

Numerical Methods¹

Alejandro Campos

March 8, 2021

¹Disclaimer: a large chunk of the work in this document is not original; instead, parts of this document are personal notes obtained from the following books: Lomax, Pulliam, Zing; Larsson, Thomee; Hirsch; Leveque; Blazek; Laney; Moin.

Contents

I	Introductory Concepts	3
II	Numerical Solution of ODE's	5
1	List of time integrators	6
1.1	Explicit	6
1.2	Implicit	6
1.3	Predictor-Corrector	6
1.4	Runge-Kutta	7
1.5	Multi-Step	7
2	Solving non-linear equations	8
2.1	Newton's method	8
III	Finite Difference for PDE's	10
3	Finite difference formulas	11
3.1	Fourier analysis	11
4	Elliptic	13
5	Parabolic	14
6	Hyperbolic	16
IV	Finite Volume for PDE's	17
7	Elliptic	18
8	Parabolic	19
9	Hyperbolic	20
9.1	One-dimensional case	20
9.2	Multi-dimensional case	20
9.2.1	An example from turbulence modeling	22
9.3	Implicit time integration	23

V	Finite Element for PDE's	25
10	Elliptic	26
11	Parabolic	27
12	Hyperbolic	28
A	Notes on functional analysis	29
A.1	Classification of methods	29
A.2	Some terminology	29
A.3	Useful equalities and inequalities	30
A.4	The weak derivative	30
A.5	Function spaces	31

Part I

Introductory Concepts

- Analytical solution u satisfies

$$D(u) = Au - f = 0 \quad \text{in } \Omega, \quad (1)$$

where A is the differential operator.

- Discrete numerical solution U_j satisfies

$$D_h(U_j^n) = 0 \quad \text{for all } n, j, \quad (2)$$

where D_h is the discrete operator for D .

- Continuous numerical solution v satisfies

$$D_h v = 0 \quad \text{in } \Omega, \quad (3)$$

Thus, $v(t_n, x_j) = U_j^n$ for all n, j .

- Local truncation error

$$l_j^{n+1} = u(t_{n+1}, x_j) - U_j^{n+1}, \quad (4)$$

given that $U_j^n = u(t_n, x_j)$ for all j .

- Global truncation error

$$e_j^{n+1} = u(t_{n+1}, x_j) - U_j^{n+1}, \quad (5)$$

given that $U_j^0 = u(t_0, x_j)$ for all j .

- Truncation error τ : difference between the discrete and analytical equations when applied to a given continuous function g

$$\tau = D_h g - Dg. \quad (6)$$

The truncation error allows us to compute global truncation errors. Also, consider the truncation error for the analytical solution, that is, $\tau = D_h u - Du$. Using equation (1), one obtains that u is also the solution to $D_h u = \tau$. This means that the **analytical solution satisfies the numerical equation but with an additional source term equal to τ** . Conversely, consider the truncation error for the function v , $\tau = D_h v - Dv$. Using equation (3), one obtains that v is also a solution to $Dv = \tau$. This means that the **numerical solution satisfies the analytical equation but with an additional source term equal to τ** .

- Order of accuracy r defined by $\tau^n = \mathcal{O}(\Delta t^r)$ as $\Delta t \rightarrow 0$ for ODE's and $\tau^n = \mathcal{O}(\Delta x^r)$ as $\Delta x \rightarrow 0$ for PDE's, where Δt is related to Δx .

Part II

Numerical Solution of ODE's

Chapter 1

List of time integrators

Consider the IVP for $u = u(t)$

$$\frac{du}{dt} = f(t, u) \quad \text{in } (0, \infty) \quad (1.1)$$

with initial condition $u(0) = u^0$. We will discretize time into a set of finite values t^n , and express the numerical solution at a given time t^n as U^n .

1.1 Explicit

- Explicit Euler method (Forward Euler method)

$$u^{n+1} = u^n + \Delta t f(t_n, u^n) \quad 1^{\text{st}} \text{ O.A.} \quad (1.2)$$

1.2 Implicit

- Implicit Euler (Backward Euler)

$$u^{n+1} = u^n + \Delta t f(t_{n+1}, u^{n+1}) \quad 1^{\text{st}} \text{ O.A.} \quad (1.3)$$

- Trapezoidal (Crank-Nicholson)

$$u^{n+1} = u^n + \frac{\Delta t}{2} [f(t_n, u^n) + f(t_{n+1}, u^{n+1})] \quad 2^{\text{nd}} \text{ O.A.} \quad (1.4)$$

- Implicit Midpoint

$$u^{n+1} = u^n + \Delta t f\left(t_{n+1/2}, \frac{1}{2}(u^n + u^{n+1})\right) \quad (1.5)$$

1.3 Predictor-Corrector

- Explicit Midpoint (Modified Euler, 2nd order RK)

$$\begin{aligned} u^{n+1/2} &= u^n + \frac{\Delta t}{2} f(t_n, u^n) \\ u^{n+1} &= u^n + \Delta t f(t_{n+1/2}, u^{n+1/2}) \end{aligned} \quad (1.6)$$

- Heun's (Explicit Trapezoidal, Improved Euler)

$$\begin{aligned} \tilde{u}^{n+1} &= u^n + \Delta t f(t_n, u^n) \\ u^{n+1} &= u^n + \frac{\Delta t}{2} [f(t_n, u^n) + f(t_{n+1}, \tilde{u}^{n+1})] \quad 2^{\text{nd}} \text{ O.A.} \end{aligned} \quad (1.7)$$

1.4 Runge-Kutta

- 4th order Runge-Kutta

$$\begin{aligned}k_1 &= f(t_n, u^n) \\k_2 &= f(t_{n+1/2}, u^n + \frac{\Delta t}{2}k_1) \\k_3 &= f(t_{n+1/2}, u^n + \frac{\Delta t}{2}k_2) \\k_4 &= f(t_n, u^n + \Delta tk_3) \\u^{n+1} &= u^n + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4)\end{aligned}\tag{1.8}$$

1.5 Multi-Step

- Leapfrog
- Adams-Bashforth

Chapter 2

Solving non-linear equations

The implicit schemes result in non-linear equations that require some sort of non-linear solver. For this chapter, we'll assume the governing ODE is

$$\frac{du}{dt} = f(u) \quad \text{in } (0, \infty), \quad (2.1)$$

or, for a system of equations,

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}) \quad \text{in } (0, \infty). \quad (2.2)$$

2.1 Newton's method

The Euler scheme requires finding u^{n+1} so that the following is satisfied

$$u^{n+1} = u^n + \Delta t f(u^{n+1}). \quad (2.3)$$

That is, we need to find the root of the nonlinear equation

$$g(x) = x - u^n - \Delta t f(x). \quad (2.4)$$

Newton's method finds the root of a non-linear equation $g(x)$ by iterating through m as follows

$$x^{m+1} = x^m - \frac{g(x^m)}{g'(x^m)}. \quad (2.5)$$

Applying Newton's method to eq. (2.4), we have

$$x^{m+1} = x^m - \frac{x^m - u^n - \Delta t f(x^m)}{1 - \Delta t f'(x^m)}, \quad (2.6)$$

The initial value for the non-linear solver is $x^0 = u^n$. Thus, using the above, u^{n+1} is approximated by x^m as $m \rightarrow \infty$. Note that this is only one example of what is referred to as a fixed-point iteration. There are other fixed-point-iteration methods that can be used to solve eq. (2.4) that are not Newton's method.

As a side note, let's assume u^{n+1} is approximated sufficiently well by x^1 , that is, only one Newton iteration is needed. Then we have

$$u^{n+1} = u^n + \frac{\Delta t f(u^n)}{1 - \Delta t f'(u^n)}, \quad (2.7)$$

which we re-write as

$$\left(\frac{1}{\Delta t} - f'(u^n) \right) \Delta u = f(u^n), \quad (2.8)$$

where $\Delta u = u^{n+1} - u^n$. The above is the approximation that gets used for pseudo-time stepping, so that the Backward Euler scheme can converge to the steady state solution as one iterates through n .

To solve for the root of a system of non-linear equations $\mathbf{g}(\mathbf{x})$, Newton's method is as follows

$$\mathbf{x}^{m+1} = \mathbf{x}^m - \mathbf{J}^{-1}(\mathbf{x}^m) \mathbf{g}(\mathbf{x}^m). \quad (2.9)$$

In the above, $\mathbf{J}^{-1}(\mathbf{x})$ is the inverse of the Jacobian matrix $\mathbf{J}(\mathbf{x})$, which is given by

$$\mathbf{J}(\mathbf{x}) = \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial g_1(\mathbf{x})}{\partial x_1} & \frac{\partial g_1(\mathbf{x})}{\partial x_2} & \cdots \\ \frac{\partial g_2(\mathbf{x})}{\partial x_1} & \frac{\partial g_2(\mathbf{x})}{\partial x_2} & \\ \vdots & & \ddots \end{pmatrix}. \quad (2.10)$$

For the backward Euler scheme, the equivalent of eq. (2.6) would be

$$\mathbf{x}^{m+1} = \mathbf{x}^m - \left(\mathbf{I} - \Delta t \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}^m} \right)^{-1} (\mathbf{x}^m - \mathbf{u}^n - \Delta t \mathbf{f}(\mathbf{x}^m)), \quad (2.11)$$

and the equivalent of eq. (2.8) would be

$$\left(\frac{\mathbf{I}}{\Delta t} - \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{u}^n} \right) \Delta \mathbf{u} = \mathbf{f}(\mathbf{u}^n), \quad (2.12)$$

where $\Delta \mathbf{u} = \mathbf{u}^{n+1} - \mathbf{u}^n$.

Part III

Finite Difference for PDE's

Chapter 3

Finite difference formulas

Forward	Backward	Central	Central (2nd order)
$\partial U_j = \frac{U_{j+1} - U_j}{h}$	$\bar{\partial} U_j = \frac{U_j - U_{j-1}}{h}$	$\hat{\partial} U_j = \frac{U_{j+1} - U_{j-1}}{2h}$	$\partial \bar{\partial} U_j = \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2}$

3.1 Fourier analysis

This analysis is based on a test function that is periodic and thus has the following form

$$u(x) = \sum_{n=-\infty}^{n=\infty} \hat{u}_n e^{i\kappa_n x}. \quad (3.1)$$

We are interested in using the different numerical methods to compute the resulting approximate derivatives of the generic mode $e^{i\kappa x}$. We note that one can then combine the approximate derivatives of all the modes to obtain the approximate derivative of our test function u .

The analytically derivative of our generic mode $e^{i\kappa x}$ is $i\kappa e^{i\kappa x}$. We now assume that the approximate derivatives at a point x_j obtained using numerical schemes will be of the form $i\kappa^* e^{i\kappa x_j}$, where κ^* is a modified wave number. The closer κ^* is to κ the higher the accuracy of the numerical scheme. For example, for the central first order scheme

$$\delta U_j = \frac{U_{j+1} - U_{j-1}}{2h} \quad (3.2)$$

we have

$$i\kappa^* e^{i\kappa x_j} = \frac{e^{i\kappa(x_j+h)} - e^{i\kappa(x_j-h)}}{2h} \quad (3.3)$$

which leads to

$$\kappa^* = \frac{\sin \kappa h}{h}. \quad (3.4)$$

For the sixth-order compact scheme

$$\alpha \delta U_{j-1} + \delta U_j + \alpha \delta U_{j+1} = \frac{a}{2h} (U_{j+1} - U_{j-1}) + \frac{b}{4h} (U_{j+2} - U_{j-2}) \quad (3.5)$$

we have

$$\alpha i\kappa^* e^{i\kappa(x_j-h)} + i\kappa^* e^{i\kappa x_j} + \alpha i\kappa^* e^{i\kappa(x_j+h)} = \frac{a}{2h} [e^{i\kappa(x_j+h)} - e^{i\kappa(x_j-h)}] + \frac{b}{2h} [e^{i\kappa(x_j+2h)} - e^{i\kappa(x_j-2h)}] \quad (3.6)$$

which leads to

$$\kappa^* = \frac{\frac{a}{h} \sin \kappa h + \frac{b}{2h} \sin 2\kappa h}{1 + 2\alpha \cos \kappa h}. \quad (3.7)$$

A spectral method, on the other hand, will explicitly express the derivative of a mode as $i\kappa e^{i\kappa x_j}$, up to the last mode $\kappa = \frac{2\pi}{L} \frac{N}{2}$, and will not be able to capture higher modes. Thus,

$$\kappa^* = \begin{cases} \kappa & \text{for } \frac{2\pi}{L} \left(-\frac{N}{2} + 1\right) \leq \kappa \leq \frac{2\pi}{L} \frac{N}{2} \\ 0 & \text{o.w.} \end{cases} \quad (3.8)$$

Chapter 4

Elliptic

Define the following norm: $|U|_S = \max_{x_j \in S} |U_j|$.

For $Au = -au'' + cu = f$ in $\Omega = (0, 1)$, where $a(x) > 0$ and $c(x) \geq 0$, with boundary conditions $u(0) = U_0$ and $u(1) = U_m$, and $A_h = -a_j \partial \bar{\partial} U_j + c_j U_j$:

- **Lemma 4.2**

$$|U|_{\bar{\Omega}} \leq \max\{|U_0|, |U_M|\} + C|A_h U|_{\Omega}$$

- **Theorem 4.1** The error bound follows,

$$|U - u|_{\Omega} \leq Ch^2 \|u\|_C^4$$

For $Au = -\Delta u = f$ in $\Omega = (0, 1) \times (0, 1)$, with boundary conditions $u = 0$ in Γ , and $A_h = -\Delta_h = -\partial_1 \bar{\partial}_1 U_j - \partial_2 \bar{\partial}_2 U_j$:

- **Lemma 4.4**

$$|U|_{\bar{\Omega}} \leq |U|_{\Gamma} + C|\Delta_h U|_{\Omega}$$

- **Theorem 4.2** The error bound follows,

$$|U - u|_{\Omega} \leq Ch^2 \|u\|_C^4$$

Chapter 5

Parabolic

For $u_t = u_{xx}$ in $\mathbf{R} \times \mathbf{R}_+$, with initial condition $u(\cdot, 0) = v$ in \mathbf{R} .

- Each scheme can be associated with its discrete solution operator E_k , defined in either of the following two ways, where U_j^n is defined only at mesh points, and $u^n(x)$ is the corresponding numerical solution defined over all space,

$$U_j^{n+1} = (E_k U^n)_j = \sum_p a_p U_{j-p}^n$$

$$u^{n+1}(x) = (E_k u^n)(x) = \sum_p a_p u^n(x - x_p)$$

- Repeated application yields $U_j^n = (E_k^n V)_j$ or $u^n(x) = (E_k^n v)(x)$, where V_i and $v(x)$ are the initial conditions, specified at mesh points and over all space, respectively.
- The **stability** of E_k (and hence the associated scheme) is defined by $\|U^n\|_{l_p} = \|E_k^n V\|_{l_p} \leq \|V\|_{l_p}$ and $\|u^n\|_{L_p} = \|E_k^n v\|_{L_p} \leq \|v\|_{L_p}$.
- The **symbol** or characteristic polynomial of E_k is defined as follows,

$$\tilde{E}(\xi) = \sum_p a_p e^{-ip\xi}.$$

If we make use of the following Fourier series and Fourier transform,

$$\hat{V}(\xi) = h \sum_{j=-\infty}^{\infty} V_j e^{-ij\xi}$$

$$\hat{v}(\xi) = \int_{-\infty}^{\infty} v(x) e^{-ix\xi} dx$$

then we obtain $(E_k V)^\sim(\xi) = \tilde{E}(\xi) \hat{V}(\xi)$ and $(E_k v)^\sim(\xi) = \tilde{E}(h\xi) \hat{v}(\xi)$, which for repeated application leads to,

$$(E_k^n V)^\sim(\xi) = \tilde{E}^n(\xi) \hat{V}(\xi)$$

$$(E_k^n v)^\sim(\xi) = \tilde{E}^n(h\xi) \hat{v}(\xi).$$

- Using Parseval's theorem one arrives at the **von Neumann's stability condition**, namely, $|\tilde{E}(\xi)| \leq 1$ for all ξ is sufficient for stability in l_2 and L_2 , and necessary for stability in l_∞ .

- The **order of accuracy** r is defined by $\tau^n = \mathcal{O}(h^r)$ as $h \rightarrow 0$. Since $u^{n+1}(x) = E_k u^n(x) + k\tau^n(x)$, where $u^n(x)$ is now the analytical solution and k is the time step, then the order of accuracy is also obtained from $u^{n+1}(x) - E_k u^n(x) = k\mathcal{O}(h^r)$. This expression for order of accuracy is equivalent to $\tilde{E}(\xi) = e^{-\lambda\xi^2} + \mathcal{O}(|\xi|^{r+2})$, as $\xi \rightarrow 0$.

Chapter 6

Hyperbolic

For $u_t = au_x$ in $\mathbf{R} \times \mathbf{R}_+$, with initial condition $u(\cdot, 0) = v$ in \mathbf{R} :

- As for parabolic equations, von Neumann's condition $|\tilde{E}(\xi)| \leq 1$ for all ξ is a necessary and sufficient condition for stability in the L_2 norm.
- Order of accuracy is defined in the same manner as for the parabolic case, except that now the symbol of E_k is given by $\tilde{E}(\xi) = e^{ia\lambda\xi} + \mathcal{O}(|\xi|^{r+1})$ as $\xi \rightarrow 0$.
- **CFL condition** for stability: a necessary condition for stability is that the domain of dependence of the finite difference scheme at (x, t) contains the domain of dependence of the continuous problem.
- The Friedrichs scheme (first order accurate) follows,

$$(E_k U^n)(x) = 1/2(1 + a\lambda)U^n(x + h) + 1/2(1 - a\lambda)U^n(x - h)$$

- The Lax-Wendroff scheme (second order accurate) follows,

$$(E_k U^n)(x) = 1/2(a^2\lambda^2 + a\lambda)U^n(x + h) + (1 - a^2\lambda^2)U^n(x) + 1/2(a^2\lambda^2 - a\lambda)U^n(x - h)$$

Part IV

Finite Volume for PDE's

Chapter 7

Elliptic

Chapter 8

Parabolic

Chapter 9

Hyperbolic

9.1 One-dimensional case

Consider the hyperbolic equation

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0, \quad (9.1)$$

where $f = f(u)$ is the flux of u . The finite volume method consists on discretizing the spatial domain into volumes, which we will denote by the index i , and which are defined by $x \in [x_{i-1/2}, x_{i+1/2}]$, where $x_{i-1/2}$ and $x_{i+1/2}$ represent the boundaries of the finite volume i .

We now proceed by averaging the equation over each control volume, that is

$$\frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial u}{\partial t} dx + \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial f}{\partial x} dx = 0, \quad (9.2)$$

where $\Delta x_i = x_{i+1/2} - x_{i-1/2}$. Moving the time derivative and using the Divergence theorem, the above becomes

$$\frac{d}{dt} \left(\frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} u dx \right) + \frac{1}{\Delta x_i} (f_{i+1/2} - f_{i-1/2}) = 0, \quad (9.3)$$

where $f_{i\pm 1/2}$ is the flux evaluated at $x_{i\pm 1/2}$. The spatially-discrete numerical solution then satisfies

$$\frac{dU_i}{dt} + \frac{1}{\Delta x_i} (F_{i+1/2} - F_{i-1/2}) = 0, \quad (9.4)$$

where $U_i = U_i(t)$ is the discrete solution at some specific point within the finite volume, and is used to approximate the average of u . $F_{i\pm 1/2}$ are the so-called numerical fluxes.

Godunov Scheme:

$$F_{j+1/2}^n = \begin{cases} \min_{U_j \leq u \leq U_{j+1}} f(u) & \text{for } U_j < U_{j+1} \\ \min_{U_{j+1} \leq u \leq U_j} f(u) & \text{for } U_j > U_{j+1} \end{cases}$$

Lax-Friedrichs:

$$F_{j+1/2}^n = \frac{h}{2k} (U_j - U_{j+1} + \frac{1}{2} (f(U_j) + f(U_{j+1})))$$

9.2 Multi-dimensional case

We follow a similar approach for the multidimensional case. Consider a generic conservation equation

$$\frac{\partial w_i}{\partial t} + \frac{\partial f_{ij}^{(c)}}{\partial x_j} = \frac{\partial f_{ij}^{(v)}}{\partial x_j} + q_i \quad (9.5)$$

where w_i is the vector of conservative variables, $f_{ij}^{(c)}$ the convective flux tensor, $f_{ij}^{(v)}$ the viscous flux tensor, and q_i some heat source.

Averaging the equation over a generic finite volume denoted by the index I , one obtains

$$\frac{d}{dt} \left(\frac{1}{\Omega_I} \int_{\Omega_I} w_i dV \right) + \frac{1}{\Omega_I} \int_{\delta\Omega_I} f_i^{(c)} dS = \frac{1}{\Omega_I} \int_{\delta\Omega_I} f_i^{(v)} dS + \frac{1}{\Omega_I} \int_{\Omega_I} q_i dV, \quad (9.6)$$

where $f_i^{(c)} = f_{ij}^{(c)} n_j$ and $f_i^{(v)} = f_{ij}^{(v)} n_j$ are the vectors of convective and viscous fluxes, respectively. In vector notation, this is written as

$$\frac{d}{dt} \left(\frac{1}{\Omega_I} \int_{\Omega_I} \mathbf{w} dV \right) + \frac{1}{\Omega_I} \int_{\delta\Omega_I} \mathbf{f}^{(c)} dS = \frac{1}{\Omega_I} \int_{\delta\Omega_I} \mathbf{f}^{(v)} dS + \frac{1}{\Omega_I} \int_{\Omega_I} \mathbf{q} dV. \quad (9.7)$$

A specific example is the Navier-Stokes equations, for which we have

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{bmatrix} \quad \mathbf{f}^{(c)} = \begin{bmatrix} \rho(u_j n_j) \\ \rho u(u_j n_j) + p n_1 \\ \rho v(u_j n_j) + p n_2 \\ \rho w(u_j n_j) + p n_3 \\ \rho \left(E + \frac{p}{\rho} \right) (u_j n_j) \end{bmatrix} \quad \mathbf{f}^{(v)} = \begin{bmatrix} 0 \\ \tau_{1j} n_j \\ \tau_{2j} n_j \\ \tau_{3j} n_j \\ u_i \tau_{ij} n_j + \kappa \frac{\partial T}{\partial x_j} n_j \end{bmatrix} \quad \mathbf{q} = \begin{bmatrix} 0 \\ \rho g_1 \\ \rho g_2 \\ \rho g_3 \\ \rho u_i g_i \end{bmatrix}. \quad (9.8)$$

The spatially-discrete numerical solution then satisfies

$$\frac{d\mathbf{W}_I}{dt} + \frac{1}{\Omega_I} \sum_{K \in N(I)} \mathbf{F}_K^{(c)} \Delta S_K = \frac{1}{\Omega_I} \sum_{K \in N(I)} \mathbf{F}_K^{(v)} \Delta S_K + \mathbf{Q}_I \quad (9.9)$$

where $\mathbf{W}_I = \mathbf{W}_I(t)$ is the numerical solution at some specific point within the finite volume, and is used to approximate the average of the vector \mathbf{w} . The faces of the finite volumes are indexed, and the set $N(I)$ consists of the indices of the faces of the finite volume I . The variables $\mathbf{F}_K^{(c)} = \mathbf{F}_K^{(c)}(\mathbf{W}_1, \mathbf{W}_2, \dots)$ and $\mathbf{F}_K^{(v)} = \mathbf{F}_K^{(v)}(\mathbf{W}_1, \mathbf{W}_2, \dots)$ are the numerical fluxes, and ΔS_K the surface area, for face K . The averaged source term $\mathbf{Q}_I = \mathbf{Q}_I(\mathbf{W}_I)$ is approximated as the source vector \mathbf{q} evaluated using \mathbf{W}_I . The above is typically rewritten as

$$\frac{d\mathbf{W}_I}{dt} = -\frac{1}{\Omega_I} \mathbf{R}_I, \quad (9.10)$$

where the residual $\mathbf{R}_I = \mathbf{R}_I(\mathbf{W}_1, \mathbf{W}_2, \dots)$ is given by

$$\mathbf{R}_I = \sum_{K \in N(I)} \mathbf{F}_K^{(c)} \Delta S_K - \sum_{K \in N(I)} \mathbf{F}_K^{(v)} \Delta S_K - \mathbf{Q}_I \Omega_I \quad (9.11)$$

The Jacobian is then

$$\frac{\partial \mathbf{R}_I}{\partial \mathbf{W}_J} = \sum_{K \in N(I)} \frac{\partial \mathbf{F}_K^{(c)}}{\partial \mathbf{W}_J} \Delta S_K - \sum_{K \in N(I)} \frac{\partial \mathbf{F}_K^{(v)}}{\partial \mathbf{W}_J} \Delta S_K - \frac{\partial \mathbf{Q}_I}{\partial \mathbf{W}_J} \Omega_I \quad (9.12)$$

Note the the Jacobian is only non zero when $J \in M(I)$, where $M(I)$ represents the set of all finite volumes that the numerical fluxes of finite volume I depend on.

9.2.1 An example from turbulence modeling

Consider the SST turbulence model. The two transport equations solved by the model are

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho k u_j}{\partial x_j} = P - \beta^* \rho w k + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} \right] \quad (9.13)$$

$$\frac{\partial \rho w}{\partial t} + \frac{\partial \rho w u_j}{\partial x_j} = \frac{\gamma}{\nu_t} P - \beta \rho w^2 + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_w \mu_t) \frac{\partial w}{\partial x_j} \right] + 2(1 - F_1) \frac{\rho \sigma_{w2}}{w} \frac{\partial k}{\partial x_j} \frac{\partial w}{\partial x_j} \quad (9.14)$$

The vector of conservative variables is $\mathbf{w} = [\rho k, \rho w]^T$. The convective and viscous flux vectors are

$$\mathbf{f}^{(c)} = \begin{bmatrix} \rho k (u_j n_j) \\ \rho w (u_j n_j) \end{bmatrix} \quad \mathbf{f}^{(v)} = \begin{bmatrix} (\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} n_j \\ (\mu + \sigma_w \mu_t) \frac{\partial w}{\partial x_j} n_j \end{bmatrix}. \quad (9.15)$$

The source \mathbf{q} is

$$\mathbf{q} = \begin{bmatrix} P - \beta^* \rho w k \\ \frac{\gamma}{\nu_t} P - \beta \rho w^2 + 2(1 - F_1) \frac{\rho \sigma_{w2}}{w} \frac{\partial k}{\partial x_j} \frac{\partial w}{\partial x_j} \end{bmatrix} \quad (9.16)$$

Convective numerical fluxes: A simple upwinding scheme is as follows

$$\mathbf{F}_K^{(c)} = \begin{bmatrix} (\rho k)_l a_0 + (\rho k)_r a_1 \\ (\rho w)_l a_0 + (\rho w)_r a_1 \end{bmatrix} \quad (9.17)$$

where subscripts l and r denote values at the center nodes of the control volumes to the left and right of the K^{th} control surface, respectively. The variables a_0 and a_1 are given by

$$a_0 = \begin{cases} q_{lr} & \text{if } q_{lr} > 0 \\ 0 & \text{o.w.} \end{cases} \quad a_1 = \begin{cases} 0 & \text{if } q_{lr} > 0 \\ q_{lr} & \text{o.w.} \end{cases} \quad (9.18)$$

where

$$q_{lr} = \frac{(\mathbf{u}_l \cdot \mathbf{n}) + (\mathbf{u}_r \cdot \mathbf{n})}{2}. \quad (9.19)$$

The corresponding Jacobian is

$$\frac{\partial \mathbf{F}_K^{(c)}}{\partial \mathbf{W}_J} = \begin{bmatrix} \frac{\partial \mathbf{F}_K^{(c)}(1)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{F}_K^{(c)}(1)}{\partial \mathbf{W}_J(2)} \\ \frac{\partial \mathbf{F}_K^{(c)}(2)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{F}_K^{(c)}(2)}{\partial \mathbf{W}_J(2)} \end{bmatrix} = \begin{bmatrix} a_0 \delta_{lJ} + a_1 \delta_{rJ} & 0 \\ 0 & a_0 \delta_{lJ} + a_1 \delta_{rJ} \end{bmatrix}. \quad (9.20)$$

Viscous numerical fluxes: the viscous fluxes can be discretized as follows

$$\mathbf{F}_K^{(v)} = \begin{bmatrix} (\mu + \sigma_k \mu_t)_{avg} \left[\frac{k_r - k_l}{l_{lr}} \mathbf{t}_{lr} + \nabla k_{avg} - (\nabla k_{avg} \cdot \mathbf{t}_{lr}) \mathbf{t}_{lr} \right] \cdot \mathbf{n} \\ (\mu + \sigma_w \mu_t)_{avg} \left[\frac{\omega_r - \omega_l}{l_{lr}} \mathbf{t}_{lr} + \nabla \omega_{avg} - (\nabla \omega_{avg} \cdot \mathbf{t}_{lr}) \mathbf{t}_{lr} \right] \cdot \mathbf{n} \end{bmatrix}. \quad (9.21)$$

where the subscript $_{avg}$ means the left and right values have been averaged, \mathbf{t}_{lr} is the unit vector that points from the center of the l^{th} control volume (the one to the left of the face) to the center of the r^{th} control volume (the one to the right of the face), and l_{lr} is the length between these two centers. The corresponding Jacobian is approximated as follows

$$\frac{\partial \mathbf{F}_K^{(v)}}{\partial \mathbf{W}_J} = \begin{bmatrix} \frac{\partial \mathbf{F}_K^{(v)}(1)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{F}_K^{(v)}(1)}{\partial \mathbf{W}_J(2)} \\ \frac{\partial \mathbf{F}_K^{(v)}(2)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{F}_K^{(v)}(2)}{\partial \mathbf{W}_J(2)} \end{bmatrix}, \quad (9.22)$$

where

$$\frac{\partial \mathbf{F}_K^{(v)}(1)}{\partial \mathbf{W}_J(1)} = (\mu + \sigma_k \mu_t)_{avg} \frac{1}{l_{lr}} \mathbf{t}_{lr} \cdot \mathbf{n} \frac{\delta_{rJ}}{\rho_r} - (\mu + \sigma_k \mu_t)_{avg} \frac{1}{l_{lr}} \mathbf{t}_{lr} \cdot \mathbf{n} \frac{\delta_{lJ}}{\rho_l}, \quad (9.23)$$

$$\frac{\partial \mathbf{F}_K^{(v)}(2)}{\partial \mathbf{W}_J(2)} = (\mu + \sigma_\omega \mu_t)_{avg} \frac{1}{l_{lr}} \mathbf{t}_{lr} \cdot \mathbf{n} \frac{\delta_{rJ}}{\rho_r} - (\mu + \sigma_\omega \mu_t)_{avg} \frac{1}{l_{lr}} \mathbf{t}_{lr} \cdot \mathbf{n} \frac{\delta_{lJ}}{\rho_l} \quad (9.24)$$

and $\frac{\partial \mathbf{F}_K^{(v)}(1)}{\partial \mathbf{W}_J(2)} = \frac{\partial \mathbf{F}_K^{(v)}(2)}{\partial \mathbf{W}_J(1)} = 0$.

Sources: The discretized source terms are computed following eq. (9.16). The Jacobian is expressed as

$$\frac{\partial \mathbf{Q}_I}{\partial \mathbf{W}_J} = \begin{bmatrix} \frac{\partial \mathbf{Q}_I(1)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{Q}_I(1)}{\partial \mathbf{W}_J(2)} \\ \frac{\partial \mathbf{Q}_I(2)}{\partial \mathbf{W}_J(1)} & \frac{\partial \mathbf{Q}_I(2)}{\partial \mathbf{W}_J(2)} \end{bmatrix}. \quad (9.25)$$

According to Wilcox p. 413, the elements of the matrix above are approximated as follows

$$\frac{\partial \mathbf{Q}_I(1)}{\partial \mathbf{W}_J(1)} \approx -\beta^* w_I \delta_{IJ} \quad \frac{\partial \mathbf{Q}_I(2)}{\partial \mathbf{W}_J(2)} \approx -2\beta w_I \delta_{IJ} \quad \frac{\partial \mathbf{Q}_I(1)}{\partial \mathbf{W}_J(2)} = \frac{\partial \mathbf{Q}_I(2)}{\partial \mathbf{W}_J(1)} \approx 0. \quad (9.26)$$

9.3 Implicit time integration

We combine eq. (9.10) for all I into a single equation as follows

$$\frac{d}{dt} \begin{pmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} -\frac{1}{\Omega_1} \mathbf{R}_1 \\ -\frac{1}{\Omega_2} \mathbf{R}_2 \\ \vdots \end{pmatrix}. \quad (9.27)$$

Note that the above is of the same form as eq. (2.2). Thus, the corresponding form of eq. (2.12) would be

$$\begin{aligned} \left[\frac{1}{\Delta t} \begin{pmatrix} \mathbf{I} & 0 & \dots \\ 0 & \mathbf{I} & \\ \vdots & & \ddots \end{pmatrix} - \begin{pmatrix} -\frac{1}{\Omega_1} \frac{\partial \mathbf{R}_1(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_1} & -\frac{1}{\Omega_1} \frac{\partial \mathbf{R}_1(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_2} & \dots \\ -\frac{1}{\Omega_2} \frac{\partial \mathbf{R}_2(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_1} & -\frac{1}{\Omega_2} \frac{\partial \mathbf{R}_2(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_2} & \\ \vdots & & \ddots \end{pmatrix}_{\mathbf{x}_\alpha = \mathbf{W}_\alpha^n} \right] \begin{pmatrix} \Delta \mathbf{W}_1 \\ \Delta \mathbf{W}_2 \\ \vdots \end{pmatrix} \\ = \begin{pmatrix} -\frac{1}{\Omega_1} \mathbf{R}_1(\mathbf{W}_1^n, \mathbf{W}_2^n, \dots) \\ -\frac{1}{\Omega_2} \mathbf{R}_2(\mathbf{W}_1^n, \mathbf{W}_2^n, \dots) \\ \vdots \end{pmatrix}. \quad (9.28) \end{aligned}$$

Multiplying each of the major rows above one obtains

$$\begin{aligned} \left[\frac{1}{\Delta t} \begin{pmatrix} \Omega_1 \mathbf{I} & 0 & \dots \\ 0 & \Omega_2 \mathbf{I} & \\ \vdots & & \ddots \end{pmatrix} + \begin{pmatrix} \frac{\partial \mathbf{R}_1(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{R}_1(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_2} & \dots \\ \frac{\partial \mathbf{R}_2(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{R}_2(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_2} & \\ \vdots & & \ddots \end{pmatrix}_{\mathbf{x}_\alpha = \mathbf{W}_\alpha^n} \right] \begin{pmatrix} \Delta \mathbf{W}_1 \\ \Delta \mathbf{W}_2 \\ \vdots \end{pmatrix} \\ = \begin{pmatrix} -\mathbf{R}_1(\mathbf{W}_1^n, \mathbf{W}_2^n, \dots) \\ -\mathbf{R}_2(\mathbf{W}_1^n, \mathbf{W}_2^n, \dots) \\ \vdots \end{pmatrix}. \quad (9.29) \end{aligned}$$

As previously noted, the only Jacobians $\partial \mathbf{R}_I / \partial \mathbf{W}_J$ that are non-zero are those for which $J \in M(I)$. Thus, the second submatrix of the above is mostly sparse. The I th major row can be written as

$$\sum_{J=M(I)} \left[\frac{\Omega_I}{\Delta t} \delta_{IJ} + \frac{\partial \mathbf{R}_I(\mathbf{x}_1, \mathbf{x}_2, \dots)}{\partial \mathbf{x}_J} \bigg|_{\mathbf{x}_\alpha = \mathbf{W}_\alpha^n} \right] \Delta \mathbf{W}_J = -\mathbf{R}_I(\mathbf{W}_1^n, \mathbf{W}_2^n, \dots). \quad (9.30)$$

Part V

Finite Element for PDE's

Chapter 10

Elliptic

- Consider the boundary value problem in which a solution u is sought such that

$$\begin{aligned} Au &= f & \text{in } \Omega = (0, 1) \\ u &= 0 & \text{on } \Gamma = 0, 1 \end{aligned} \quad (10.1)$$

where $Au = -(au')' + cu$.

- The weak formulation of the above is to find the weak solution $u \in H_0^1$ such that

$$a(u, v) = (f, v) \quad \forall v \in H_0^1, \quad (10.2)$$

where $a(u, v) = \int_{\Omega} au'v' + cuv \, dx$, and $(f, v) = \int_{\Omega} fv \, dx$.

- Partition the domain into M grid elements, and introduce $S_h \subset H_0^1$ as the set of functions that are linear within each grid element. We now aim to find $u_h \in S_h$ such that

$$a(u_h, v_h) = (f, v_h) \quad \forall v_h \in S_h. \quad (10.3)$$

- $\{\Phi_i\}_{i=1}^{M-1}$ is the basis for S_h , where $M - 1$ is the total number of interior points. That is, any $v_h \in S_h$ can be written as $v_h = \sum_{i=1}^{M-1} V_i \Phi_i$. Equation (10.3) is then automatically satisfied if the following is satisfied

$$a(u_h, \Phi_i) = (f, \Phi_i) \quad \text{for } i = 1, \dots, M - 1. \quad (10.4)$$

- Since $u_h \in S_h$, we can write $u_h(x) = \sum_{j=1}^{M-1} U_j \Phi_j(x)$. Thus, the above becomes

$$\sum_{j=1}^{M-1} U_j a(\Phi_j, \Phi_i) = (f, \Phi_i) \quad \text{for } i = 1, \dots, M - 1. \quad (10.5)$$

- Define the vectors \mathbf{U} and \mathbf{b} , as those whose components are U_j and (f, Φ_i) , respectively. Define the matrix \mathbf{A} as that whose components are $a(\Phi_j, \Phi_i)$, where i denotes the row and j the column. Then, the equation above can be written as

$$\mathbf{A}\mathbf{U} = \mathbf{b}. \quad (10.6)$$

Chapter 11

Parabolic

Chapter 12

Hyperbolic

Appendix A

Notes on functional analysis

A.1 Classification of methods

- Finite element method: local trial functions
- Spectral method: global trial functions
- Galerkin method: trial functions equal to test functions
- Collocation method: trial functions not equal to test functions (test functions are delta functions)

A.2 Some terminology

- Cauchy sequence: a sequence v_1, v_2, v_3, \dots is a Cauchy sequence if for every positive real number ϵ , there is a positive integer N such that for all positive integers $m, n > N$, $\|v_m - v_n\| < \epsilon$.
- Complete inner product space: an inner product space V is complete if every Cauchy sequence $\{v_i\}_{i=1}^{\infty}$ in V has a limit $v = \lim v_i \in V$.
- Compact set: a set is compact if it is bounded and closed.
- Coercive bilinear form: a bilinear form $a(\cdot, \cdot)$ is coercive in a Hilbert space V if

$$a(v, v) \geq \alpha \|v\|_V^2, \quad \forall v \in V, \quad \text{with } \alpha > 0. \quad (\text{A.1})$$

- Bounded linear form: a linear form is bounded in the normed vector space V if there exists an $M > 0$ such that $|L(v)| \leq M \|v\|$, for every $v \in V$.
- Bounded bilinear form: a bilinear form is bounded in the normed vector space V if there exists an $M > 0$ such that $|a(w, v)| \leq M \|w\| \|v\|$, for every $w, v \in V$.

A.3 Useful equalities and inequalities

$$|ab| = |a||b| \quad \forall a, b \in \mathbb{C} \quad (\text{A.2})$$

$$|a + b| \leq |a| + |b| \quad \forall a, b \in \mathbb{C} \quad (\text{A.3})$$

$$|(w, v)| \leq \|w\| \|v\| \quad \forall v, w \in \text{Inner product space (Cauchy-Schwarz inequality)} \quad (\text{A.4})$$

$$\|w + v\| \leq \|w\| + \|v\| \quad \forall v, w \in \text{Normed space (triangle inequality)} \quad (\text{A.5})$$

$$\left| \int_a^b v(x) dx \right| \leq \int_a^b |v(x)| dx \quad (\text{A.6})$$

$$\left\| \int_a^b v(x, y) dx \right\| \leq \int_a^b \|v(x, y)\| dx \text{ where the norm is over the } y\text{-domain.} \quad (\text{A.7})$$

A.4 The weak derivative

If $v \in C^1(\bar{\Omega})$, then through integration by parts

$$\int_{\Omega} \frac{\partial v}{\partial x_i} \phi dx = - \int_{\Omega} v \frac{\partial \phi}{\partial x_i} dx \quad \forall \phi \in C_0^1(\Omega). \quad (\text{A.8})$$

However, if $v \in L_2(\Omega)$ but not necessarily in $C^1(\bar{\Omega})$, we cannot write the equation above. Instead, we ask, is there a function w such that the following holds?

$$\int_{\Omega} w \phi dx = - \int_{\Omega} v \frac{\partial \phi}{\partial x_i} dx \quad \forall \phi \in C_0^1(\Omega). \quad (\text{A.9})$$

This can be rewritten as $(w, \phi) = L(\phi)$, where $L(\phi) = - \int_{\Omega} v \frac{\partial \phi}{\partial x_i} dx$. If $L(\phi)$ is bounded in L_2 , Riesz' representation theorem then states a unique solution $w \in L_2(\Omega)$ exists. This w is the weak derivative.

More generally, if $v \in C^k(\bar{\Omega})$, then through integration by parts

$$\int_{\Omega} D^{\alpha} v \phi dx = (-1)^{|\alpha|} \int_{\Omega} v D^{\alpha} \phi dx \quad \forall |\alpha| \leq k, \forall \phi \in C_0^{|\alpha|}(\Omega). \quad (\text{A.10})$$

However, if $v \in L_2(\Omega)$ but not necessarily in $C^k(\bar{\Omega})$, we cannot write the equation above. Instead, we ask, is there a function w such that the following holds?

$$\int_{\Omega} w \phi dx = (-1)^{|\alpha|} \int_{\Omega} v D^{\alpha} \phi dx \quad \forall \phi \in C_0^{|\alpha|}(\Omega). \quad (\text{A.11})$$

As before, if the left-hand-side operator is bounded, then we have a unique solution $w \in L_2(\Omega)$. This w is the weak derivative. Often, weak derivatives are referred to as “ $D^{\alpha}v$ in the weak sense.”

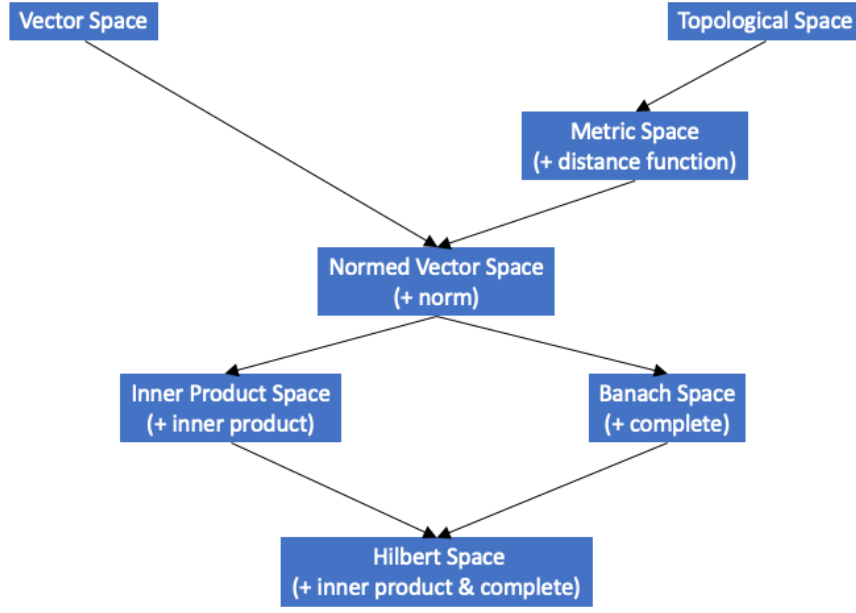


Figure A.1: Map of function spaces.

A.5 Function spaces

Space	Norm	Inner product
$C(M)$	$\ v\ _C = \sup_{x \in M} v(x) $	X
$C^k(M)$	$\ v\ _{C^k} = \max_{ \alpha \leq k} \ D^\alpha v\ _C$ $ v _{C^k} = \max_{ \alpha =k} \ D^\alpha v\ _C$	X
$L_p(\Omega)$	$\ v\ _{L_p} = \left(\int_{\Omega} v ^p dx \right)^{1/p}$	X
$L_2(\Omega)$	$\ v\ _{L_2} = \left(\int_{\Omega} v ^2 dx \right)^{1/2}$	$(v, w) = \int_{\Omega} v w^* dx$
$H^k(\Omega)$	$\ v\ _k = \left(\sum_{ \alpha \leq k} \ D^\alpha v\ ^2 \right)^{1/2}$ $ v _k = \left(\sum_{ \alpha =k} \ D^\alpha v\ ^2 \right)^{1/2}$	$(v, w)_k = \sum_{ \alpha \leq k} (D^\alpha v, D^\alpha w)$

- Let Ω be a bounded domain in \mathbb{R}^d with smooth or polygonal boundary. Then part of the Sobolev embedding theorem can be written as

$$H^k(\Omega) \subset C^l(\bar{\Omega}) \text{ if } k > l + d/2. \quad (\text{A.12})$$

Thus, we have $H^m(\Omega) \subset C^{m-1}(\bar{\Omega})$ for $\Omega \in \mathbb{R}$ and $H^m(\Omega) \subset C^{m-2}(\bar{\Omega})$ for $\Omega \in \mathbb{R}^2$ or \mathbb{R}^3 .