

A Textbook for a First Course in Computational Fluid Dynamics

D. W. Zingg

Institute for Aerospace Studies

University of Toronto

Toronto, Ontario, Canada

T. H. Pulliam

NASA Ames Research Center

Moffett Field, CA 94035, USA

September 9, 1999

Abstract

This paper describes and discusses the textbook, *Fundamentals of Computational Fluid Dynamics* by Lomax, Pulliam, and Zingg, which is intended for a graduate-level first course in computational fluid dynamics. This textbook emphasizes fundamental concepts in developing, analyzing, and understanding numerical methods for the partial differential equations governing the physics of fluid flow. Its underlying philosophy is that the theory of linear algebra and the attendant eigenanalysis of linear systems provides a mathematical framework to describe and unify most numerical methods in common use in the field of fluid dynamics. Two linear model equations, the linear convection and diffusion equations, are used to illustrate concepts throughout. Emphasis is on the semi-discrete approach, in which the governing partial differential equations (PDE's) are reduced to systems of ordinary differential equations (ODE's) through a discretization of the spatial derivatives. The ordinary differential equations are then reduced to ordinary difference equations (O Δ E's) using a time-marching method. This methodology, using the progression from PDE through ODE's to O Δ E's, together with the use of the eigensystems of tridiagonal matrices and the theory of O Δ E's, gives the book its distinctiveness and provides a sound basis for a deep understanding of fundamental concepts in computational fluid dynamics.

distinctive. Topics covered, in this paper, include the model equations, matrix difference operators, Taylor Tables, the semi-discrete approach, converting time-marching methods to ordinary difference equations ($O\Delta E$'s), solution of linear $O\Delta E$'s, the λ - σ relation, and numerical stability concepts. Although this omits a significant portion of the material covered in the book, it is sufficient to illustrate the basic framework. Most of the material presented is excerpted directly from the book.

2 The Model Equations

Many concepts are presented in the context of two one-dimensional model equations, the linear convection equation and the diffusion equation. These two equations are commonly used in understanding CFD algorithms, primarily because they are linear, scalar, and they represent phenomena of importance to the analysis of certain aspects of fluid dynamics. The linear convection equation can be written as:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad (1)$$

where $u(x, t)$ is a scalar quantity propagating with constant speed a . Emphasis is on periodic boundary conditions, which permit the study of many of the basic properties of numerical methods applied to problems involving convection without consideration of boundaries. The diffusion equation is:

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2} \quad (2)$$

where ν is a positive real constant. Consideration is given to both Dirichlet (specified u) and Neumann (specified $\partial u / \partial x$) boundary conditions.

3 Finite-Difference Operators

Finite-difference operators are presented initially through linear combinations of Taylor series expansions, producing, for example, the three-point centered difference operator for a second-derivative:

$$(\delta_{xx}u)_j = \frac{1}{\Delta x^2}(u_{j+1} - 2u_j + u_{j-1}) \quad (3)$$

The idea of a local order of accuracy is also defined based on the leading term in the Taylor series expansion* of the error.

*The discrete Taylor series expansion is defined as $u_{j+k} = u_j + (k\Delta x)\left(\frac{\partial u}{\partial x}\right)_j + \frac{1}{2}(k\Delta x)^2\left(\frac{\partial^2 u}{\partial x^2}\right)_j + \dots + \frac{1}{n!}(k\Delta x)^n\left(\frac{\partial^n u}{\partial x^n}\right)_j + \dots$ for a general increment k

notation is used:

$$B_p(M : a, b, c) = \begin{bmatrix} b & c & & a \\ a & b & c & \\ & & \ddots & \\ & & & a & b & c \\ c & & & a & b \end{bmatrix} \begin{matrix} 1 \\ \\ \\ \vdots \\ M \end{matrix} \quad (8)$$

With this notation, the matrix difference operator for a first derivative with periodic boundary conditions on an M -point grid can be written as

$$(\delta_x)_p = \frac{1}{2\Delta x} B_p(M : -1, 0, 1) \quad (9)$$

Notice that there is no boundary condition vector (\vec{bc}) since this information is interior to the matrix itself.

3.2 Taylor Tables

A Taylor table is a simple and convenient way of forming linear combinations of Taylor series on a term by term basis. This enables one to derive finite-difference expressions for specified derivatives with a specified stencil. An example is given below. The table is constructed so that some of the algebra is simplified. At the top of the table we see an expression with a question mark. This represents one of the questions that a study of this table can answer; namely, what is the local error caused by the use of this approximation? Notice that all of the terms in the equation appear in a column at the left of the table (although, in this case, Δx^2 has been multiplied into each term which simplifies the terms to be put into the table). Then notice that at the head of each column there appears the common factor that occurs in the expansion of each term about the point j , that is,

$$\Delta x^k \cdot \left(\frac{\partial^k u}{\partial x^k} \right)_j \quad k = 0, 1, 2, \dots$$

The columns to the right of the leftmost one, under the headings, make up the Taylor table. Each entry is the coefficient of the term at the top of the corresponding column in the Taylor series expansion of the term to the left of the corresponding row. For example, the last row in the table corresponds to the Taylor series expansion of $c u_{j+1}$:

$$\begin{aligned} c u_{j+1} = & c u_j + c \cdot (1) \cdot \frac{1}{1} \Delta x \cdot \left(\frac{\partial u}{\partial x} \right)_j + c \cdot (1)^2 \cdot \frac{1}{2} \Delta x^2 \cdot \left(\frac{\partial^2 u}{\partial x^2} \right)_j \\ & + c \cdot (1)^3 \cdot \frac{1}{6} \Delta x^3 \cdot \left(\frac{\partial^3 u}{\partial x^3} \right)_j + c \cdot (1)^4 \cdot \frac{1}{24} \Delta x^4 \cdot \left(\frac{\partial^4 u}{\partial x^4} \right)_j + \dots \quad (10) \end{aligned}$$

4 The Semi-Discrete Approach

One strategy for obtaining finite-difference approximations to a partial differential equation (PDE) is to start by differencing the space derivatives only, thereby converting the PDE to a set of coupled ordinary differential equations (ODE's). We refer to this as the *semi-discrete* approach. For some methods it is not possible to separate the spatial and temporal discretization, and two of these, the Lax-Wendroff and MacCormack methods, are discussed later in the book. However, we primarily concentrate on the semi-discrete approach, reducing the PDE's to ODE's by discretizing the spatial terms and using the well developed theory of ODE solutions to aid us in the development of an analysis of accuracy and stability.

For the model PDE's, we can approximate the space derivatives with difference operators and express the resulting ODE's with a matrix formulation. This is a natural formulation when the ODE's are linear. For example, using the 3-point central-differencing scheme to represent the second derivative in the diffusion equation leads to the following ODE diffusion model

$$\frac{d\vec{u}}{dt} = \frac{\nu}{\Delta x^2} B(1, -2, 1)\vec{u} + (\vec{bc}) \quad (13)$$

with Dirichlet boundary conditions folded into the (\vec{bc}) vector.

The generic matrix form of a semi-discrete approximation is expressed by the equation

$$\frac{d\vec{u}}{dt} = A\vec{u} - \vec{f}(t) \quad (14)$$

Note that the elements in the matrix A depend upon both the PDE and the type of differencing scheme chosen for the space terms. The vector $\vec{f}(t)$ is usually determined by the boundary conditions and possibly source terms. In general, even the Euler and Navier-Stokes equations can be expressed in the form of Eq. 14. In such cases the equations are nonlinear, that is, the elements of A depend on the solution \vec{u} and are usually derived by finding the Jacobian of a flux vector. Although the equations are nonlinear, linear analysis leads to diagnostics that are surprisingly accurate when evaluating many aspects of numerical methods as they apply to the Euler and Navier-Stokes equations.

When the $M \times M$ matrix A is independent of \vec{u} and t and has a complete set of linearly independent eigenvectors, the generic system can be written as a set of independent first-order equations in the form:

$$w_1' = \lambda_1 w_1 - g_1(t)$$

The numerical solution to a set of linear ODE's (in which A is not a function of t) is entirely equivalent to the solution obtained if the equations are decoupled, solved in uncoupled form, and recoupled. The importance of this concept resides in its message that we can analyze time-marching methods by applying them to a single, uncoupled equation and our conclusions will apply in general. This is helpful both in analyzing the accuracy of numerical time-march methods and in studying their numerical stability.

As a single representative ODE we choose:

$$\frac{du}{dt} = \lambda u + ae^{\mu t} \quad (21)$$

where λ , μ , and a are complex constants, which can be used to evaluate all manner of time-marching methods. In such evaluations the parameters λ and μ must be allowed to take the worst possible combination of values that might occur in the ODE eigensystem. The exact solution of the representative ODE is (for $\mu \neq \lambda$):

$$u(t) = ce^{\lambda t} + \frac{ae^{\mu t}}{\mu - \lambda} \quad (22)$$

Although it is quite simple, the numerical analysis of Eq. 21 displays many of the fundamental properties and issues involved in the construction and study of most popular time-marching methods.

5 Time-Marching Methods for ODE's

5.1 Converting Time-Marching Methods to O Δ E's

As we have seen, application of a spatial discretization to a PDE produces a coupled system of ODE's, which can be solved numerically using a time-marching method. Application of a time-marching method to an ODE produces an ordinary *difference* equation (O Δ E). In this section, we shall analyze and contrast explicit and implicit schemes to illustrate these methods. For example, consider the explicit Euler time-marching method, which can be written as:

$$u_{n+1} = u_n + hu'_n \quad (23)$$

where

$$\frac{du}{dt} = u' = F(u, t) \quad (24)$$

5.3 The λ - σ Relation

Next consider the application of the explicit Euler method to the generic form of the ODE's obtained using the semi-discrete approach, Eq. 14, with \vec{f} independent of time. The exact solution is

$$u(t) = c_1 (e^{\lambda_1 h})^n \vec{x}_1 + \dots + c_m (e^{\lambda_m h})^n \vec{x}_m + \dots + c_M (e^{\lambda_M h})^n \vec{x}_M + A^{-1} \vec{f} \quad (33)$$

where $t = nh$, while the explicit Euler method produces the solution

$$u_n = c_1 (\sigma_1)^n \vec{x}_1 + \dots + c_m (\sigma_m)^n \vec{x}_m + \dots + c_M (\sigma_M)^n \vec{x}_M + P.S. \quad (34)$$

where the c_m and the \vec{x}_m in the two equations are identical, $\sigma_m = (1 + \lambda_m h)$, and *P.S.* denotes the particular solution. Comparing Eqs. 33 and 34, we see a correspondence between σ_m and $e^{\lambda_m h}$. Since the value of $e^{\lambda h}$ can be expressed in terms of the series

$$e^{\lambda h} = 1 + \lambda h + \frac{1}{2} \lambda^2 h^2 + \frac{1}{6} \lambda^3 h^3 + \dots + \frac{1}{n!} \lambda^n h^n + \dots \quad (35)$$

the truncated expansion $\sigma = 1 + \lambda h$ is an approximation to $e^{\lambda h}$ valid for small λh .

In general, application of the same time-marching method to all of the equations in a coupled system of linear ODE's in the form of Eq. 14, always produces one σ -root for every λ eigenvalue that satisfies the relation

$$\sigma = 1 + \lambda h + \frac{1}{2} \lambda^2 h^2 + \dots + \frac{1}{k!} \lambda^k h^k + O(h^{k+1}) \quad (36)$$

where k is the order of the time-march method. We refer to the root that has the above property as the *principal* σ -root. The property can be stated regardless of the details of the time-march method, knowing only that its leading error is $O(h^{k+1})$.

Thus the principal root is an approximation to $e^{\lambda h}$ up to $O(h^k)$. When the time-marching method incorporates the solution at time level $n-1$ or earlier (for example, the leapfrog method), additional σ -roots arise, known as *spurious* roots. These play no role in accuracy analysis, but must be considered in stability analysis.

Using O Δ E theory, a relation between the eigenvalues of the spatial operator matrix, λ , and the σ -roots can be found for any time-marching method, including explicit and implicit linear multi-step, predictor-corrector, and Runge-Kutta methods. The λ - σ relation encodes the properties of a time-marching method in terms of both stability and accuracy of the homogeneous solution. The accuracy of the particular solution can also be determined from the O Δ E solution. Examples of λ - σ relations include

$$\sigma = 1 + \lambda h \quad (37)$$

2. Will the numerical solution converge to the exact solution of the PDE as the grid is refined?

The second question leads to the notion of Lax stability and Lax's theorem, which can quickly lead to analysis that is quite complex for a first course. Hence we concentrate on the first question, that is, we approach stability primarily from the ODE point of view.

This approach leads to the criterion[†]

$$|\sigma_m| \leq 1 \quad \text{for all } m \quad (44)$$

for both principal and spurious roots, i.e., all roots must lie on or within the unit circle in the complex plane. Using the λ - σ relation for any time-marching method, one can plot its stability contour, which is the locus of points for which the largest $|\sigma|$, considering all σ -roots, both principal and spurious, is equal to unity. This contour is plotted in the complex λh plane and divides this plane into stable and unstable regions. For a given ODE in the form of Eq. 14, the spectrum of eigenvalues of A must lie within the stability contour of the time-marching method. This requires the choice of an appropriate method and time step.

For example, consider the family of one-step time-marching methods given by the following expression

$$u_{n+1} = u_n + h[(1 - \theta)u'_n + \theta u'_{n+1}] \quad (45)$$

This family includes the explicit Euler ($\theta = 0$), the trapezoidal ($\theta = \frac{1}{2}$), and the implicit Euler methods ($\theta = 1$). Its $\lambda - \sigma$ relation is

$$\sigma = \frac{1 + (1 - \theta)\lambda h}{1 - \theta\lambda h} \quad (46)$$

Fig. 1 displays the corresponding stability contours. The explicit Euler method is stable for a finite portion of the left half-plane. This leads to *conditional stability*. For a given set of λ -eigenvalues, there is always an upper bound on the time step above which the method is unstable. Note that for imaginary eigenvalues the explicit Euler method is *unconditionally unstable*. Hence it is inappropriate for application to the linear convection equation with periodic boundary conditions and centered space differencing, since this leads to pure imaginary λ -eigenvalues, as shown in Eq. 20. Both the trapezoidal and implicit Euler methods are stable for the entire left half-plane and are thus *unconditionally stable*. These examples show typical stability characteristics for explicit and implicit methods, which must be weighed against the increased cost per time step associated with implicit methods. The deciding factor is often the *stiffness* of the ODE system being solved.

[†]Actually the criterion is $|\sigma_m| < 1$ if one includes defective systems, which are covered in the book.

References

- [1] Lomax, H., Pulliam, T.H., and Zingg, D.W., *Fundamentals of Computational Fluid Dynamics*, 1999, available at <http://goldfinger.utias.utoronto.ca/dwz/> and <http://george.arc.nasa.gov/~tpulliam/home.html>.
- [2] Lomax, H., "Recent Progress in Numerical Techniques for Flow Simulation," *AIAA J.*, Vol. 14, No. 4, April 1976, pp. 512-518.
- [3] Hirsch, C., *Numerical Computation of Internal and External Flows*, volume 1,2. John Wiley & Sons, New York, 1988.
- [4] Anderson, D., Tannehill, J., and Pletcher, R., *Computational Fluid Mechanics and Heat Transfer*. McGraw-Hill Book Company, New York, 1984.