t) = \lim_{y\to+\infty} v(x,y,t) = 0,$$

then the second integral in (10.17) also vanishes. In this case we therefore have that the energy is conserved. Notice that for the finite domain case with periodicity conditions on y, any right boundary condition of the form $u(1,y,t) = c$ a constant, will yield well posedness, where now $E'(t) \leq -2\pi c^2$. As mentioned at the beginning of this section, Kreiss theory is based on *looking at one boundary at a time*, and we can now illustrate how things can be put together if we consider the general right boundary condition:

$$v(1,y,t) = ru(1,y,t).$$

Then for $|r| \leq 1$ and $|l| \leq 1$ the problem is strongly well posed. The case where either $|r| > 1$ or $|l| > 1$ is studied through the Kreiss theory, as we did before, for one boundary at a time.

Theorems 10.3 and 10.4 have been extended to more general domains with variable coefficients. For these problems, it is not yet clear what the analog of Theorem 10.4 should be.

11

Selected applications

In this section, we list a few selected applications.

11.1 TV norm minimization and Poisson equation

The Laplacian operator emerges in many classical and popular optimization algorithms for total variation (TV) norm minimization problems. As an example, we will consider the TV norm minimization for image denoising [10, 11]. All derivatives in this section should be understood as weak derivatives as in Chapter 3.

11.1.1 Continuum ROF image denoising model

Consider a rectangular domain $\Omega = [0,1] \times [0,1]$, and a function $u(x,y) \in H^1(\Omega)$, which represents an image with infinite resolution. Then its total variation is defined as

$$\|u\|_{TV} = \iint_\Omega |\nabla u| dxdy,$$

where $\nabla u = (u_x, u_y)$ and $|\nabla u| = \sqrt{|u_x|^2 + |u_y|^2}$. With L^2-norm as $\|u\|_{L^2} = \sqrt{\iint_\Omega |u|^2 dxdy}$, for a given $a(x,y)$, the ROF (Rudin, Osher, and Fatemi, 1992) model [10] is to minimize (over u in a proper function space)

$$\|u\|_{TV} + \frac{1}{2}\lambda \|u - a\|_{L^2}^2,$$

where λ is a fixed parameter.

The function space that the minimizer should belong to, is a subspace of $H^1(\Omega)$ with suitable boundary conditions. For instance, periodic or homogeneous Dirichlet boundary conditions make sense for MRI images, but not for a generic image. For convenience, for a generic image, we just consider homogeneous Neumann boundary conditions, which will naturally emerge

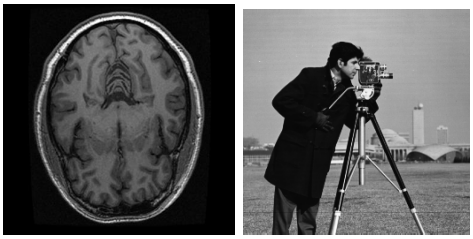


Figure 11.1: Periodic or zero bounary conditions are suitable for MRI images, but not for a generic image.

in the discrete setup as will be seen in the following subsections. See Figure 11.1.

To this end, we define

$$\mathcal{H} = \{u \in H^1(\Omega) : \nabla u \cdot \mathbf{n}|_{\partial\Omega} = 0\},$$

where $\mathbf{n}$ is the unit normal vector of the boundary $\partial\Omega$.

The gradient operator ∇ is a linear mappping, and we use an abstract name for it $\mathcal{K} = \nabla$:

$$\begin{aligned} \mathcal{K} = \nabla : \mathcal{H} &\longrightarrow \mathcal{V} = (L^2(\Omega), L^2(\Omega)) \\ u &\longmapsto \nabla u = (u_x, u_y) \end{aligned}$$

To understand the adjoint operator of $\mathcal{K} = \nabla$, we need the $H(div)$-space:

$$H(div) = \{\mathbf{q} = (q^1, q^2) \in (L^2(\Omega), L^2(\Omega)) : \nabla \cdot (q^1, q^2) \in L^2(\Omega)\} \subset \mathcal{V}.$$

Remark 11.1. *Elements in $H(div)$ are not necessarily in $H^1(\Omega)$. For instance, let $f(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$, then $\mathbf{q}(x, y) := (0, f(x))$ is in $H(div)$ but $f(x) \notin H^1(\Omega)$.*

The divergence operator $\nabla\cdot$ is a linear mapping from $H(div)$ to $L^2(\Omega)$. If we assume suitable boundary conditions for smooth $\mathbf{q}$, then $\mathcal{K}^* = -\nabla\cdot$ is the adjoint operator of $\mathcal{K} = \nabla$ since

$$\langle \mathcal{K}u, \mathbf{q} \rangle := \iint_\Omega \nabla u \cdot \mathbf{q} dxdy = -\iint_\Omega u \nabla \cdot \mathbf{q} dxdy = \langle u, -\nabla \cdot \mathbf{q} \rangle, \forall \mathbf{q} \in (C_0^1(\Omega), C_0^1(\Omega)).$$

11.1.2 Discrete ROF model

Consider an image of size $n \times n$, corresponding to domain $[0, 1] \times [0, 1]$ and a uniform grid $x_i, y_j = (j - 1)h$, $j = 1, \cdots, n$ with $h = \frac{1}{n-1}$. Notice that an image does not have any necessary association of a domain of size

$[0,1]\times[0,1]$, and this assumption of domain $[0,1]\times[0,1]$ should not affect the final implementation.

Recall that D is the finite difference matrix approximating first order derivative, then we have

$$D^T=\begin{pmatrix}-1 & & & & & \\ 1 & -1 & & & & \\ & 1 & -1 & & & \\ & & \ddots & \ddots & & \\ & & & 1 & -1 & \\ & & & & 1 & 0\end{pmatrix}_{n\times n}, D^TD=\begin{pmatrix}1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1\end{pmatrix}.$$

Now for a bivariate function $u(x,y)$, let U be a 2D array with $U(j,i)=u(x_i,y_j)$, then

$$U\approx u\Rightarrow \frac{1}{h}UD^T\approx u_x,\quad \frac{1}{h}DU\approx u_y.$$

Remark 11.2. *We may also choose the notation $U(i,j)=u(x_i,y_j)$, then $\frac{1}{h}DU\approx u_x$.*

For function $u(x,y)$ and $a(x,y)$, we have

$$||u||_{TV}\approx\sum_i\sum_j h^2\sqrt{|u_x(x_i,y_j)|^2+|u_y(x_i,y_j)|^2}$$

$$||u-a||_{L^2}^2\approx\sum_i\sum_j h^2|u(x_i,y_j)-a(x_i,y_j)|^2.$$

Introduce $U_x:=\frac{1}{h}UD^T$ and $U_y:=\frac{1}{h}DU$. Let A be a 2D array with $A(j,i)=a(x_i,y_j)$, then the discrete ROF model is

$$\min_{U\in\mathbb{R}^{n\times n}}\sum_i\sum_j\left(h^2\sqrt{U_x(j,i)^2+U_y(j,i)^2}+h^2\frac{1}{2}\lambda|U(j,i)-A(j,i)|^2\right)$$

$$=\min_{U\in\mathbb{R}^{n\times n}}\sum_i\sum_j\left(h\sqrt{[UD^T](j,i)^2+[DU](j,i)^2}+h^2\frac{1}{2}\lambda|U(j,i)-A(j,i)|^2\right).$$

Notice that its minimizer does not depend on the choice of h if $\lambda=\frac{C}{h}$ for some constant C. An image has no physical grid spacing anyway, so h should be arbitrary, for which we should take $\lambda=\frac{C}{h}$. In practice, $C=10$ usually produces a reasonable result. See Figure 11.2 and Figure 11.3.

11.1.3 Primal, dual and primal-dual forms

Using all notation above, the discrete ROF model can be written as

$$\min_{U\in\mathbb{R}^{n\times n}} f(\mathcal{K}U)+g(U), \tag{11.1a}$$

(a) Noisy Image

(b) $C = 4$

(c) $C = 8$

(d) $C = 12$

Figure 11.2: ROF solutions using isotropic TV-norm with different $\lambda = \frac{C}{h}$.

where $\mathcal{K} = \nabla_h : \mathbb{R}^{n\times n} \longrightarrow \mathbb{R}^{2(n\times n)}$ is a linear mapping

$$\mathcal{K}(U) = \nabla_h U = \frac{1}{h}(UD^T, DU), \tag{11.1b}$$

and

$$f(P,Q) = \sum_{i,j} h^2\sqrt{P^2(i,j) + Q^2(i,j)}, \quad g(U) = \lambda \sum_{i,j} h^2 |U(i,j) - a(i,j)|^2. \tag{11.1c}$$

It is straightforward to verify that the adjoint operator of $\mathcal{K}$ is given by

$$\begin{aligned} \mathcal{K}^* = -\nabla_h \cdot : \mathbb{R}^{2(n\times n)} &\longrightarrow \mathbb{R}^{n\times n} \\ (P,Q) &\longmapsto \frac{1}{h}(PD + D^TQ) \end{aligned}$$

The convex minimization (11.1a) is called *primal* form. To solve (11.1a),

(a) Noisy Image

(b) $C = 4$

(c) $C = 8$

(d) $C = 12$

Figure 11.3: ROF solutions using isotropic TV-norm with different $\lambda = \frac{C}{h}$.

equivalently we can solve its *dual* form

$$-\min_{\mathbf{P}\in\mathbb{R}^{2(n\times n)}} f^*(\mathbf{P}) + g^*(-\mathcal{K}^*\mathbf{P}). \tag{11.2}$$

Both (11.1a) and (11.2) are also equivalent to the *primal-dual* form:

$$\min_{U\in\mathbb{R}^{n\times n}} \max_{\mathbf{P}\in\mathbb{R}^{2(n\times n)}} \langle \mathcal{K}U, \mathbf{P}\rangle - f^*(\mathbf{P}) + g(U), \tag{11.3}$$

Recall that the minimizer U^* to (11.1a) and the minimizer $\mathbf{P}^*$ to (11.2) are related via the optimality condition in the primal-dual form in the previous chapter. To recover the physical image U from $\mathbf{P}$, we need the relation obtained from the Legendre transform of $g(U)$:

$$0 \in \mathcal{K}^*P + \partial g(U),$$

which gives

$$0 = \mathcal{K}^*P + \lambda(U - A) \Rightarrow U = A - \frac{1}{\lambda}\mathcal{K}^*P.$$

11.1.4 ADMM and Douglas-Rachford splitting

Both alternating direction method of multipliers (ADMM) (Glowenski and Marrocco 75) and Douglas-Rachford splitting (Lions and Mercier 79) are popular and successful splitting convex minimization algorithms, and they are equivalent in the following sense:

$$\text{ADMM on primal} \Leftrightarrow \text{Douglas-Rachford on dual},$$

$$\text{ADMM on dual} \Leftrightarrow \text{Douglas-Rachford on primal}.$$

The ADMM method is for solving $\min f(x) + g(y)$ under the linear constraint $Ax + By = b$, and it is given as

$$x_{k+1} = \text{argmin}_x f(x) + g(y_k) - \langle \lambda_k, Ax + By_k - b\rangle + \frac{\beta}{2}\|Ax + By_k - b\|^2$$

$$y_{k+1} = \text{argmin}_y f(x_{k+1}) + g(y) - \langle \lambda_k, Ax_{k+1} + By - b\rangle + \frac{\beta}{2}\|Ax_{k+1} + By - b\|^2$$

$$\lambda_{k+1} = \text{argmax}_\lambda f(x_{k+1}) + g(y_{k+1}) - \langle \lambda, Ax_{k+1} + By_{k+1} - b\rangle - \frac{1}{2\beta}\|\lambda - \lambda_k\|^2,$$

where λ is the Lagrangian multiplier and $\beta > 0$ is a step size.

11.1.5 Discrete Laplacian in ADMM on primal

By plugging in the linear constraint $Ax + By = b$ as $-\mathbf{P} + \mathcal{K}U = 0$, ADMM applied on $f(\mathbf{P}) + g(U)$ in the primal form (11.1a) becomes

$$\mathbf{P}_{k+1} = \text{argmin}_{\mathbf{P}} f(\mathbf{P}) + g(U_k) - \langle \lambda_k, -\mathbf{P} + \mathcal{K}U_k\rangle + \frac{\beta}{2}\| - \mathbf{P} + \mathcal{K}U_k\|^2$$

$$U_{k+1} = \text{argmin}_U f(\mathbf{P}_{k+1}) + g(U) - \langle \lambda_k, -\mathbf{P}_{k+1} + \mathcal{K}U\rangle + \frac{\beta}{2}\| - \mathbf{P}_{k+1} + \mathcal{K}U\|^2$$

$$\lambda_{k+1} = \text{argmax}_\lambda f(\mathbf{P}_{k+1}) + g(U_{k+1}) - \langle \lambda, -\mathbf{P}_{k+1} + \mathcal{K}U_{k+1}\rangle - \frac{1}{2\beta}\|\lambda - \lambda_k\|^2.$$

To implement the second line, by ignoring constants, we consider

$$U_{k+1} = \text{argmin}_U g(U) - \langle \lambda_k, \mathcal{K}U\rangle + \frac{\beta}{2}\|\mathcal{K}U - \mathbf{P}_{k+1}\|^2.$$

Notice that $g(U)$ is a simple quadratic function, thus the minimizer is obtained by finding critical point, for which we need to take derivative of $\|\mathcal{K}U - \mathbf{P}_{k+1}\|^2$ w.r.t. U:

$$\frac{\partial}{\partial U}\langle \mathcal{K}U - \mathbf{P}_{k+1}, \mathcal{K}U - \mathbf{P}_{k+1}\rangle = h^2(2\mathcal{K}^*\mathcal{K}U - 2\mathcal{K}^*\mathbf{P}_{k+1}).$$

So the second line can be equivalently written as

$$\lambda(U_{k+1} - A) - \mathcal{K}^*\lambda_k + \beta\mathcal{K}^*\mathcal{K}U_{k+1} - 2\mathcal{K}^*\mathbf{P}_{k+1} = 0$$

which is

$$(\lambda I + \beta \mathcal{K}^*\mathcal{K})U_{k+1} = -\lambda A + \mathcal{K}^*\lambda_k + 2\mathcal{K}^*\mathbf{P}_{k+1}.$$

Notice that $\mathcal{K}^*\mathcal{K} = -\Delta_h$ is precisely the discrete Laplacian with purely Neumann boundary conditions, and $\lambda I - \beta\Delta_h$ can be inverted similar as in Chapter 2 (notice that eigenvalues of $\lambda I - \beta\Delta_h$ are strictly positive).

11.1.6 Discrete Laplacian in Douglas-Rachford on the dual

Though ADMM on the primal problem is mathematically equivalently to the Douglas-Rachford splitting on the dual problem, the Poisson equation arises in a seemingly very different manner.

Using notation in this section, for the TV-norm denoising problem of a 2D image $B \in \mathbb{R}^{n\times n}$, the primal problem is equivalently written as

$$\min_{U\in\mathbb{R}^{n\times n}} \|\mathcal{K}U\|_1 + \frac{\lambda}{h}\|U - B\|_F^2,$$

where $\|\cdot\|_F$ is the Frobenius norm for a matrix $\|U-B\|_F = \sqrt{\sum_{i,j} |U(i,j) - a(i,j)|^2}$ and the 1-norm for a pair of matrices $\mathbf{V} = (P, Q)$ is

$$F(\mathbf{V}) = \|(P,Q)\|_1 = \sum_{i,j} \sqrt{P(i,j)^2 + Q(i,j)^2}.$$

The convex conjugate of $F(\mathbf{V})$ is

$$F^*(\mathbf{V}) = \sum_{i,j} \iota_{\{P(i,j)^2+Q(i,j)^2\le 1\}}.$$

Up to a constant shift, the dual problem can be written as

$$-\min F^*(\mathbf{V}) + \frac{h}{2\lambda}\|\mathcal{K}^*\mathbf{V} - \frac{\lambda}{h}B\|_F^2.$$

Problem 11.1. *Derive the dual problem.*

The proximal operator of F^* can be easily computed as the projection to the unit ball for each entry (i, j).

Now consider the proximal operator of the function

$$G^*(\mathbf{V}) = \frac{h}{2\lambda}\|\mathcal{K}^*\mathbf{V} - \frac{\lambda}{h}B\|_F^2,$$

which is written as

$$\mathrm{Prox}_{G^*}^{\eta}(\mathbf{W}) = \operatorname*{argmin}_{\mathbf{V}} \frac{h}{2\lambda}\|\mathcal{K}^*\mathbf{V} - \frac{\lambda}{h}B\|_F^2 + \frac{1}{2\eta}\|\mathbf{V} - \mathbf{W}\|_F^2.$$

Let $\mathbf{V} = \mathrm{argmin}$, then the critical point equation gives

$$\frac{h}{\lambda}\mathcal{K}(\mathcal{K}^*\mathbf{V} - \frac{\lambda}{h}B) + \frac{1}{\eta}(\mathbf{V} - \mathbf{W}) = 0$$

$$\Rightarrow (\frac{1}{\eta}\mathbb{I} + \frac{h}{\lambda}\mathcal{K}\mathcal{K}^*)\mathbf{V} = \mathcal{K}B + \frac{1}{\eta}\mathbf{W}.$$

We need to solve $\mathbf{V}$ in an equation in the form

$$\mathcal{K}\mathcal{K}^*\mathbf{V} + \beta\mathbf{V} = \mathbf{F}$$

where $\beta = \frac{\lambda}{\eta h}$ and $\mathbf{F} = \eta\mathcal{K}B + \mathbf{W}$ is some known vector field. At first glance, this corresponds to an equation

$$\nabla(-\nabla \cdot \boldsymbol{p}) + \beta\boldsymbol{p} = \boldsymbol{f},$$

which is a harder equation to solve due to the mixed second order derivatives, compared to the Poisson equation.

However, to solve this seemingly difficult equation, we can just compute $(-\Delta_h + \beta\mathbb{I})^{-1}$, mainly due to a simple linear algebra fact:

Lemma 11.1. *For a linear operator $\mathcal{K} : \mathbb{R}^{n\times n} \longrightarrow \mathbb{R}^{2(n\times n)}$, assume $\mathcal{K}^*\mathcal{K}$ has an inverse or a right pseudo-inverse $(\mathcal{K}^*\mathcal{K})^{-1}$, then the solution to the equation $\mathcal{K}\mathcal{K}^*\mathbf{V} + \beta\mathbf{V} = \mathbf{F}$ can be written as*

$$\mathbf{V} = \frac{1}{\beta}[\mathbf{F} - \mathcal{K}(\beta\mathbb{I} + \mathcal{K}^*\mathcal{K})^{-1}\mathcal{K}^*\mathbf{F}].$$

Proof. The kernal of $\mathcal{K}^*$ is orthogonal to the range of $\mathcal{K}$ (column space of a matrix K is orthogonal to the left null space of K), thus

$$\mathbb{R}^{2(n\times n)} = \text{Kernel}(\mathcal{K}^*) \oplus \text{Range}(\mathcal{K}),$$

which implies a very useful fact (corresponding to Helmholtz decomposition for suitable vector fields):

$$\mathbf{V} = \mathcal{K}W + \mathbf{G} = \nabla_h W + \mathbf{G}, \quad \text{where } \mathbf{G} \in \text{Kernel}(\mathcal{K}^*), \text{ i.e., } \nabla_h \cdot \mathbf{G} = 0.$$

Apply $\mathcal{K}^*$ to both sides of the equation, we can first solve for W as follows

$$\mathcal{K}^*\mathcal{K}\mathcal{K}^*\mathbf{V} + \beta\mathcal{K}^T\mathbf{V} = K^*\mathbf{F} \Rightarrow \mathcal{K}^*\mathbf{V} = (\beta\mathbb{I} + \mathcal{K}^*\mathcal{K})^{-1}\mathcal{K}^*\mathbf{F},$$

$$\mathbf{V} = \mathcal{K}W + \mathbf{G} \Rightarrow \mathcal{K}^*\mathbf{V} = \mathcal{K}^*(\mathcal{K}W + \mathbf{G}) \Rightarrow W = (\mathcal{K}^*\mathcal{K})^{-1}\mathcal{K}^*\mathbf{V} = (\mathcal{K}^*\mathcal{K})^{-1}(\beta\mathbb{I} + \mathcal{K}^*\mathcal{K})^{-1}\mathcal{K}^*\mathbf{F}.$$

Then we can solve $\mathbf{G}$ by

$$\mathcal{K}\mathcal{K}^*(\mathcal{K}W + \mathbf{G}) + \beta(\mathcal{K}W + \mathbf{G}) = \mathbf{F} \Rightarrow \mathbf{G} = \frac{1}{\beta}[\mathbf{F} - \mathcal{K}\mathcal{K}^*\mathcal{K}W] - \mathcal{K}W.$$

Finally we get

$$\mathbf{V} = \mathcal{K}W + \mathbf{G} = \frac{1}{\beta}[\mathbf{F} - \mathcal{K}\mathcal{K}^*\mathcal{K}W] = \frac{1}{\beta}[\mathbf{F} - \mathcal{K}(\beta\mathbb{I} + \mathcal{K}^*\mathcal{K})^{-1}\mathcal{K}^*\mathbf{F}].$$

□

Remark 11.3. *It is not a surprise that the seemingly more difficult equation $\nabla(-\nabla\cdot\boldsymbol{p}) + \beta\boldsymbol{p} = \boldsymbol{f}$ can actually be solved by computing $(-\Delta_h + \beta\mathbb{I})^{-1}$, since the Douglas-Rachford splitting on the dual problem is equivalent to ADMM on the primal problem, which involves solving $(-\Delta_h + \beta\mathbb{I})^{-1}$.*

Appendices

Appendix A

Linear algebra

A.1 Eigenvalues and Courant-Fischer-Weyl min-max principle

Notations and quick facts:

- A^T denote the transpose. A^* denote the conjugate transpose of A.
- A matrix $A \in \mathbb{C}^{n\times n}$ is called Hermitian if $A^* = A$. Any Hermitian matrix A has real eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ with a complete set of orthonormal eigenvectors.
- Any real symmetric matrix has real eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ with a complete set of **real** orthonormal eigenvectors.

For a Hermitian matrix A, Rayleigh-Ritz quotient is defined as

$$R_A(x) = \frac{x^*Ax}{x^*x}, \quad x \in \mathbb{C}^n.$$

Theorem A.1 (Courant-Fischer-Weyl min-max principle)**.** *Let λ_1 and λ_n be the largest and the smallest eigenvalues of a Hermitian matrix A, then for any vector $x \in \mathbb{C}^n$,*

$$\lambda_n \leq \frac{x^*Ax}{x^*x} \leq \lambda_1,$$

$$\lambda_n = \min_x \frac{x^*Ax}{x^*x},$$

$$\lambda_1 = \max_x \frac{x^*Ax}{x^*x}.$$

Proof. Let $\{v_j \in \mathbb{C}^n : j = 1, \cdots, n\}$ be orthonormal eigenvectors of A then they form a basis. Thus $x = \sum_{j=1}^{n} a_j v_j$. Let V be a matrix with columns as v_j and a be a column vector with entries a_j. Then $x = Va$ and

$x^*x = a^*V^*Va = a^*a = \sum_{j=1}^{n} |a_j|^2$. Let Λ be a diagonal matrix with diagonal entries λ_j. We have $Av_j = \lambda_j v_j$ thus $Ax = \sum_{j=1}^{n} a_j Av_j = \sum_{j=1}^{n} a_j \lambda_j v_j = V\Lambda a$. Thus $x^*Ax = a^*V^*V\Lambda a = a^*\Lambda a = \sum_{j=1}^{n} \lambda_j |a_j|^2$. The min-max principle holds because

$$\lambda_n \sum_{j=1}^{n} |a_j|^2 \leq \sum_{j=1}^{n} \lambda_j |a_j|^2 \leq \lambda_1 \sum_{j=1}^{n} |a_j|^2.$$

Obviously, the minimum is attained when $x = v_n$, and the maximum is attanned when $x = v_1$. □

A.2 Singular values

For a matrix $A \in \mathbb{C}^{m\times n}$, let A^* denote the conjugate transpose of A. Then A^*A and AA^* are both positive semi-definite (or definite) Hermitian matrices thus have real non-negative eigenvalues, denoted as $\lambda_i(A^*A)$ and $\lambda_i(AA^*)$ ordering by magnitudes.

The matrix A has $l = \min\{m, n\}$ singular values, defined as

$$\sigma_i(A) = \sqrt{\lambda_i(A^*A)} = \sqrt{\lambda_i(AA^*)}.$$

The singular values are defined for any matrix A and are always real non-negative. Eigenvalues are defined for square matrices and are not necessarily real.

A.3 Singular value decomposition

Theorem A.2. *Let $l \leq \min\{m, n\}$. Any matrix $A \in \mathbb{C}^{m\times n}$ of rank k has a decomposition $A = U\Sigma V^*$ (**singular value decomposition (SVD)**) where U of size $m \times l$ and V of size $n \times l$ have orthonormal columns and Σ of size $l \times l$ is diagonal matrix with singular values of A. It also has a compact decomposition $A = U_1\Sigma_1 V_1$ (**compact SVD**) where where U of size $m \times k$ and V of size $n\times k$ have orthonormal columns and Σ_1 of size $k\times k$ is diagonal matrix with nonzero singular values of A.*

Proof. Assume $n \leq m$, we consider the matrix A^*A (if $n > m$, similar procedure for AA^*). The matrix A^*A is positive semi-definite Hermitian thus has non-negative real eigenvalues with a complete set of orthonormal eigenvectors. And A^*A has the same rank as A (why? good excercise to figure it out), thus A^*A has k nonzero eigenvalues. Let D be a $k\times k$ diagonal matrix with all nonzero eigenvalues of A^*A as diagonal entries, and V be a

$n \times n$ matrix with orthonormal eigenvectors as columns. Then

$$V^* A^* A V = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}.$$

Let $V = [V_1\, V_2]$ corresponding to nonzero and zero eigenvalues, then

$$\begin{bmatrix} V_1^* \\ V_2^* \end{bmatrix} A^* A \begin{bmatrix} V_1 & V_2 \end{bmatrix} = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}.$$

By multiplying matrices in the left hand side above, we get

$$V_1^* A^* A V_1 = D, \quad V_2^* A^* A V_2 = 0.$$

Recall $V = [V_1\, V_2]$ has orthonormal columns thus $VV^* = I$, which implies $V_1 V_1^* + V_2 V_2^* = I$.

Next, since V_2 consists of eigenvectors to zero eigenvalue of A^*A, we get $A^* A V_2 = 0$ thus $V_2^* A^* A V_2 = 0$. So we must have $AV_2 = 0$ because it contradicts with $V_2^* A^* A V_2 = 0$ otherwise.

Let $U_1 = A V_1 D^{-\frac{1}{2}}$ where $D^{\frac{1}{2}}$ is defined as taking square root for diagonal entries of D. Then

$$U_1 D^{\frac{1}{2}} V_1^* = A V_1 V_1^* = A(I - V_2 V_2^*) = A - (AV_2)V_2^* = A.$$

The decomposition $A = U_1 D^{\frac{1}{2}} V_1^*$ is exactly the compact SVD. Pick any U_2 of size $n \times (n-k)$ such that $U = [U_1\, U_2]$ is a unitary matrix and define Σ of size $n \times n$ as

$$\Sigma = \begin{bmatrix} D^{\frac{1}{2}} & 0 \\ 0 & 0 \end{bmatrix},$$

then $A = U \Sigma V$ is the full SVD. □

From the proof above, we get the following facts:

- The columns of V (right-singular vectors) are eigenvectors of A^*A.
- The columns of U (left-singular vectors) are eigenvectors of $AA*$.
- A real matrix A has real singular vectors.
- Let u_i and v_i be i-th columns of U and V corresponding i-th singular value $\sigma_i(A)$, then

$$A v_i = \sigma_i u_i, \quad A^* u_i = \sigma_i v_i.$$

- The rank of A is also the number of nonzero singular values of A.

- The compact SVD of A looks like this:

$$A = \boxed{U_1}\Sigma_1\boxed{V_1^*}$$

with

$$\Sigma_1 = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix}.$$

It is a convention to order σ_i in decreasing order: $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$.

- For a Hermitian (or real symmetric) positive semi-definite (PSD) matrix A and its SVD $A = U\Sigma V^*$ we must have $U = V$, thus its SVD $A = U\Sigma U^*$ is also its eigenvalue decomposition. Therefore, singular values are also eigenvalues for PSD matrices.

A.4 Vector norms

For $x = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}^T$:

- *2-norm*: $\|x\| = \sqrt{\sum_{j=1}^{n} |x|_j^2}$.
- *1-norm*: $\|x\|_1 = \sum_{j=1}^{n} |x|_j$.
- ∞*-norm*: $\|x\|_\infty = \max_j |x|_j$.

A.5 Matrix norms

For a rank k matrix $A = (a_{ij})$ of size $m \times n$, assume its SVD is $A = U\Sigma V$ with nonzero singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$. Let $\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 & \sigma_2 & \cdots & \sigma_k \end{bmatrix}^T$. There are many norms of matrices. The following are a few important ones:

- *Spectral norm*: $\|A\|$ is defined as $\|A\| = \max_{x \in \mathbb{C}^n} \frac{\|Ax\|}{\|x\|}$ ($x \in \mathbb{R}^n$ for real matrices) and $\|A\|$ is equal to the largest singular value of A. By Courant-Fischer-Weyl min-max principle Theorem A.1,

$$\frac{\|Ax\|}{\|x\|} = \sqrt{\frac{\|Ax\|^2}{\|x\|^2}} = \sqrt{\frac{x^*A^*Ax}{x^*x}} \leq \sqrt{\lambda_1(A^*A)}.$$

By taking $x = v_1$, the eigenvector of A^*A corresponding to $\lambda_1(A^*A)$, we get $\|A\| = \sqrt{\lambda_1(A^*A)} = \sigma_1$.

- *Frobenius norm*: $\|A\|_F = \sqrt{tr(A^*A)} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$. We have $\|A\|_F = \|\boldsymbol{\sigma}\|$ because

  $$\|A\|_F = \sqrt{tr(V^*\Sigma U^* U\Sigma V)} = \sqrt{tr(V^*\Sigma^2 V)} = \sqrt{tr(VV^*\Sigma^2)} = \sqrt{\sum_j \sigma_j^2},$$

  where we have used the property of trace function $tr(ABC) = tr(CAB)$ for three matrices A, B, C of proper sizes.

- *Nuclear norm*: $\|A\|_* = \sigma_1 + \sigma_2 + \cdots \sigma_k$. Then the nuclear norm of A is simply $\|\boldsymbol{\sigma}\|_1$.

- *Matrix 1-norm*: $\|A\|_1 = \max_{x\in\mathbb{C}^n} \frac{\|Ax\|_1}{\|x\|_1}$ ($x \in \mathbb{R}^n$ for real matrices). Since Ax is a linear combination of columns of A, therefore $\|Ax\|_1$ for $\|x\|_1 = 1$ is less than or equal to a convex combination of 1-norm of columns of A thus $\|A\|_1 = \max_j \sum_{i=1}^{m} |a_{ij}|$.

- *Matrix ∞-norm*: $\|A\|_\infty = \max_{x\in\mathbb{C}^n} \frac{\|Ax\|_\infty}{\|x\|_\infty}$ ($x \in \mathbb{R}^n$ for real matrices). It is easy to show $\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|$.

Useful facts:

- For a matrix norm $|||A|||$ induced by vector norms such as spectral norm, $1-norm$ and ∞-norm, by definition we have

  $$|||Ax||| \le |||A||| \cdot |||x|||.$$

  Since $|||ABx||| \le |||A||| \cdot |||Bx||| \le |||A||| \cdot |||B||| \cdot |||x|||$, we also have

  $$|||AB||| \le |||A||| \cdot |||B|||.$$

- For a matrix norm $|||A|||$ defined through singular values such as spectral norm, Frobenius norm and nuclear norm, it is invariant after unitary transformation: let T and S be unitary matrices, then $|||A||| = |||TAS|||$. Notice that $TAS = (TU)\Sigma(V^*S)$ is the SVD of TAS, so TAS has the same singular values as A.

A.6 Normal matrices

A matrix A is normal if $A^*A = AA^*$. The following are equivalent:

- $A^*A = AA^*$.

- $\sigma_i(A) = |\lambda_i(A)|$.
- A is diagonalizable by unitary matrix: $A = U\Lambda U^*$ where Λ is diagonal. (Obviously, $A = U\Lambda U^*$ is also its eigenvalue decomposition. In other words, A has a complete set of orthonormal eigenvectors (but eigenvalues could be negative, could be complex). If Λ has negative or complex diagonal entries, then $A = U\Lambda U^*$ is not SVD and its SVD has the form $A = U|\Lambda|V^*$ where $|\Lambda|$ is a diagonal matrix with diagonal entries $|\lambda_i|$.)

The equivalency can be easily established by SVD. All Hermitian matrices including PSD matrices are normal. Here is one non-Hermitian normal matrix example: a matrix A is skew-Hermitian if $A^* = -A$. Skew-Hermitian matrices are normal and always have purely imaginary eigenvalues.

Appendix B

Taylor expansion

Lemma B.1 (Second-order Mean Value Theorem)**.** *Suppose that $I \subset \mathbb{R}$ is an open interval and that $f(x)$ is a function of class C^2 ($f''(x)$ exists and is continuous) on I. For $a \in I$ and h such that $a + h \in I$, there exists some $\theta \in (0, 1)$ such that*

$$f(a + h) = f(a) + hf'(a) + \frac{h^2}{2} f''(a + \theta h).$$

Proof. Consider $g_1(x) = f(x) - f(a) - (x - a)f'(a)$ then $g_1(a) = g_1'(a) = 0$. Define

$$g(x) = g_1(x) - \left(\frac{x - a}{h}\right)^2 g_1(a + h),$$

then $g(a) = g'(a) = g(a + h) = 0$. By Mean Value Theorem , we have

$$g(a) = g(a + h) = 0 \Longrightarrow g'(a + \alpha h) = 0,$$

for some $\alpha \in (0, 1)$. Use Mean Value Theorem again on $g'(a) = g'(a+\alpha h) = 0$, we get $g''(a+\theta h) = 0$ for some $\theta \in (0, \alpha)$. Since $g''(x) = f''(x) - \frac{2}{h^2} g_1(a + h)$, $g''(a + \theta h) = 0$ implies that we get the explicit remainder for the second order Taylor expansion as $g_1(a + h) = \frac{h^2}{2} f''(a + \theta h)$. □

Theorem B.1 (Multivariate Quadratic Taylor's Theorem)**.** *Suppose that $S \subset \mathbb{R}^n$ is an open set and that $f : S \longrightarrow \mathbb{R}$ is a function of class C^2 on S. Then for $\mathbf{a} \in S$ and $\mathbf{h} \in \mathbb{R}^n$ such that the line segment connecting $\mathbf{a}$ and $\mathbf{a} + \mathbf{h}$ is contained in S, there exists $\theta \in (0, 1)$ such that*

$$f(\mathbf{a} + \mathbf{h}) = f(\mathbf{a}) + \nabla f(\mathbf{a}) \cdot \mathbf{h} + \frac{1}{2} \mathbf{h}^T \nabla^2 f(\mathbf{a} + \theta \mathbf{h}) \mathbf{h}.$$

Proof. Define $g(t) = f(\mathbf{a} + t\mathbf{h})$. By Lemma B.1 on $g(t)$, there is $\theta \in (0, 1)$ s.t.

$$g(1) = g(0) + g'(0) + \frac{1}{2} g''(\theta).$$

By chain rule, we have $g'(0) = \nabla f(\mathbf{a}) \cdot \mathbf{h}$ and $g''(\theta) = h^T \nabla^2 f(\mathbf{a} + \theta \mathbf{h}) \mathbf{h}$, which complete the proof. □

Appendix C

Convex functions

Consider a function $f : \mathbb{R}^n \to \mathbb{R}$ and any $x, y \in \mathbb{R}^n$ and any $t \in (0, 1)$.

- $f(x)$ is called convex if $f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$.
- $f(x)$ is called strictly convex if $f(tx + (1-t)y) < tf(x) + (1-t)f(y)$.
- $f(x)$ is called strongly convex with a constant parameter $m > 0$ if

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{m}{2}t(1-t)\|x-y\|^2.$$

- $f(x)$ is (strictly or strongly) concave if $-f(x)$ is (strictly or strongly) convex.
- East to verify that $f(x)$ is strongly convex with $m > 0$ if and only if $f(x) - \frac{m}{2}\|x\|^2$ is convex. Strong convexity with $m = 0$ is convexity.
- A convex function by definition satisfies the **Jensen's inequality**:

$$f(a_1x + a_2y) \leq a_1f(x) + a_2f(y), \quad \forall a_1, a_2 \geq 0, a_1 + a_2 = 1.$$

A convex function does not need to be differentiable, e.g., the single variable absolute value function $f(x) = |x|$. If a single variable function is continuously differentiable, then being convex (concave) simply means that the derivative $f'(x)$ is increasing (decreasing), i.e., $[f'(y) - f'(x)](y - x) \geq 0$. If twice continuously differentiable, then convexity simply means $f''(x) \geq 0$ (Hessian matrix $\nabla^2 f(x)$ is positive semi-definite for multivariable case).

Lemma C.1. *Assume $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable. Then the following are equivalent definitions of $f(x)$ being convex:*

- $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle, \quad \forall x, y.$
- $\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq 0, \quad \forall x, y.$

If replacing $\geq$ with $>$ above, then we get equivalent definitions for strict convexity. For strong convexity with parameter $m > 0$, the following are equivalent definitions:

- $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x - y\|^2, \quad \forall x, y.$
- $\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq m\|x - y\|^2, \quad \forall x, y.$

Proof. We only prove the equivalency for strong convexity, since convexity is simply strong convexity with $m = 0$ and discussion for strict convexity is similar to convexity.

First, assume $f(x)$ is strongly convex, then $f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{m}{2}t(1-t)\|x-y\|^2 \Rightarrow \frac{f(tx+(1-t)y)-f(y)}{t} \leq f(x) - f(y) - \frac{m}{2}(1-t)\|x-y\|^2$. Let $g(t) = f(tx + (1-t)y)$ then $g(0) = f(y)$ and $g'(t) = \nabla f(tx + (1-t)y)^T(x-y) = \langle \nabla f(tx + (1-t)y), x - y \rangle$. By the Mean Value Theorem on $g(t)$, there exists $s \in (0, t)$ such that $g'(s) = \frac{g(t)-g(0)}{t}$. So $\frac{f(tx+(1-t)y)-f(y)}{t} = \frac{g(t)-g(0)}{t} = g'(s) = \langle \nabla f(sx + (1-s)y), x - y \rangle$ thus

$$\langle \nabla f(sx + (1-s)y), x - y \rangle \leq f(x) - f(y) - \frac{m}{2}(1-t)\|x-y\|^2.$$

Let $t \to 0$ then $s \to 0$, we get $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x - y\|^2$.

Second, assume $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x-y\|^2$. Then combing with $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{m}{2}\|x - y\|^2$, we get $\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq m\|x - y\|^2$.

Third, assume $\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq m\|x-y\|^2$. Let $x_t = tx + (1-t)y$, then

$$\langle \nabla f(x_t) - \nabla f(y), x_t - y \rangle \geq m\|x_t - y\|^2,$$

thus

$$\langle \nabla f(tx + (1-t)y) - \nabla f(y), t(x - y) \rangle \geq mt^2\|x - y\|^2,$$

and

$$\langle \nabla f(tx + (1-t)y), x - y \rangle \geq \langle \nabla f(y), x - y \rangle + mt\|x - y\|^2.$$

Consider $g(t) = f(tx + (1-t)y)$, then

$$\int_0^1 g'(t)dt = \int_0^1 \langle \nabla f(tx+(1-t)y), x-y \rangle dt \geq \int_0^1 (\langle \nabla f(y), x-y \rangle + mt\|x-y\|^2)dt$$

$$= \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x - y\|^2.$$

So $f(x) - f(y) = g(1) - g(0) \geq \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x - y\|^2$.

Finally, assume $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{m}{2}\|x - y\|^2$. Let $x_t = tx + (1-t)y$, then

$$f(x) \geq f(x_t) + \langle \nabla f(x_t), x - x_t \rangle + \frac{m}{2}\|x - x_t\|^2,$$

$$f(y) \geq f(x_t) + \langle \nabla f(x_t), y - x_t \rangle + \frac{m}{2}\|y - x_t\|^2.$$

Combining two inequalities with coefficients t and $1-t$, we get $f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{m}{2}t(1-t)\|x-y\|^2$. □

Lemma C.2. *Assume $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable (second order partial derivatives exist and are continuous).*

- *$f(x)$ is convex if $\nabla^2 f(x) \geq 0$ (Hessian matrix is positive semi-definite) for all x. This is also a necessary condition for single variable functions.*
- *$f(x)$ is strictly convex if $\nabla^2 f(x) > 0$ for all x. This is not necessary even for single variable functions: $f(x) = x^4$ is strictly convex but $f''(x) > 0$ is not true at $x = 0$.*
- *$f(x)$ is strongly convex if $\nabla^2 f(x) \geq mI$ ($\nabla^2 f(x) - mI$ is positive semi-definite) for all x. This is also a necessary condition for single variable function.*

Proof. Apply Multivariate Quadratic Taylor's Theorem (Theorem B.1), we get

$$f(x) = f(y) + \nabla f(y)^T(x-y) + \frac{1}{2}(y-x)^T\nabla^2 f(y+\theta x)(x-y), \theta \in (0,1).$$

Strong convexity is proven by Lemma C.1 and the fact that

$$\nabla^2 f(y+\theta x) \geq mI \Rightarrow \frac{1}{2}(x-y)^T\nabla^2 f(y+\theta x)(x-y) \geq \frac{m}{2}\|x-y\|^2.$$

Convexity and strict convexity are similarly proven. □

Problem C.1. *In gas dynamics, governing hydrodynamics equations are defined by conservation of mass ρ, momentum $\mathbf{m} = (m_x, m_y, m_z)$ and total energy E. The pressure is defined as $p = (\gamma - 1)(E - \frac{1}{2}\frac{\|\mathbf{m}\|^2}{\rho})$ in equation of state for for ideal gas where $\gamma > 1$ is a constant parameter, e.g., $\gamma = 1.4$ for air. Regard p as a function of conservative variables ρ, m_x, m_y, m_z, E, verify that $p(\rho, \mathbf{m}, E)$ is a concave function for $\rho > 0$ thus satisfies the Jensen's inequity:*

$$p\left(a_1 \begin{bmatrix}\rho\\ \mathbf{m}\\ E\end{bmatrix} + a_2 \begin{bmatrix}\rho\\ \mathbf{m}\\ E\end{bmatrix}\right) \leq a_1 p\left(\begin{bmatrix}\rho\\ \mathbf{m}\\ E\end{bmatrix}\right) + a_2 p\left(\begin{bmatrix}\rho\\ \mathbf{m}\\ E\end{bmatrix}\right), \quad a_1, a_2 > 0, a_1 + a_2 = 1.$$

Hint*: show the Hessian matrix is negative definite. Start with an easier problem by considering 1D case: $p = (\gamma - 1)(E - \frac{1}{2}\frac{m^2}{\rho})$ where m is scalar.*

Appendix D

Sobolev Spaces

D.1 Poincaré inequalities

Theorem D.1. *Assume $\Omega \subset \mathbb{R}^n$ is a bounded open set. Then for any $u \in W_0^{1,p}(\Omega)$ with $1 \leq p < n$, we have*

$$\|u\|_{L^q(\Omega)} \leq C\|\nabla u\|_{L^q(\Omega)},$$

for each $q \in [1, p^*]$ ($p^* := \frac{np}{n-p}$), where the constanc C depends only on p, q, n, Ω. In particular, for all $1 \leq p \leq +\infty$, we have

$$\|u\|_{L^p(\Omega)} \leq C\|\nabla u\|_{L^p(\Omega)} \quad , \forall u \in W_0^{1,p}(\Omega).$$

Also, for $p = 2$, we have

$$\|u\|_{L^2(\Omega)} \leq C\|\nabla u\|_{L^2(\Omega)}, \quad \forall u \in H_0^1(\Omega).$$

Theorem D.2. *Assume $\Omega \subset \mathbb{R}^n$ is a bounded connected open set and its boundary $\partial\Omega$ is C^1. Then for any $u \in W^{1,p}(\Omega)$ with $1 \leq p \leq +\infty$, we have*

$$\|u - \bar{u}\|_{L^p(\Omega)} \leq C\|\nabla u\|_{L^p(\Omega)},$$

where $\bar{u} = \frac{1}{|\Omega|}\int_\Omega u dx$ is the average of u.

References

Bibliography

[1] William L Briggs, Van Emden Henson, and Steve F McCormick. *A multigrid tutorial*. SIAM, 2000.

[2] Lothar Collatz. *The numerical treatment of differential equations*, volume 60. Springer Science & Business Media, 2012.

[3] Lawrence C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998.

[4] Bertil Gustafsson, Heinz-Otto Kreiss, and Joseph Oliger. *Time dependent problems and difference methods*, volume 24. John Wiley & Sons, 1995.

[5] L Kantorovich and V Krylov. Approximate methods of higher analysis. *Bull. Amer. Math. Soc*, 66(3):146–147, 1960.

[6] Heinz-Otto Kreiss and Jens Lorenz. *Initial-boundary value problems and the Navier-Stokes equations*. SIAM, 2004.

[7] Randall J LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[8] Hao Li, Danielö Appelo, and Xiangxiong Zhang. Accuracy of Spectral Element Method for Wave, Parabolic, and Schrödinger Equations. *SIAM Journal on Numerical Analysis*, 60(1):339–363, 2022.

[9] Hao Li and Xiangxiong Zhang. Superconvergence of high order finite difference schemes based on variational formulation for elliptic equations. *Journal of Scientific Computing*, 82(2):36, 2020.

[10] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

[11] Ken Sauer and Charles Bouman. Bayesian estimation of transmission tomograms using segmentation based optimization. *IEEE Transactions on Nuclear Science*, 39(4):1144–1152, 1992.

[12] John C Strikwerda. *Finite difference schemes and partial differential equations*. SIAM, 2004.