Introduction to Computational Fluid Dynamics

Bram van Leer and Kenneth G. Powell
Department of Aerospace Engineering, University of Michigan, Ann Arbar, MI, USA

1 Origin and Scope of CFD 1
2 CFD and Aerospace Engineering 2
3 Principles of CFD 5
4 Analysis Tools 9
5 Anatomy of a Flow Solver 12
6 Concluding Remarks 13
References 13

1 ORIGIN AND SCOPE OF CFD

Computational fluid dynamics (CFD) was born during World War II, when scientists at Los Alamos National Laboratory were not only developing the atomic bomb, but also the numerical tools to describe the violent flow created by such a device. Among them was mathematician J. von Neumann, who contributed the keystone method of artificial viscosity for "capturing" shocks in numerical solutions (Richtmyer and Morton, 1967) and is regarded as the father of CFD. His method immediately challenged the world's earliest programmable electronic computers.

In the 65 years of its existence, CFD has earned itself a respectable place alongside the established disciplines of theoretical and experimental fluid dynamics. It is a branch of science that attracts mathematicians, physicists, and engineers alike, possibly because of the strong sense of empowerment it confers on its practitioners. Developers of CFD methods create a virtual reality that users may populate with anything that flows, regardless of scale. It could be a trickle of molecules finding their way through a micro-electro-mechanical system (MEMS) microchannel or the river of air that lifts an entire airplane; it could be the flame moving through a combustion chamber or the melting core of a nuclear reactor gone unstable; it could be the earth's restless oceans and atmosphere or the gaseous disk of a spiral galaxy. Moreover, the concept of flowing is broad: traffic flows over multilane highways, and an ensemble of stars flows through phase space; both may be treated with CFD. The bottom line: CFD methods allow the user virtual experiments that in the words of P.L Roe, would be "expensive, difficult, or dangerous, [or] impossible" in the real world.

For most purposes, the fluid can be regarded as a continuum, and its dynamics are described by conservation laws cast in the form of partial differential equations (PDEs) or, when formulated for a small but finite volume of fluid, as integral equations. In CFD, these modeling equations are discretized on a computational grid, leading to finite-difference, finite-volume, or finite-element approximations. If there are not enough collisions in the fluid within the space and time scales of interest, the fluid may no longer be regarded as a continuum; often, pseudo-particles are then introduced, moving through a background grid, and transporting/exchanging the fluid's properties. In aerospace engineeering, this technique is particularly used to describe the flow around a reentry vehicle in the upper atmosphere (see Computational Modeling of Rarefied Gas Flows).

Although CFD methods are often applied beyond the range of flow parameters and space – time scales of experiments, the reliability of computed flows can only be assessed under conditions for which experimental and/or theoretical results are available for comparison. Standard practice is to at least carry

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

out the same computation on a sequence of grids with increasing resolution; such a "convergence study" provides insight into the proximity of the exact solution to the PDEs modeling the flow physics. But modeling errors remain, while, on the other hand, the experiments used for comparison have their own uncertainties in set-up conditions and measurement.

With so much riding on flow simulations, there is a growing need and demand for methods of measuring and predicting reliability of computed solutions and computational methods. Hence the current boom in the areas of Validation and Verification (V&V) and Uncertainty Quantification (UQ) (see Verification and Validation of CFD Based Turbulent Flow Experiments).

2 CFD AND AEROSPACE ENGINEERING

While aerospace engineering is currently the discipline leading the ongoing development of CFD, the previous section points out that CFD was not born within aerospace engineering. In fact, it was not until 1980 that aeronautics caught up with the advances in CFD made in weapons' research and astrophysics; from then on, it did take the lead. This historical development is sketched in the following section; the technical terms appearing in the text are explained in subsequent technical sections.

2.1 CFD before 1980

Throughout the 1950s, CFD development in the United States remained largely in the hands of the national laboratories in Los Alamos and Livermore, home of the largest existing computers, and was therefore chiefly used for weapons research. The only academic player in the early development of CFD was the Mathematics Department of New York University, which boasted R. Courant, K.O. Friedrichs, and P.D. Lax in its Mathematics Department and received major funding from the Atomic Energy Commission (AEC). In the USSR, the work of mathematician S.K. Godunov was the most significant; he contributed a method for compressible flow (Godunov, 1959) after which most modern finite-volume and discontinuous Galerkin (finite-element) methods are modeled. For an account of the early development of CFD in the USSR, read Godunov's (1999) memoir.

The national laboratories produced large computer codes capable of dealing with any combination of deformable media: gaseous, liquid, or solid. There was little use for these codes in aerospace engineering, which dealt chiefly with low-speed steady flows that could be described as small perturbations to a uniform stream, governed by the linear potential equation.

The availability of electronic computers made it possible to compute such flows with panel methods. In this approach, the flow about, say, a wing is thought of as the resultant of a continuum of vortices distributed over the discrete panels approximating the wing shape. Imposing the no-penetration condition on each panel generates a large linear system for the panel-wise parameters of the vortex distribution. Once this has been solved, the flow in the entire three-dimensional domain exterior to the wing is known. This makes panel methods unbeatable in efficiency – but they cease to be useful when nonlinear effects, due to compressibility, become significant. Early panel code development took place in the 1960s at Douglas Aircraft J. Hess and A. Smith, and at Boeing P.R. Rubbert and G.R. Saaris. The methods were useful enough that Boeing, Douglas, Lockheed, McDonnell, and NASA Ames Research Center (ARC) all had a substantial effort in panel-method and panel-code development (Hess and Smith, 1966).

At this point, it is worth noticing that supersonic flight had already been achieved by the Air Force as early as 1947 and that NACA changed to NASA in 1958, heralding the space age, with its emphasis on rocket flight. But no focused effort in CFD for compressible flow appears to have developed within the aerospace community.

The core algorithm in the early compressible-flow codes was only of first-order accuracy, meaning that the error in the numerical solution decays with the first power of the mesh width when the grid is refined. Not surprisingly, the 1960s brought the exploration of second-order methods, which promised higher efficiency, that is, greater accuracy achieved for given computing time. In this decade, the contribution to CFD by universities rose sharply as International Business Machines (IBM) and Control Data Corporation (CDC) computers became available to the wider academic community.

Best known from this era are the second-order method of Lax and B. Wendroff 1960 and its 1969 variant due to aerospace engineer R.W. MacCormack. Unfortunately, this class of methods turned out to be of limited utility. Numerical oscillations created in the vicinity of a discontinuity would drive temperatures, densities, and/or species concentrations negative, making such methods suitable neither for computing supersonic flows with strong shocks nor for the mere advection of water vapor or trace elements in the atmosphere.

A breakthrough in numerical analysis, and with it the birth of modern CFD, came in the early 1970s as *high-resolution methods* appeared on the scene. These are at least second-order accurate when the solution is smooth, while they capture discontinuities as narrow, monotone structures, owing to the use of a *limiter*.

The first families of high-resolution methods were the flux-corrected transport (FCT) methods of astrophysicist

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

J.P. Boris and Collaborators, and the MUSCLs (Monotone Upstream Scheme for Conservation Laws) family of astrophysicist B. van Leer. Both kinds of schemes include a *limiter* that reduces the higher-order terms in the update scheme so as to avoid the onset of numerical oscillations. In an FCT method, which is a predictor–corrector method, the corrector step creates the higher-order accuracy; therefore, its fluxes are limited so as not to create any new extremum. MUSCL is a Godunov-type method, which means that the interaction of discrete fluid cells at their interface is computed based on the solution to the local *shock-tube* or *Riemann* problem (see Riemann Solvers in Aerospace Engineering). In MUSCL, numerical oscillations are regarded as the result of non-monotone interpolation of the discrete initial values; the remedy therefore is to limit the values of the second- and higher derivatives in the interpolant.

By the end of the 1970s, both techniques were mature enough to be suited for widespread use. For a variety of reasons, FCT methods were and are mostly applied to unsteady, highly energetic, often reacting flows (Oran and Boris, 2001); MUSCL (Van Leer, 1979), on the other hand, has become the method of preference in many disciplines, particularly in aerospace engineering.

Meanwhile, the aeronautical community had ventured into nonlinearity in order to tackle the modeling of transonic flow. While panel methods still remained an important tool, computational aerodynamics passed through a sequence of nonlinear flow models of increasing fidelity. The first two were still potential-flow models, hence isentropic and irrotational, but did deal with the transonic nonlinearity. Murman and Cole (1971) were the first to compute discrete solutions to the transonic small-disturbance equation; computations based on the full transonic potential equation followed within a few years. At NASA ARC, Beam and Warming (1978) made a large leap, pioneering an implicit, compressible Navier–Stokes scheme, implemented with aid of a kind of dimensional operator splitting, "approximate factorization," also developed at ARC. In retrospect, the development of this Navier–Stokes solver must be regarded as premature. Knowledge of limiting had not yet reached aeronautics, so the Euler discretization embedded in the Beam–Warming scheme was oscillatory in the presence of shocks, hence not useful by itself and not a good foundation for a Navier–Stokes scheme.

2.2 CFD development in the 1980s

The credit for introducing Godunov-type high-resolution methods to the aerospace community goes to the Institute for Computer Applications in Science and Engineering (ICASE, 1972–2002), established at NASA's Langley Research Center (LaRC) established by the Universities Space Research Administration (USRA). ICASE was set up to have Langley engineers interact with US and foreign scientists, and it handled this task so well that LaRC became the new leader in the development of CFD, for aeronautics as well as for general use, in the United States as well as worldwide.

Among ICASE's visitors and their contributions, the most prominent were mathematician A. Brandt (Israel), who introduced multi-grid relaxation in solving the full potential equation (1976); mathematician D. Gottlieb (Israel, †2008), who introduced spectral methods; M.Y. Hussaini (India), who headed a group studying transition to turbulence with spectral methods; astrophysicist B. van Leer (Netherlands), who introduced Godunov-type methods and matching classical relaxation methods, leading to LaRC's CFL2D/CFL3D Euler/Navier–Stokes codes (Rumsey, Biedron and Thomas, 1997); mathematician A. Harten (Israel, †1994), who contributed total-variation-diminishing (TVD) schemes; P.L. Roe (UK), who developed genuinely multidimensional Euler methods, now called residual-distriobution schemes (Deconinck and Koren, 1997); finally, mathematician S.J. Osher (US), who, together with Harten, developed essentially non-oscillatory (ENO) interpolation. An anthology of ICASE-based papers on the subjects of limiters, non-oscillatory interpolation, and Riemann solvers, with historical and technical notes, can be found in the book *Upwind and High-Resolution Schemes* (Hussaini, Van Leer and Van Rosendale, 1997). Another source on high-resolution schemes is the 30th anniversary issue of the *Journal of Computational Physics* (Brackbill, 1997), which reprinted a number of landmark papers on computational physics, including the key papers on FCT and MUSCL, and Roe's (1981)much-cited paper, introducing the Roe–Riemann solver.

The CFD methodology created in this period was so rich that the interdisciplinary flow of information reversed itself, with aerospace engineering now impacting weapons and high-energy research.

The most influential contributions made outside the high-resolution arena, and limited to aeronautical applications, were due to aerospace scientist A. Jameson, who created a suite of efficient Euler and Navier–Stokes codes using higher-order artificial viscosity for stability, Runge–Kutta methods for marching toward a steady state, and multigrid relaxation for convergence acceleration (Jameson, Schmidt and Turkel, 1981).

The pure Euler era in aerospace engineering lasted only 5 years; in 1985, the first Navier–Stokes codes built upon high-resolution Euler methods appeared. Convergence acceleration became more necessary than ever; both implicit and explicit marching methods were explored. As vector

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

computing became the norm, the relative efficiency of these two strategies was modulated by the degree of vectorization they would allow. A decade later, the same would happen with regard to the degree of parallelization afforded. In the later 1980s, grid adaptation, especially tree-structured adaptive Cartesian grids with cells cut at a solid boundary, suited for Euler calculations, emerged as a research subject. At the same time, fully unstructured grids for aeronautical problems, first with triangular, later with tetrahedral cells, were being developed. T.J. Baker (†2006) and Jameson (Jameson, Baker and Weatherill, 1986) were the first to produce 3-D inviscid flow solutions for a simplified airplane on a tetrahedral grid.

Toward 1990, most developers and users of high-resolution codes for gas dynamics and aerodynamics were rather satisfied with the performance of their codes in a wide range of flow problems – except in the hypersonic flow regime. By 1988, it was clear that highly successful finite-volume codes like CFL2D produced bizarre solutions of steady flow around a blunt body on perfectly smooth grids, if the Mach number rose above 5. Typically, the bow-shock would exhibit an asymmetric, tumor-like growth, or "carbuncle." Although insight in this phenomenon has greatly increased since then, to date, no discretization has been found that is "carbuncle-proof."

The 1980s were the decade in which CFD started to be taught as a new discipline at major universities. A brief history of the teaching of CFD can be found in Van Leer (2000), which is one contribution to a *CFD Journal* issue (Hafez, 2000) dedicated entirely to CFD education.

2.3 CFD development in the 1990s

In spite of the carbuncle, around 1990, CFD research started to move away from developing basic Euler/Navier–Stokes discretizations, turning to adaptive and unstructured grids and formulating schemes for such grids. This opened up the possibility of computing multiscale flows. In addition, there was a growing research effort in treating more complex flow physics, such as multi-fluid dynamics. The latter area became more easily accessible in 1992 with the development of the *level-set* method by W.A. Mulder, Osher and J. Sethian, in which the fluid interface is defined as the zero-level surface of a distance function for which an evolutionary equation is available. Furthermore, convergence acceleration for Euler and Navier–Stokes solvers remained a research focus; a key achievement in 1998 was the computation of inviscid lifting flow over an airfoil with N unknowns in $O(N)$ operations by D.L. Darmofal and K. Siu using a fully explicit methodology that included local preconditioning and multigrid relaxation with semicoarsening as essential building blocks. Finally, grid optimization, design optimization and uncertainty quantification made great strides with the advent of methods for solving the *adjoint* flow equations, which detail the influence of each fluid element on the value of some global design parameter such as the drag; the implementation of this methodology in aeronautical CFD is due to Jameson (1989).

A quantum jump in computational power made the 1990s the first decade of massively parallel, high-performance computing and communication (HPCC, see High Performance Computing for CFD) and of grand-challenge applications; both types of activities were well funded and, to a significant degree, steered research efforts worldwide.

2.4 CFD development in the twenty-first century

During the first decade of the twenty-first century, CFD has become more powerful than ever owing to the circumstance that massively parallel computing has come within reach of every research group. There is a great diversity of challenging applications that require the treatment of increasingly complex physics in the presence of an increasingly complex geometry. Given this trend toward complexity, it is understandable that grid optimization, design optimization, and uncertainty quantification remain important research subjects.

To handle physical complexity efficiently, it is advisable to use a formula for the interface flux that requires very little knowledge of the physical content of the system of equations, such as the three-wave Riemann solver of Harten, Lax and van Leer (1983); to handle geometric complexity, it is advantageous to borrow generously from finite-element methods (FEMs). Specifically, when expanding the subcell solution in terms of a set of basis functions, it is becoming commonplace to treat the coefficients of the expansion as independent quantities, each with its own update equation, rather than to compute these by interpolating the solution in the cell's neighborhood. This potentially allows one to maintain an arbitrarily high order of accuracy (Barth and Deconinck, 1999), which in turn may be used to counter the effects of poor grid quality. By trading the mesh size (h) for the order of accuracy (p) where the solution is smooth, and vice versa where intricate spatial detail asks to be resolved, one arrives at a so-called h–p refinement strategy, also regarded as essential to modern CFD.

The chief example of a method in this class is the vintage discontinuous Galerkin (DG) method (1973), which typically uses a polynomial basis in each cell; the update equations result from taking in a cell the inner product of the equations with each of the basis functions. How to match a DG method with a multidimensional limiter that does not degrade its potentially high accuracy is still a subject of research.

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

2.5 Recommended reading

This brief history of CFD has focussed on the advancement of basic algorithms for compressible flow. A more elaborate narrative by the same authors, limited to developments in North America (Van Leer, 2009), can be found in Volume 100 of the Springer series *Notes on Numerical Fluid Mechanics* (Hirschel and Krause, 2009), which is a worldwide retrospective on 40 years of numerical fluid mechanics and aerodynamics. The volume also includes contributions describing state-of-the-art CFD applications in a wide variety of disciplines. Methods for low-speed flows are reviewed in Drikakis and Rider (2005). For applications within aerospace engineering, we refer to the various specialized chapters in this encyclopedia, especially those in Volume 1, Fluid Dynamics and Aerothermodynamics.

3 PRINCIPLES OF CFD

As CFD preoccupies itself with the numerical approximation of partial differential equations, it must subscribe to a set of sound principles of numerical analysis. In this and the next section we discuss these principles and some associated analysis tools, illustrated by simple finite-difference examples.

3.1 Discretization

A computer can only store and handle a finite amount of information; thus, solutions of PDEs must be represented by limited data. For this purpose, we usually divide the space–time continuum into small domains (meshes or cells) by a computational net or grid. The data may, for instance, be prescribed as point samples in the nodal points of the net, suited for *finite-difference* methods (FDMs), or as averages over the interior of the meshes, suited for *finite-volume* methods (FVMs). In these cases, the detailed behavior of the solution inside a mesh must be estimated by interpolation. In contrast, in *finite-element* methods (FEMs), the solution inside a mesh is described in detail by a sum of basis functions, the weights of which are independent data. The process of representing the solution by a reduced set of data is called *discretization*, as is any approximation of the PDE using these data.

3.2 Consistency

A numerical approximation to a PDE must be *consistent* with the PDE, meaning that the error committed in the discretization must vanish when the mesh size of the grid is made infinitesimally small. To understand this, assume the PDE has the form $\mathcal{D}_{\text{ex}}(U) = 0$ and its finite-difference approximation is $\mathcal{D}_{\text{fd}}(u) = 0$, where u is a discrete approximation at the grid's nodal points $(j\Delta x, n\Delta t)$ to the PDE's exact solution U. We may regard these discrete values, u_j^n, as point samples taken from the exact solution to a slightly different PDE $\mathcal{D}_{\text{ex}}(u) = TE(u)$; this is called the *modified equation* and its right-hand side the *truncation error*, named so because it basically results from truncating a series expansion of the exact solution around a nodal point. It should vanish with some positive power of Δx and/or Δt. To illustrate consistency and other concepts, we consider the one-dimensional (1D) linear advection–diffusion equation,

$$U_t + aU_x - \mu U_{xx} = 0, \quad a, \mu \text{ constant}, \quad \mu \geq 0 \qquad (1)$$

widely used as a model for the Navier–Stokes equations. A family of consistent update schemes with one free dimensionless parameter $q > 0$ can be formulated on the stencil of four nodal points shown in Figure 1; cast in the form most resembling the PDE it reads,

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} - \left(\mu + \frac{q(\Delta x)^2}{2\Delta t}\right)\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} = 0 \qquad (2)$$

The q term is needed to stabilize the advection term (see below). A full truncation-error analysis of this advection–diffusion scheme is lengthy; here, we restrict ourselves to the two limiting cases.

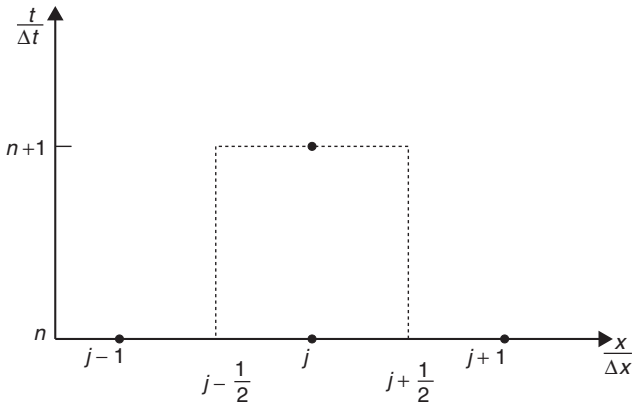

Figure 1. Nodal-point stencil for the q-scheme. The dashed box indicates the domain to which the conservation law applies.

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

We first study pure advection, that is, $\mu = 0$; the update scheme then reads

$$u_j^{n+1} = u_j^n - \frac{\nu}{2}(u_{j+1}^n - u_{j-1}^n) + \frac{q}{2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n),$$
$$\nu = \frac{a\Delta t}{\Delta x} \tag{3}$$

The dimensionless quantity ν first appeared in a 1928 paper by Courant, Friedrichs and H. Lewy and is widely known as the "CFL number." By assuming a truncation error with leading term $\Delta x C_2 u_{xx}$ and expanding the finite differences in equation (2) with Taylor series, using the modified equation to eliminate time derivatives, we find

$$\text{TE} = \frac{(\Delta x)^2}{2\Delta t}(q - \nu^2)u_{xx} + \text{higher-order terms} \tag{4}$$

The presence of Δt in the denominator of the leading error term suggests that q should include a factor Δt or, rather, the nondimensional factor $|\nu|$; otherwise, the error will blow up if we take an arbitrarily *small* time step on a given spatial grid. A choice yielding consistency is $q = |\nu|$; this produces the so-called *upwind-differencing* advection scheme,

$$u_j^{n+1} = u_j^n - \begin{cases} \nu(u_j^n - u_{j-1}^n), & a, \nu \geq 0 \\ \nu(u_{j+1}^n - u_j^n), & a, \nu < 0 \end{cases} \tag{5}$$

Another consistent choice, $q = \nu^2$, due to Lax and Wendroff, makes the leading error in (4) vanish, rendering the scheme second-order accurate in space and time. In contrast, the least accurate q-scheme, due to Lax and Friedrichs, has $q = 1$ for all ν and therefore becomes inconsistent if we choose Δt to be as small as $O((\Delta x)^2)$. Below, we shall see that this choice is actually required for stability once diffusion enters and dominates.

For pure diffusion, that is, $a, \nu, q = 0$; the update scheme reads

$$u_j^{n+1} = u_j^n + r(u_{j+1}^n - 2u_j^n + u_{j-1}^n), \quad r = \frac{\mu\Delta t}{(\Delta x)^2} \tag{6}$$

Its truncation error, with assumed leading term $(\Delta x)^2 C_4 u_{xxxx}$, is found to be

$$\text{TE} = \frac{(\Delta x)^4}{2\Delta t} r(\frac{1}{6} - r)u_{xxxx} + \text{higher-order terms} \tag{7}$$

Anticipating that $\Delta t \sim (\Delta x)^2$, we estimate the leading term to scale with $(\Delta x)^2$, indicating consistency.

3.3 Stability

An initial-value problem for a PDE is well posed if a small perturbation to the initial values leads to a small perturbation in the solution at any later time. When applied to such a problem, a finite-difference method is said to be *stable* if a small perturbation of the initial values leads to a small perturbation in the numerical solution at any fixed later time, regardless of the mesh size. For steady-flow problems, this "later time" gets moved to infinity, necessitating the stronger *absolute* stability.

The most powerful tool for investigating if a difference scheme is stable is the Von Neumann analysis, described further in Section 4. It is much more straightforward to carry out than a truncation-error analysis; for the general q-scheme (2), the stability condition becomes

$$\nu^2 \leq q + 2r \leq 1 \tag{8}$$

Observe that (8) includes a condition on the CFL number, $|\nu| \leq 1$ or $\Delta t \leq \Delta x/|a|$; in words: the time step is not to exceed the time needed to advect across one mesh (see Figure 2). We may also say that the absolute advection speed must not exceed the *grid speed* $\Delta x/\Delta t$, which is the speed at which information can spread through the grid.

For pure advection one may actually choose $\nu = \pm 1$; stability then requires that $q = 1$ for all q schemes. From (5), it is seen that the schemes reduce to the upwind shift operator, $u_j^{n+1} = u_{j\mp 1}^n$, that is, exact advection over a distance $\mp\Delta x$. This exactness for $|\nu| = 1$ is a desirable property, not shared by all advection schemes.

The lower stability bound for q can also be gleaned from the truncation error (4): the leading term is a stabilizing

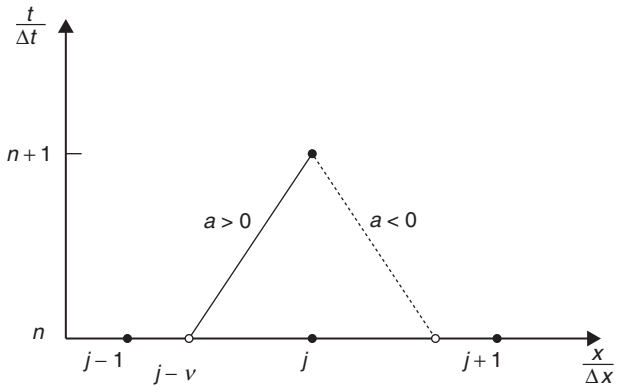

Figure 2. Graphical depiction of the CFL condition. Shown are advection paths for positive (solid) and negative (dashed) advection speeds. The point where the advection starts at t^n (open circle) must remain within one mesh width from x_j.

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

diffusion term if $q > \nu^2$. When $q = \nu^2$, the term vanishes; the Lax–Wendroff scheme is stabilized by another error term $\sim (\Delta x)^3 u_{xxxx}$, with negative coefficient.

The presence of diffusion restricts the time step more strongly than the CFL condition. The balance between advection and diffusion can be described by the Péclet number $Pe = |a|L/\mu$ (similar to the Reynolds number), where L is a significant length scale; if we choose $L = \Delta x$, we get the computationally relevant cell-Péclet number $Pe_\Delta = |a|\Delta x/\mu = |\nu|/r$. To fix our thoughts we take the specific case $q = |\nu|$; stability then requires $|\nu| + 2r \le 1$ or $\Delta t \le (\Delta x/|a|)(Pe_\Delta/(Pe_\Delta + 2))$. When Pe_Δ is large, which can be achieved by making Δx sufficiently large, advection dominates; the diffusion scale $\mu/|a| \ll \Delta x$ is not resolved on the grid. When refining the mesh, Pe decreases $\sim \Delta x$; Δt initially also scales with Δx. When refinement continues, Pe_Δ eventually becomes small, diffusion dominates, and $\Delta t \sim Pe_\Delta \Delta x \sim (\Delta x)^2$; the diffusion scale is now well resolved. With $q \sim |\nu|$, the truncation error for scheme (2) is not larger than $O(\Delta t)$ for any value of Pe_Δ, so the advection–diffusion equation is always consistently approximated.

For pure diffusion, that is, $a, \nu, q = 0$, the stability condition reduces to $r \le 1/2$. The truncation error (7) provides no insight in the stability of scheme (6). In fact, for $r < 1/6$, the leading term $\sim (\Delta x)^2 u_{xxxx}$ has a positive coefficient and is destabilizing, but it is overcome by the diffusion term in the PDE itself.

Observe that when diffusion dominates, the grid speed $\Delta x/\Delta t$ will go to infinity with mesh refinement. This relates to the fact that the characteristic speeds of the diffusion equation are $\pm\infty$: an initial disturbance propagates instantaneously along the entire real axis. An explicit update scheme like (6) cannot mimic this but will approach the effect as the grid gets refined and the PDE better approximated. In contrast, an implicit update scheme such as the 1947 Crank–Nicholson scheme,

$$u_j^{n+1} - \frac{r}{2}(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}) = u_j^n + \frac{r}{2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \qquad (9)$$

produces $\{u_j\}^{n+1}$ as the solution of a linear system composed of equation (9) for all interior j and the boundary conditions. Thus, a disturbance spreads to the boundaries in one time step, mimicking infinite propagation speed. The scheme is stable for all $r > 0$; its truncation error is $O((\Delta t)^2) + O((\Delta x)^2)$ when Δt and Δx are independently refined.

At domain boundaries, a discretization generally needs modification as the stencil used in the domain's interior cannot be maintained, and boundary conditions are added that need discretization. The tool to evaluate the stability of an interior/boundary-scheme combination, fully developed by 1972, is due to B. Gustafsson, H.O. Kreiss and P. Sundström (1972); its treatment falls beyond the scope of this chapter.

3.4 Convergence

When computing with one difference method a sequence of discrete solutions on grids of ever-increasing spatial and temporal resolution, we want to be sure that this sequence *converges* to the exact solution of the problem. Already in 1953, Lax proved his famous *Equivalence Theorem* (Richtmyer and Morton, 1967), valid for a broad class of linear approximations to linear PDEs. It says that the combination of consistency and stability of the difference scheme is necessary and sufficient for convergence of the numerical solution. This theorem is often applied beyond its bounds, notably to nonlinear problems; still, consistency and stability remain *necessary* conditions for convergence.

3.5 Monotonicity

Starting from a monotone initial-value distribution, PDEs like the advection–diffusion equation generate a solution that remains monotone for all times. It is desirable for a discretization to share this property with the PDE; such schemes are called *monotonicity-preserving* or *non-oscillatory*. Monotonicity-preserving schemes evidently guarantee the *positivity* of a numerical solution starting from positive data.

To design such a scheme, one imposes the condition that the update scheme not create a new local extremum from a monotone sequence of initial data. Godunov (1959) showed that a linear update scheme of the form $u_j^{n+1} = \sum_{l=-k}^{k} c_l u_{j+l}^n$ must have nonnegative coefficients in order to preserve monotonicity; for an advection scheme, this unfortunately means that it cannot be better than first-order accurate, that is, TE $\sim O(\Delta x, \Delta t)$. This is known as *Godunov's barrier theorem*; it can be circumvented by making the scheme nonlinear, as explained in Section 4.

The q-scheme (2) has $k = 1$ and coefficients $c_{-1} = \frac{1}{2}(q + 2r + \nu)$, $c_0 = 1 - (q + 2r)$, and $c_1 = \frac{1}{2}(q + 2r - \nu)$; requiring nonnegativity yields the monotonicity condition

$$|\nu| \le q + 2r \le 1 \qquad (10)$$

which is seen to be stronger than the stability condition (8). Making $q + 2r$ equal to the lower limit yields the most accurate monotonicity-preserving scheme of the family; for pure advection, $r = 0$, it turns out to be the upwind-differencing

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

scheme (5). For pure diffusion, $\nu, q = 0$, the monotonicity condition is the same as the stability condition: $r \leq 1/2$.

A generalization of nonnegativity of an update scheme's coefficients is the property that increasing or decreasing any one of the scheme's input values causes the output value to vary in the same direction; this property is called *positivity* and remains useful in multiple dimensions. Another multidimensional concept, this one from pure boundary-value problems, is the *maximum principle*, which says that the solution maximum on the domain or any subdomain is always found on the (sub)domain's boundary; it is desirable for discrete solutions to have the same property.

3.6 Conservation

The equations of fluid dynamics express conservation of mass, momentum, and energy; it is important for a discrete approximation to express these same principles, in particular, for the sake of producing the correct propagation speed of shock waves. To understand this, consider the 1D Navier–Stokes equations written in the *divergence form* or conservation form,

$$\boldsymbol{U}_t + (\boldsymbol{F}(\boldsymbol{U}) - \boldsymbol{D}\boldsymbol{U}_x)_x = 0 \tag{11}$$

where $\boldsymbol{U}$ is the vector of conserved state variables (density, momentum density, and total-energy density), $\boldsymbol{F}$ is the vector of their inviscid fluxes or Euler fluxes, and $\boldsymbol{D}$ is the matrix-valued dissipation coefficient containing the viscosity and thermal conductivity. Referring to the grid of Figure 1, we recover an integral conservation principle by integrating the PDE over the computational space–time cell $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [t^n, t^{n+1}]$:

$$\frac{\overline{\boldsymbol{U}}_j^{n+1} - \overline{\boldsymbol{U}}_j^n}{\Delta t} + \frac{\langle \boldsymbol{F} - \boldsymbol{D}\boldsymbol{U}_x \rangle_{j+\frac{1}{2}}^{n+\frac{1}{2}} - \langle \boldsymbol{F} - \boldsymbol{D}\boldsymbol{U}_x \rangle_{j-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} = 0 \tag{12}$$

here, overhead bar and angled brackets indicate uniform averaging over the space and time intervals, respectively, with the intervals labeled by their midpoint. The above equation shows that the amount of the quantity $\boldsymbol{U}$ contained in the space interval ($= \overline{\boldsymbol{U}}\Delta x$) can only change in time because of what flows in or out through the boundaries ($= \mp\langle \boldsymbol{F} - \boldsymbol{D}\boldsymbol{U}_x \rangle \Delta t$). This is the integral conservation property from which in the inviscid case ($\boldsymbol{D} = 0$) the jump equations for shock propagation derive. Equation (12) looks like a discrete scheme but is *exact*; numerical errors will be committed once we approximate the time integral of the flux.

As an example, we extend the finite-difference q-scheme (2) to the nonlinear system (11), while converting it to a finite-volume scheme by regarding the discrete data as cell averages. The resulting scheme approximates the Navier–Stokes flux $\langle F_{\mathrm{NS}} \rangle_{j+\frac{1}{2}}^{n+\frac{1}{2}}$ by

$$\boldsymbol{F}_{\mathrm{NS}}(\overline{\boldsymbol{u}}_j^n, \overline{\boldsymbol{u}}_{j+1}^n) = \frac{1}{2}\{\boldsymbol{F}(\overline{\boldsymbol{u}}_j^n) + \boldsymbol{F}(\overline{\boldsymbol{u}}_{j+1}^n)\} - \left(\frac{(\Delta x)^2}{2\Delta t} \boldsymbol{Q}(\overline{u}_j^n, \overline{u}_{j+1}^n) + \boldsymbol{D}(\overline{u}_j^n, \overline{u}_{j+1}^n) \right) \frac{\overline{\boldsymbol{u}}_{j+1}^n - \overline{\boldsymbol{u}}_j^n}{\Delta x} \tag{13}$$

where $\boldsymbol{Q}$ is a suitable matrix satisfying a stability or monotonicity constraint. $\boldsymbol{F}_{\mathrm{NS}}(u_j^n, u_{j+1}^n)$ is called a "numerical flux function"; it must at least satisfy the consistency condition $\boldsymbol{F}(\boldsymbol{U}, \boldsymbol{U}) = \boldsymbol{F}(\boldsymbol{U})$. The use of a unique numerical flux value at each flux point $x_{j+\frac{1}{2}}$ and a unique way of evaluating $\overline{u}_j$ at each time level results in cancellation of all interior terms when summing equation (12) over a domain in space–time, leaving only values near the domain boundary to contribute; thus, conservation in the large is ensured.

Within the Navier–Stokes flux it is especially the inviscid contribution that has been the subject of extensive numerical research. A classification of numerical Euler flux functions starts with the flux arising at $x_{j+\frac{1}{2}}$ after the initial discontinuity between the input states $\boldsymbol{u}_j$ and $\boldsymbol{u}_{j+1}$ (thought to extend uniformly away from the interface) breaks up, as in a shock tube. This interaction problem is known as *Riemann's initial-value problem*; its exact solution is known and was first used by Godunov (1959). The numerical flux based on this solution is automatically *upwind biased*. Any other flux function may be regarded as deriving from an "approximate Riemann solver;" Riemann Solvers in Aerospace Engineering is dedicated to this subject.

3.7 Irreversibility

For a discretization of a nonlinear PDE to be capable of describing shocks, it must be both conservative and *irreversible*. Irreversibility is needed to account for the irreversible processes (friction and heat conduction) that occur inside a shock, even if the shock is regarded as a mathematical discontinuity. This was already understood by Von Neumann, who was the first to include *artificial viscosity* (Richtmyer and Morton, 1967) in a scheme for compressible flow. Without such a term, schemes will become unstable or violently oscillatory when shocks arise. Hence, the time-reversible *leap-frog* scheme,

$$u_j^{n+2} = u_j^n - \nu(u_{j+1}^{n+1}) - u_{j-1}^{n+1}) \tag{14}$$

may be used for linear-wave propagation, for example, in acoustics, but should not be extended and applied to the compressible Euler equations. Discretizations of the Euler equations generally also include artificial heat conduction and

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

even artificial mass diffusion; all kinds are collected under the rubric of *artificial dissipation*. Although needed for stability in shocks, artificial dissipation is still a numerical error and must be kept as low as possible elsewhere. Particularly in Navier–Stokes discretizations, the artificial dissipation errors stemming from the Euler terms must be kept sufficiently small compared with the physical dissipation terms modeled in the equations, lest boundary layers thicken unphysically and computed drag values become inaccurate. Note that the thickness of captured shock structures, typically a few mesh widths, is grossly exaggerated, but this usually does not affect the quantities of interest. Only if the actual shock structure is of importance, as in hypersonics, it must be accurately resolved by the grid.

4 ANALYSIS TOOLS

4.1 Von Neumann analysis

Von Neumann estimated the growth of solution perturbations by a Fourier analysis of the update scheme linearized around a constant reference state (Richtmyer and Morton, 1967); we shall demonstrate the procedure in one dimension.

Consider a complex initial perturbation of the form $\epsilon(x, 0) = \epsilon_0 \exp(2\pi \mathrm{i} kx)$, where k is the spatial frequency of a single Fourier mode; when inserting these initial values into the finite-difference scheme linearized around the state $\boldsymbol{U}_0$, the updated solution takes the form $\epsilon(x_j, \Delta t) = g_{\mathrm{fd}}(\boldsymbol{U}_0, \Delta x, \Delta t, \ldots)\epsilon(x_j, 0)$. Here g_{fd} is the *amplification factor* associated with the scheme; this complex number depends on U_0, the mesh constants, and possibly some free parameters. The condition for stability over a finite time interval now reads $|g_{\mathrm{fd}}| \leq 1 + O(\Delta t)$. The margin $O(\Delta t)$ allows finite growth over a finite time and takes care of a possible source term in the PDE; for advection-type problems, we have $O(\Delta x) = O(\Delta t)$, so this margin also allows smooth spatial variations in U_0 and in Δx. By omitting the O term, we get the condition for absolute stability, $|g_{\mathrm{fd}}| \leq 1$.

When applying this procedure to the q-scheme (2) we find,

$$g(\beta, \nu, q, r) = 1 - (q + 2r)(1 - \cos\beta) - \nu \mathrm{i} \sin\beta, \qquad \beta = 2\pi k \Delta x \tag{15}$$

and

$$|g(\beta, \nu, q, r)| = \left[1 - 4\left\{q + 2r - \nu^2 - ((q + 2r)^2 - \nu^2)\sin^2\frac{\beta}{2}\right\}\sin^2\frac{\beta}{2}\right]^{\frac{1}{2}} \tag{16}$$

requiring absolute stability leads to the earlier result (8).

When applied to a system of PDEs, the Von Neumann analysis yields an *amplification matrix* $\boldsymbol{G}$; the stability condition must now be applied to all eigenvalues of $\boldsymbol{G}$. It is relatively easy to apply the Von Neumann analysis to the Euler version of the q-scheme using the flux (13) with $\boldsymbol{D} = 0$. We first linearize the Euler equations, yielding $\boldsymbol{U}_t + \boldsymbol{A}(\boldsymbol{U}_0)\boldsymbol{U}_x = 0$, where the matrix $\boldsymbol{A}(\boldsymbol{U}) = \mathrm{d}\boldsymbol{F}(\boldsymbol{U})/\mathrm{d}\boldsymbol{U}$ is the flux Jacobian; the amplification matrix then becomes

$$\boldsymbol{G} = 1 - \boldsymbol{Q}(1 - \cos\beta) - \mathrm{i}\frac{\boldsymbol{A}\Delta t}{\Delta x}\sin\beta \tag{17}$$

The matrix $\boldsymbol{Q}$ is usually chosen to have the same basis of eigenvectors as $\boldsymbol{A}$; thus, $\boldsymbol{G}$ will also share these eigenvectors, and stability can be ensured by applying the scalar condition (8), with $r = 0$, to all eigenvalues g_k of $\boldsymbol{G}$, that is,

$$|g_k| \leq 1, \quad k = 1, 2, 3 \tag{18}$$

with

$$g_k = 1 - q_k(1 - \cos\beta) - \mathrm{i}\frac{a_k \Delta t}{\Delta x}\sin\beta \tag{19}$$

in which the corresponding eigenvalues a_k of $\boldsymbol{A}$ and q_k of $\boldsymbol{Q}$ appear. Typically, the eigenvalues q_k, $k = 1, 2, 3$, are taken to be the same scalar function of the corresponding a_k or, more precisely, of $\nu_k = a_k \Delta t/\Delta x$, that is, $q_k = q(\nu_k)$; this makes the analogy with the scalar case complete.

The eigenvalues a_k are the so-called characteristic speeds; for the Euler equations, these are $a_1 = u - c$, $a_2 = u$, $a_3 = u + c$, where u and c are the flow and sound speeds, respectively. The stable range of the time step will be determined by the speed largest in absolute value: $\Delta t \leq \Delta x/(|u| + c)$. This must be regarded as a *local* stability condition, as u and c, and, possibly, Δx vary in space; at each time level, the time step must be made so small as to satisfy the condition in each nodal point of the grid.

Multidimensional schemes require a multidimensional Fourier analysis, including independent spatial frequencies in all directions.

4.2 Dissipation and dispersion analysis

As the Von Neumann analysis yields the exact solution to a difference equation, in the form of its Fourier transform g, a comparison with the amplification factor g_{ex} of the underlying PDE yields complete information about the numerical errors committed in the discretization. In g, these show up as an amplitude error, or dissipation, and a phase error, or

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

dispersion. To illustrate these two types of errors, we consider again the advection–diffusion equation (1) and its discretization (2), emphasizing the case of pure advection ($\mu, r = 0$), which is most interesting.

The amplification factor of the exact solution to the 1D advection–diffusion equation (1) is

$$g_{\text{ex}}(\beta, \nu, r) = \exp(-r\beta^2 - \mathrm{i}\nu\beta) \tag{20}$$

to which we wish to compare the q-scheme's amplification factor g_{fd} given by (15); for this purpose, we rewrite the latter as $g_{\text{fd}} = |g| \exp(\mathrm{i}\phi)$, with $|g|$ given by (16) and ϕ by

$$\phi = \arg(g_{\text{fd}}) = \arctan \frac{-\nu \sin\beta}{1 - (q + 2r)(1 - \cos\beta)} \tag{21}$$

We will first consider the embedded advection scheme ($r = 0$), which is seen to be both dissipative and dispersive. It changes the amplitude of a Fourier mode per time step by a fraction

$$|g| - 1 = -\frac{1}{2}(q - \nu^2)\beta^2 + O(\beta^4) \tag{22}$$

which means damping if the stability condition (8) is satisfied, and it changes the advection speed by a fraction

$$\frac{\phi - (-\nu\beta)}{-\nu\beta} = -\frac{1}{3}(1 - 3q + 2\nu^2)\beta^2 + O(\beta^4) \tag{23}$$

which can take both negative and positive values. The amplitude error $\sim \beta^2$, typical of a first-order-accurate update scheme, dominates the phase error $\phi - (-\nu\beta) \sim \beta^3$, except when $q = \nu^2$ (Lax–Wendroff), which reduces the amplitude error to $O(\beta^4)$, making the scheme second-order accurate. The phase error can be lowered to $O(\beta^5)$ by choosing $q = (1 + 2\nu^2)/3$; this yields the low-phase-error (LPE) scheme, which still is only first-order accurate because of its $O(\beta^2)$ dissipative error. The latter can be reduced to $O(\beta^4)$ by an *anti-diffusion* corrector step of the form

$$(u_j^{n+1})_{\text{cor}} = u_j^{n+1} - \frac{1}{2}(q - \nu^2)(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}) \tag{24}$$

starting from the LPE scheme, it yields a third-order-accurate result. This predictor–corrector procedure is the basis of the FCT methods by Boris.

The upwind scheme, with $q = |\nu|$, coincides with the LPE scheme for $|\nu| = \frac{1}{2}$; for this CFL number, it can be shown from (21) that the phase error is not just small but vanishes for all values of β. This is a desirable property, shared by higher-order upwind-biased advection schemes.

Time-reversible advection schemes like (14) have $|g| = 1$, hence no dissipative error; as argued in Section 3, this is not a desirable property when approximating nonlinear conservation laws. On the other hand, discretizations of the diffusion equation usually have no dispersive errors. For instance, scheme (6) has a real amplification factor, $g(\beta, r) = 1 - 2r(1 - \cos\beta)$, so it cannot affect phase.

4.3 Modified equation

Used primarily in the early years of CFD to describe properties of difference schemes, the modified equation usually does not give more information than a *truncated* Fourier expansion. For instance, the diffusion term of equation (4) on the right-hand side of the modified equation contains the same information as the damping term on the right-hand side of equation (22). Thus, for a linear equation, the full Fourier transform is to be preferred over the modified equation and, in fact, may be used to conveniently derive error terms in the modified equation up to any order. The only advantage of the modified equation is that it can also be obtained for a nonlinear difference scheme. In this case, though, it is often used beyond its applicability, in particular, inside shock structures, where the truncated right-hand side makes little sense because the power series does not converge.

4.4 Limiting

Godunov's barrier theorem (Section 3) can be circumvented by allowing an update scheme for the linear advection equation to be *nonlinear*, meaning that the scheme's coefficients may depend on the input data. There are several conceptually different ways to achieve this; most widely accepted is the notion that the subcell solution, which is either explicitly prescribed (FEM), obtained by interpolation (FVM), or tacitly assumed (FDM), must satisfy some kind of monotonicity or non-oscillatory constraint. In practice, this comes down to *limiting* the nominal derivative values of the subcell distribution in cells at the foot and head of captured discontinuities.

Non-oscillatory interpolation combined with conservation requires at least three initial data per flux evaluations. Suitable for limiting is 1968 upwind-biased second-order-accurate advection scheme by J.E. Fromm, which for $\nu \geq 0$ reads

$$\overline{u}_j^{n+1} = \overline{u}_j^n - \nu\left[\left(\overline{u}_j^n + \frac{1 - \nu}{2}\delta u_j^n\right) - \left(\overline{u}_{j-1}^n + \frac{1 - \nu}{2}\delta u_{j-1}^n\right)\right] \tag{25}$$

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

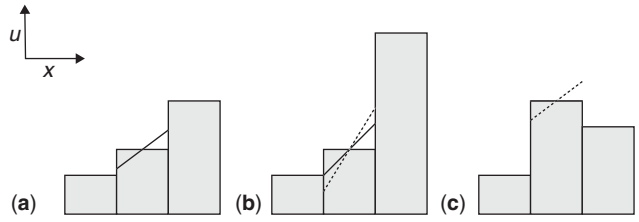

Figure 3. Non-oscillatory interpolation for Fromm's scheme. (a) Standard linear interpolation in the middle cell; (b) Unlimited (dashed) and limited (solid) linear interpolation in the middle cell, in the case that the unlimited distribution undershoots the left cell average; (c) Unlimited (dashed) and limited (solid) linear interpolation in the middle cell, in the case of a local extremum in the middle cell.

where $\delta u_j^n = \frac{1}{2}(\Delta_- \overline{u}_j + \Delta_+ \overline{u}_j)$ with $\Delta_- \overline{u}_j \equiv \overline{u}_{j+1} - \overline{u}_j$, $\Delta_+ \overline{u}_j \equiv \overline{u}_{j+1} - \overline{u}_j$; the bars over the discrete solution values again indicate that we choose to regard the latter as cell averages. To complete the finite-volume interpretation of the scheme, we assert that at time t^n in each cell, say, cell j, a linear distribution exists of the form $u(x, t^n) = \overline{u}_j^n + (x - x_j)\delta u_j^n/\Delta x, \quad x_{j-\frac{1}{2}} < x < x_{j+\frac{1}{2}}$ (see Figure 3a); applying the exact advection operator to this initial-value distribution means translating it over a distance $\nu\Delta x$ during the following time step. At time t^{n+1}, the cell contains a piecewise linear distribution; averaging it yields precisely the value $\overline{u}_j^{n+1}$ given by equation (25).

An initially nondecreasing sequence of cell averages may develop an extremum if the gradient of any subcell distribution is too large, specifically, *if the values in the subcell distribution go beyond the interval spanned by the neighboring cell averages*. This happens when at t^n the *smoothness monitor* $s_j = \Delta_+ \overline{u}_j / \Delta_- \overline{u}_j$, which in a well-resolved smooth solution takes values close to unity, exceeds 3, or is smaller than 1/3. In Figure 3b, limiting is required to restrict δu_j to *twice* the lesser of $\Delta_+ \overline{u}_j$ and $\Delta_- \overline{u}_j$. In addition, an existing extremum should not be allowed to grow; this means that the gradient in the cell with the extreme $\overline{u}$-value should be set to zero (see Figure 3c). The full limiter reads

$$(\delta u_j)_{\text{lim}} = \begin{cases} \min\{2|\Delta_- \overline{u}_j|, |\delta u_j|, 2|\Delta_+ \overline{u}_j|\}\text{sgn } \delta u_j \\ \quad \text{if sgn } \Delta_- \overline{u}_j = \text{sgn } \delta u_j = \text{sgn } \Delta_+ \overline{u}_j \\ 0 \quad \text{otherwise} \end{cases} \tag{26}$$

This formula remains valid if equation (25) represents a discontinuous Galerkin scheme (a kind of FEM), with $\overline{u}_j^n$ and δu_j^n as independent discrete data per mesh. (In that case, an extra update equation for δu_j^n is needed.)

The limiter (26), published by Van Leer in 1974, is called "double minmod"; it includes a large range of the smoothness monitor, $1/3 \le s \le 3$, for which no limiting is performed. The nonlimited, linear scheme (Fromm's) is called the *target* scheme. On the other hand, the "minmod" limiter, first used by V.V. Kolgan in 1972,

$$(\delta u_j)_{\text{lim}} = \begin{cases} \min\{|\Delta_- \overline{u}_j|, |\Delta_+ \overline{u}_j|\}\text{sgn } \Delta_- \overline{u}_j \\ \quad \text{if sgn } \Delta_- \overline{u}_j = \text{sgn } \Delta_+ \overline{u}_j \\ 0 \quad \text{otherwise} \end{cases} \tag{27}$$

which always chooses the finite-difference gradient smallest in modulus to act as the subcell gradient, has no linear target scheme. Another type of limiter that always acts is Van Leer's *harmonic* limiter also from 1974:

$$(\delta u_j)_{\text{lim}} = \begin{cases} \left[\frac{1}{2}\left(\frac{1}{\Delta_- \overline{u}_j} + \frac{1}{\Delta_+ \overline{u}_j}\right)\right]^{-1} = \frac{4s_j}{(s_j+1)^2}\delta u_j \\ \quad \text{if sgn } \Delta_- \overline{u}_j = \text{sgn } \Delta_+ \overline{u}_j \\ 0 \quad \text{otherwise} \end{cases} \tag{28}$$

It deviates from "double minmod" only by $O\{(1-s)^2\}$ when s is close to 1 and may be regarded as a smoothed version of "double minmod".

Next, we discuss the 1982 Van Albada limiter:

$$(\delta u_j)_{\text{lim}} = \frac{\{(\Delta_+ \overline{u}_j)^2 + \epsilon\}\Delta_- \overline{u}_j + \{(\Delta_- \overline{u}_j)^2 + \epsilon\}\Delta_+ \overline{u}_j}{(\Delta_+ \overline{u}_j)^2 + (\Delta_- \overline{u}_j)^2 + 2\epsilon}, \quad \epsilon > 0,\ \epsilon \sim (\Delta x)^3 \tag{29}$$

It has two properties by design: (i) near a smooth extremum, where $\Delta_\pm \overline{u}_j$ has the magnitude $O(\Delta x)^2$, the limiting action fades out, avoiding the clipping of the extremum that would result from locally setting $\Delta_\pm \overline{u}_j = 0$; (ii) $(\delta u_j)_{\text{lim}}$ is continuously differentiable with respect to the input data $\overline{u}_j^n$ and $\overline{u}_{j\pm1}^n$, which helps prevent the solution from getting trapped in a limit cycle when marching toward a steady state.

Finally, we present Roe's "Superbee" from 1985,

$$(\delta u_j)_{\text{lim}} = \begin{cases} \min\{2|\Delta_- \overline{u}_j|, 2|\Delta_+ \overline{u}_j|, \max(|\Delta_- \overline{u}_j|, |\Delta_+ \overline{u}_j|)\}\text{sgn } \delta u_j \\ \quad \text{if sgn } \Delta_- \overline{u}_j = \text{sgn } \delta u_j = \text{sgn } \Delta_+ \overline{u}_j \\ 0 \quad \text{otherwise} \end{cases} \tag{30}$$

a combination of a limiter, which reduces δu_j where there is danger of over- or undershooting, and *artificial compression*, which increases δu_j where there is no danger of over- or undershooting. The artificial compression is particularly

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

effective in steepening the numerical profile representing a contact discontinuity, which otherwise would spread with time.

For discretizations of nonlinear advection–diffusion PDEs, Harten developed in 1983 a monotonicity analysis based on monitoring the discrete solution's total variation, $\sum_j |u_j - u_{j-1}|$, which increases with the appearance of a new extremum. Schemes that do not increase their total variation are called total-variation dimishing (TVD); all above limiters except (29) render the Fromm scheme (25) TVD.

The TVD concept does not have a useful extension into multiple dimensions, but a 1986 concept by Harten and Osher, ENO interpolation, does. This is a hierarchical interpolation technique of arbitrarily high order, in which the stencil is expanded to allow the computation of the next higher derivative, by adding the nodal point or cell that will result in the lowest value of that derivative. ENO schemes are total-variation bounded (TVB); the growth of the total variation is only $O(\Delta x)^2$), in consequence of the preservation of smooth extrema without clipping. The Van Albada limiter (29) also makes a scheme TVB. ENO schemes are highly nonlinear, having a continuously varying stencil and no embedded target scheme; to overcome associated problems, C.-W. Shu 1996 developed weighted ENO (WENO), which does return to a target scheme when there is no need for limiting. WENO is used exclusively on structured grids for studying fundamental flow physics problems such as transition to turbulence.

When computing multidimensional solutions on unstructured grids, the limiter used is most often an extension of "double minmod" (equation (26)). Multidimensional limiting is still a subject of research; the ENO principle of selecting the lowest local value of a higher derivative among possible candidates is likely to prevail.

5 ANATOMY OF A FLOW SOLVER

The principles described above lead to strong constraints on CFD schemes. A practical scheme for aerodynamic problems – which often include shocks and are typically at high Reynolds numbers – must meet the constraints of consistency, stability, monotonicity, conservation, and irreversibility. Even within these constraints, however, there are many possible schemes and many decisions to be made in designing or choosing a scheme. A very brief outline of some of the decisions is given below; a more full overview can be found in Hirsch's (1989–1991) reference work on CFD.

5.1 Physical model

The first decision that affects the design of the solver is what physical processes are to be modeled. If compressibility effects are unimportant, a solver specifically designed for incompressible flow will be simpler and more efficient than the one that is able to span a range of Mach numbers. Turbulence in flows can be treated in a number of different ways; this can have profound effects on the design of the solver. For flows that are assumed to be entirely laminar (a small percentage of aerodynamic flows) or for flows in which the turbulent length and time scales can be fully resolved on the chosen grid (an even smaller percentage of aerodynamic flows, see Direct Numerical Simulation of Turbulent Fluid Flow), no separate closure is necessary for modeling turbulence. For the vast majority of aerodynamic flows, however, turbulence will be present but not resolvable on the grid. In this case, a stable discretization of the Reynolds stress terms, subgrid-scale terms, and additional equations for quantities required to close the equation set may be necessary. (See Reynolds-Averaged Navier–Stokes Methods for a discussion of Reynolds-averaged Navier–Stokes approaches and Large Eddy Simulation for a discussion of large-eddy simulation.)

Other physical effects may also be present: chemical reactions, multi-phase flows, and fluid–structure interactions are three of the more common ones in aerodynamics. Reacting flows lead to large, stiff systems of ordinary differential equations coupled to the flow equations; multi-phase flows require tracking of interfaces between materials and special treatment at the interfaces; fluid–structure interactions require careful design of numerical boundary procedures, and often lend themselves to Lagrangean or mixed Lagrangean/Eulerian treatment of the flow.

5.2 Basic discretizations and grids

As mentioned previously, the three most common discretizations lead to finite-difference, finite-volume, and finite-element methods. In addition to the choice of discretization, a grid type is also chosen. Finite-difference methods are commonly used on structured grids, in which the grid points can naturally be stored in array-type data structures. Finite-volume and finite-element techniques are preferred on grids that do not lend themselves to arrays; linked-list data structures are typically used in this case. In addition, grids can be constructed from multiple abutting array-structured blocks (multi-block grids), overlapping array-structured blocks (overset grids), or dynamically adapting sets of points (adaptive grids, see Grid Generation Techniques).

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

5.3 Spatial discretization of advective and viscous terms

The advective and viscous terms of the Navier–Stokes equations pose separate challenges. Because of the nonlinearity inherent in the advective terms, stability and monotonicity dominate in the design of their discretization. For the compressible Navier–Stokes equations, the choice of flux function and limiter are key aspects of the scheme. Incompressible Navier–Stokes solvers are not typically cast in terms of flux functions, but the choice of limiter is key. For the viscous terms, accuracy dominates the choice of discretization.

5.4 Temporal discretization

In almost all practical aerodynamic solvers, stability considerations require a scheme with some implicit character. That is, the scheme cannot be written as

$$\frac{\bar{\boldsymbol{U}}_j^{n+1} - \bar{\boldsymbol{U}}_j^{n}}{\Delta t} = \boldsymbol{R}(\boldsymbol{U}^n) \tag{31}$$

but must be written as

$$\frac{\bar{\boldsymbol{U}}_j^{n+1} - \bar{\boldsymbol{U}}_j^{n}}{\Delta t} = \boldsymbol{R}(\boldsymbol{U}^n, \boldsymbol{U}^{n+1}) \tag{32}$$

where $\boldsymbol{R}$ is called the *residual*; the scheme represented by equation (9) can be written in this form. An implicit scheme such as this requires an approximate linearization of $\boldsymbol{R}$ and a method, typically iterative, of solving the resulting large system of simultaneous equations. For high-Reynolds-number flows, the resulting systems, besides being large, are poorly conditioned and must be preconditioned to accelerate convergence of the iterative method.

6 CONCLUDING REMARKS

CFD is playing a major role in aerospace analysis and design. Conversely, aerospace engineering is leading the way in CFD development for many disciplines. Owing to the increasing power of the computers on which they are run, CFD codes are now able to tackle more and more complex problems. As codes and applications become more sophisticated, there is still a core of principles and methods for design and analysis of CFD codes that remains indispensable.

REFERENCES

Barth, T.J. and Deconinck, H. (eds) (1999) *High-Order Methods for Computational Physics*, Springer.

Beam, R.M. and Warming, R.F. (1978) An implicit factored scheme for the compressible Navier–Stokes equations. *AIAA J.*, **16**, 393–402.

Brackbill, J.U. (ed.) (1997) *J. Comput. Phys.*, **35**(2), (30th Anniversary Issue).

Deconinck, H. and Koren, B. (eds) (1997) *Euler and Navier–Stokes Solvers Using Multidimensional Upwind Schemes and Multigrid Acceleration*, Vieweg, vol. 57 in the series *Notes on Numerical Fluid Mechanics*.

Drikakis, D. and Rider, W. (2005) *High-Resolution Methods for Incompressible and Low-Speed Flows*, Springer.

Godunov, S.K. (1959) A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.*, **47**, 271–306.

Godunov, S.K. (1999) Reminiscences about difference schemes. *J. Comput. Phys.*, **153**, 6–25.

Hafez, M. (ed.) (2000) CFD and education. *Comput. Fluid Dyn. J.*, **9**, 3 (Special issue dedicated to K. Oshima).

Harten, A., Lax, P.D. and van Leer, B. (1983) Upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, **25**, 35–61.

Hess, J. and Smith, A. (1966) Calculation of potential flow about arbitrary bodies. *Progr. Aeronaut. Sci.*, **8**, 1–138.

Hirsch's, C. (1989-1991) *Numerical Computation of Internal and External Flows*, Wiley (two volumes, paperback edition).

Hirschel, E.H. and Krause, E. (eds) (2009) *Notes on Numerical Fluid Mechanics – 40 Years of Numerical Fluid Mechanics and Aerodynamics in Retrospect*, Springer, vol. 100 in the series *Notes on Numerical Fluid Mechanics*.

Hussaini, M.Y., van Leer, B. and van Rosendale, J.H. (eds) (1997) *Upwind and High-Resolution Schemes*, Springer.

Jameson, A. (1989) Computational aerodynamics for aircraft design. *Science*, **245**, 361–371.

Jameson, A., Baker, T.J. and Weatherill, N.P. (1986) Calculation of inviscid transonic flow over a complete aircraft. AIAA 86-0103.

Jameson, A., Schmidt, W. and Turkel, E. (1981) Numerical solutions of the Euler equations by a finite-volume method using Runge–Kutta time-stepping schemes. AIAA 81-1259.

Murman, E. and Cole, J. (1971) Calculations of plane steady transonic flow. *AIAA J.*, **9**, 114–121.

Oran, E.S. and Boris, J.P. (eds) (2001) *Numerical Simulation of Reactive Flow*, 2nd edn, Cambridge University Press.

Richtmyer, R.D. and Morton, K.W. (1967) *Difference Methods for Initial-Value Problems*, Interscience.

Roe's, P.L. (1981) Approximate Riemann solvers, parameter vectors and difference schemes. *J. Comput. Phys.*, **43**, 357–372.

Rumsey, C.L., Biedron, R.T. and Thomas, J.L. (1997) CFL3D: its history and some recent applications. *Technical Report TM 112861*, NASA.

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048

Van Leer, B. (1979) Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *J. Comput. Phys.*, **32**, 101–136.

Van Leer, B. (2000) CFD education: past, present, future. *Comput. Fluid Dyn. J.*, **9**, 157–163.

Van Leer, B. (2009) Development of numerical fluid mechanics and aerodynamics since the 1960s: US and Canada, in vol. 100, *Notes on Numerical Fluid Mechanics – 40 Years of Numerical Fluid Mechanics and Aerodynamics in Retrospect* (eds E.H. Hirschel and E. Krause), Springer, pp. 159–185.

View publication stats

Encyclopedia of Aerospace Engineering, Online © 2010 John Wiley & Sons, Ltd.
This article is © 2010 John Wiley & Sons, Ltd.
This article was published in the *Encyclopedia of Aerospace Engineering* in 2010 by John Wiley & Sons, Ltd.
DOI: 10.1002/9780470686652.eae048